SINGLE-LAYER WIRE ROUTING

by

FRANKLIN MILLER MALEY

B.A., Mathematics and Physics, May 1983
Amherst College
M.S., Computer Science, June 1986
Massachusetts Institute of Technology

Submitted to the Department of
Electrical Engineering and Computer Science
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

at the
Massachusetts Institute of Technology
August, 1987

© F. Miller Maley, 1987

The author hereby grants to MIT permission to reproduce and to
distribute copies of this thesis document in whole or in part.

Signature of Author

Department of Electrical Engineering and Computer Science
August 7, 1987

Certified by

Charles E. Leiserson
Associate Professor of Computer Science and Engineering

Accepted by

Arthur Smith
Chairman, Department Committee on Graduate Students

- 1 -
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
MAR 22 1988

ARCHIVES
The research presented in this thesis was performed in the Theory of Computation Group of the MIT Laboratory for Computer Science. Funding was provided in part by a Graduate Fellowship from the Office of Naval Research, and in part by the Defense Advanced Research Projects Agency under contract N00014-80-C-0622.
Single-Layer Wire Routing

by

F. Miller Maley

Submitted on 7 August 1987 to the Department of
Electrical Engineering and Computer Science
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

Abstract

This dissertation concerns the problem of routing wires on a single layer of an
integrated circuit or printed circuit board, starting from a sketch of the layer. A
sketch specifies the positions of layout features and the topology of the interconnect-
ing wires. Efficient algorithms are presented that (1) determine whether a sketch
is routable, and (2) produce for a routable sketch a proper routing that minimizes
both individual and total wire length. Both algorithms run in time \( O(n^2 \log n) \) on
input of size \( n \), and both are simple to implement. They can be adapted to a variety
of wiring models, and they subsume most of the polynomial-time algorithms in the
literature for single-layer routing and routability testing.

The algorithms are based on two theorems concerning the routings of a sketch.
One states that a sketch is routable if and only if for each cut between fixed features,
the total amount of wiring forced to cross the cut is no greater than the length of
the cut. The second theorem states that every routable sketch has a routing that
simultaneously minimizes the length of every wire, and it characterizes the wires in
this routing. To formalize and prove these theorems, a rich mathematical theory
of single-layer wire routing is developed. Its central tool, which is new to the wire-
routing literature, is the lifting of wires and cuts to a simply connected topological
covering space of the routing region.

As another application of this theory, the thesis presents a general algorithm for
one-dimensional layout compaction. Given a routable sketch, it finds a proper sketch
of minimal width obtainable by displacing the features horizontally and moving the
wires, always maintaining routability. Thus it automatically inserts into wires all
jog points that help in compressing the layout. In the worst case the compaction
algorithm uses time \( O(n^4) \) and space \( O(n^3) \) on input of size \( n \). The technique on
which the algorithm is founded is nearly independent of the wiring model, and it
applies to many-layer as well as single-layer compaction problems.

Key words: channel routing, compaction, computational geometry, constraint solv-
ing, covering space, homotopy, global routing, graph algorithms, jog insertion, river
routing, routability, routing, topology, VLSI layout, wiring, wire length minimization.

Thesis Supervisor: Charles E. Leiserson
Title: Associate Professor of Computer Science and Engineering
Preface

This dissertation is the product of a four-year study on the general problem of wire routing under separation and homotopy constraints. Originally intended as a master's thesis, the project quickly grew out of control when repeated attempts to solve the fundamental problems ended in failure. The driving force behind the growth was a desire for mathematical rigor. I devised the central algorithm of this thesis, the sketch routing algorithm, and was convinced of its correctness, long before finding any technical justification for it. All attempted correctness proofs using elementary tools broke down, and the breakdowns could be traced to a single source: a lack of technical tools for dealing with the concept of homotopy at the heart of the routing problem. Since homotopy is a topological notion, I turned to algebraic topology, and thus was born the theory that accounts for the bulk of this thesis.

Though my approach to single-layer wire routing has been lengthier and more involved than one might like, I expect it to support further fruitful work on wire routing, both practical and theoretical. This research has had two goals: to establish certain theorems and algorithms concerning wire routing and compaction, and to blaze a trail through the vast terrain between homotopy theory and circuit design. The tension between these aims accounts for the technical depth of this study. To read it carefully is likely to be a laborious task; yet I hope scholars of algorithms will find it rewarding. Being a worked-out example of wire routing in two specific models, this dissertation may serve as a source of ideas and a prototype for studies of other models of wiring. Subsequent treatments should be simpler, or at least easier, with the steps and missteps of this thesis as a guide. And as a first cut at a theory of single-layer routing, it demonstrates the power of bringing topological concepts to bear on routing problems.

Organization and prerequisites

Because this thesis addresses topics that run from topology through algorithms and circuit design, I have tried to make it accessible to specialists and students in
several areas. The danger, of course, is that I might make it accessible to nobody. To guard against that possibility, I have separated the algorithms from the underlying mathematics, and confined the advanced topology to a pair of chapters, namely Chapters 2 and 3. The glossary includes definitions of mathematical terms that may be unfamiliar, and I have provided a table of notations on pages 10–12.

Those who are primarily interested in wire routing and compaction should read Chapter 1, which shows how to solve routability and routing problems, and Chapter 9, which presents and justifies a compaction procedure. Chapter 10 discusses refinements and extensions of these algorithms. When describing algorithms, I assume some knowledge of the techniques of computational geometry and algorithmic graph theory.

Those who are interested in the application of topology to routing problems should read the remaining chapters, beginning with the definitions in Chapter 2. The core of this dissertation is the development of a theory of single-layer wire routing in Chapters 3 through 7. Chapter 8 uses this theory to derive results about the sketch model. Most of these chapters require familiarity with point-set topology; a knowledge of elementary homotopy theory is also helpful. For those with no prior exposure to algebraic topology, I have provided a short introduction to homotopy theory in Chapter 2.

Acknowledgements

This thesis, and the degree it represents, could never have been completed without the help and support of three people. One is my mother, Ann Maley, whose love and encouragement kept me going at the most difficult times. Another is my thesis advisor, Charles Leiserson, who, along with his former student Ron Pinter, provided the starting point and the motivation for this entire line of research; they discovered the connection between routability conditions and compaction with automatic jog insertion. Chapter 1 is derived from a joint paper [21] with Prof. Leiserson. Charles also taught me how to write and speak on technical matters, and has contributed greatly to this exposition in innumerable discussions. The third is John Baez, my former roommate and good friend, who rekindled my interest in pure mathematics by introducing me to algebraic topology. Discussions with John convinced me to take seriously my idea of using covering spaces to study wire routing. Without the inspiration that arose from those and further conversations, my program of research would have stagnated. I wish to express my heartfelt appreciation for all that these people have done for me.

Many other people have contributed to this thesis in great and small ways. I especially thank my other two readers, Prof. David Anick and Prof. Bill Dally, for their willingness to serve on my thesis committee, and their helpful comments.
and referrals. In a pair of wonderful courses (or subjects), Prof. Anick taught me everything I know about algebraic topology, and for this too I am grateful. Thanks also to Bonnie Berger, Bard Bloom, Ray Hirschfeld, Joe Kilian, Bruce Maggs, and Su-Ming Wu for reading and pasting figures into various drafts of the thesis. Their help was instrumental in finishing the final copy.

Part of the inspiration for this research came from the success of actual programs. Encouraging experimental results were provided by Michel Doreau of Digital Equipment Corporation, whose use of cut constraints in his PCB router “TWIGGY” first suggested the sketch routability theorem. Some special-case compaction algorithms developed by Leiserson and Pinter were implemented by Andrew Hume of at AT&T Bell Laboratories, and used for channel routing.

Over the years, several people have offered technical advice on improving my proofs and presentations, among them John Baez, Ravi Boppana, Ron Greenberg, and Johan Hastad. I am grateful for their insights. I would also like to thank the referees and editors of my published papers, particularly Prof. Thomas Lengauer, for their comments and encouragement.

Readers will note that I take pleasure in inventing fanciful and amusing terminology. I cannot take all the credit for my strained analogies and mixed metaphors, however; those who suggested terms I later adopted include Ray Hirschfeld (who is responsible for ‘blanket’), Phil Klein, Charles Leiserson, Mark Newman (‘embedding’ into a sheet), and Alan Sherman.

Thanks to Bard Bloom and Joe Kilian for providing places to live when I overran the Spring 1987 thesis deadline ... and to Albert Meyer for his comfortable couch.

Finally, I wish to acknowledge the organizations and people who supported my research environment. This thesis was carried out using office space and computing resources of the Theory of Computation Group of the MIT Laboratory for Computer Science. I thank all the members of the TOC group for making it a great place to work. I am especially grateful to its underrecognized software and hardware gurus, Ray Hirschfeld and Mark Reinhold, for their constant support of our computing environment and their willingness to answer any computer-related question, however naive. Thanks also to the Academic Computer Center at Amherst College and its friendly denizens for computer time and blackboard space. Funding for research-related expenses was provided by the Defense Advanced Research Projects Agency. Last but far from least, I thank the Office of Naval Research and the American Society for Engineering Education for the generous fellowship that has supported me over the past four years.

F. M. M.

Cambridge, Massachusetts
August, 1987
# Table of Contents

Introduction: **Single-Layer Wire Routing**  
A. Background .................................................. 14  
B. Thesis Overview ........................................... 20  

Chapter 1: **Sketch Algorithms**  
1A. The Sketch Model ........................................... 28  
1B. The Rubber-Band Equivalent of a Sketch ................. 33  
1C. Testing the Routability of a Sketch .................... 40  
1D. Routing a Sketch ......................................... 44  
1E. Efficiency Concerns .................................... 50  
1F. Faster Routability Testing ............................... 54  

Chapter 2: **Topological Preliminaries**  
2A. Homotopies and the Fundamental Group ................ 65  
2B. Covering Spaces ........................................... 69  
2C. Paths and Loops in the Plane .......................... 76  
2D. Topological Manifolds .................................. 79  

Chapter 3: **The Topology of Blankets**  
3A. Constructing Paths in Blankets ......................... 88  
3B. Separation Results ...................................... 93  
3C. Properties of Separations ............................... 99  
3D. Elastic Chains in Sheets ............................... 104  

Chapter 4: **Flow Across Cuts and Half-Cuts**  
4A. The Design Model ......................................... 111  
4B. Flow: A Characterization of Congestion ................. 115  
4C. Relations Among Cuts and Wires ....................... 122  
4D. Properties of Flow ...................................... 128  
4E. The Branches of a Blanket ............................. 135  
4F. Safety of Cuts and Half-Cuts ......................... 140  

Chapter 5: **Routing a Safe Design**  
5A. Construction of Ideal Routes ......................... 148  
5B. Construction of Safe Routes ............................ 152
Table of Contents

5B. Ideal Routes Are Taut ........................................ 160
5C. Ideal Routes Form a Design .................................... 168
5D. Ideal Designs Are Properly Connected ...................... 171
5E. Ideal Wires Are Self-Avoiding ................................ 176

Chapter 6: Routability Conditions for Designs ..................... 182
6A. Unsafe Designs Are Unroutable ................................. 183
6B. Ideal Embeddings Have Optimal Length ...................... 189
6C. Summary of Design Theorems .................................. 194
6D. Cuts That Decide Routability .................................. 196

Chapter 7: From Theory to Algorithms ............................... 203
7A. Geometric Representations of Path Classes .................. 204
7B. Crossing Sequences ............................................ 211
7C. Two Methods for Computing Plans .............................. 218
7D. The Geometry of Ideal Wires .................................. 226
7E. Routing Through a Maze ....................................... 235

Chapter 8: Return to the Sketch Model .............................. 242
8A. The Correspondence Between Designs and Sketches .......... 243
8B. Sketch Theorems ................................................ 249
8C. Correctness of the Sketch Algorithms ......................... 255

Chapter 9: Sketch Compaction ...................................... 261
9A. Problem Statement ............................................. 265
9B. Computing Flows During Compaction .......................... 268
9C. The Compaction Algorithm .................................... 275
9D. The Adjacency Graph of a Sketch .............................. 281
9E. The Abstract Compaction Algorithm .......................... 287
9F. Implementing the Abstract Algorithm .......................... 294

Chapter 10: Extensions of the Theorems and Algorithms .......... 300
10A. Representation Issues ........................................ 303
10B. Wiring Rules and Wiring Norms ............................... 310
10C. The Terminals of Traces ..................................... 318
10D. An Alternative to the Sketch Model ......................... 325

Conclusion: A Critical Review ..................................... 331
A. Summary of Results ............................................ 331
B. Directions for Future Research ................................ 335

Glossary .......................................................... 339
References ....................................................... 358
# Table of Symbols

For the most part, uppercase Roman letters denote data structures or topological spaces, lowercase Roman letters represent points in those spaces, lowercase Greek letters denote paths, and uppercase Greek letters represent sets of paths or points.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\overline{xy}$</td>
<td>Line segment with endpoints $x$ and $y$</td>
<td>17</td>
</tr>
<tr>
<td>$|\cdot|$</td>
<td>A norm, the <em>wiring norm</em>, on the plane $\mathbb{R}^2$</td>
<td>29</td>
</tr>
<tr>
<td>$|P - Q|$</td>
<td>Distance between the regions $P$ and $Q$</td>
<td>30</td>
</tr>
<tr>
<td>$</td>
<td>D</td>
<td>$</td>
</tr>
<tr>
<td>$O(f)$</td>
<td>Functions $g$ such that $</td>
<td>g(n)</td>
</tr>
<tr>
<td>$\Omega(f)$</td>
<td>Functions $g$ such that $g(n) \geq cf(n)$ for some $c &gt; 0$ as $n \to \infty$</td>
<td>44</td>
</tr>
<tr>
<td>$\Theta(f)$</td>
<td>Functions in both $O(f)$ and $\Omega(f)$</td>
<td>50</td>
</tr>
<tr>
<td>$I$</td>
<td>Unit interval $[0,1]$</td>
<td>62</td>
</tr>
<tr>
<td>$\mathbb{R}$</td>
<td>Real line</td>
<td>62</td>
</tr>
<tr>
<td>$\mathbb{R}^2$</td>
<td>Cartesian plane</td>
<td>62</td>
</tr>
<tr>
<td>$\mathbb{H}^2$</td>
<td>Upper closed half-plane of $\mathbb{R}^2$</td>
<td>63</td>
</tr>
<tr>
<td>$\mathbb{S}^1$</td>
<td>Unit circle in the plane $\mathbb{R}^2$</td>
<td>63</td>
</tr>
<tr>
<td>$Int A$</td>
<td>Topological interior of $A$</td>
<td>63</td>
</tr>
<tr>
<td>$Cl A$</td>
<td>Topological closure of $A$</td>
<td>63</td>
</tr>
<tr>
<td>$Fr A$</td>
<td>Frontier or topological boundary of $A$</td>
<td>63</td>
</tr>
<tr>
<td>$t \mapsto E(t)$</td>
<td>Function that takes $t$ to $E(t)$</td>
<td>63</td>
</tr>
<tr>
<td>$F(x, \cdot)$</td>
<td>Function that takes $y$ to $F(x, y)$</td>
<td>63</td>
</tr>
<tr>
<td>$f</td>
<td>_U$</td>
<td>Restriction of $f$ to $U$</td>
</tr>
<tr>
<td>$id_X$</td>
<td>Identity map on the space $X$</td>
<td>63</td>
</tr>
<tr>
<td>$Im \phi$</td>
<td>Image of the function $\phi$</td>
<td>63</td>
</tr>
<tr>
<td>$\alpha: A \to B$</td>
<td>The path $\alpha$ runs from $A$ to $B$</td>
<td>63</td>
</tr>
<tr>
<td>$Mid \alpha$</td>
<td>Middle $\alpha((0,1))$ of the path $\alpha$</td>
<td>63</td>
</tr>
<tr>
<td>$x \triangledown y$</td>
<td>Linear path from $x$ to $y$</td>
<td>64</td>
</tr>
<tr>
<td>$\alpha_{a:b}$</td>
<td>Subpath of $\alpha$ from $\alpha(a)$ to $\alpha(b)$</td>
<td>64</td>
</tr>
<tr>
<td>$\alpha \star \beta$</td>
<td>Concatenation of $\alpha$ and $\beta$</td>
<td>64</td>
</tr>
<tr>
<td>Symbol</td>
<td>Meaning</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>$\hat{\alpha}$</td>
<td>Reverse of the path $\alpha$, namely $\alpha_{1,0}$</td>
<td>64</td>
</tr>
<tr>
<td>$|\alpha|$</td>
<td>Arc length of $\alpha$ in the norm $| \cdot |$</td>
<td>64</td>
</tr>
<tr>
<td>$</td>
<td>\alpha</td>
<td>$</td>
</tr>
<tr>
<td>$\alpha \simeq_P \beta$</td>
<td>The paths $\alpha$ and $\beta$ are path-homotopic</td>
<td>65</td>
</tr>
<tr>
<td>$[\alpha]_P$</td>
<td>Set of paths that are path-homotopic to $\alpha$</td>
<td>66</td>
</tr>
<tr>
<td>$[\alpha]_P \star [\beta]_P$</td>
<td>Equivalent to $[\alpha \star \beta]_P$</td>
<td>66</td>
</tr>
<tr>
<td>$\pi_1(X, x_0)$</td>
<td>Fundamental group of $X$ at $x_0$</td>
<td>66</td>
</tr>
<tr>
<td>$f_*$</td>
<td>Homomorphism of fundamental groups induced by $f$</td>
<td>67</td>
</tr>
<tr>
<td>$\text{Ker } \phi$</td>
<td>Kernel of the homomorphism $\phi$</td>
<td>68</td>
</tr>
<tr>
<td>$f \simeq g$</td>
<td>The maps $f$ and $g$ are homotopic</td>
<td>68</td>
</tr>
<tr>
<td>$\tilde{g}$</td>
<td>Lift of the map $g$</td>
<td>71</td>
</tr>
<tr>
<td>$\text{inside}(\lambda)$</td>
<td>Inside of the simple loop $\lambda$</td>
<td>76</td>
</tr>
<tr>
<td>$\text{outside}(\lambda)$</td>
<td>Outside of the simple loop $\lambda$</td>
<td>76</td>
</tr>
<tr>
<td>$\text{Bd } M$</td>
<td>Boundary of the manifold $M$</td>
<td>79</td>
</tr>
<tr>
<td>$\text{Bd } f$</td>
<td>Restriction of $f$ to the boundary of its domain</td>
<td>80</td>
</tr>
<tr>
<td>$\bigoplus P_i$</td>
<td>Topological sum or disjoint union of the spaces $P_i$</td>
<td>80</td>
</tr>
<tr>
<td>$\alpha \simeq_L \beta$</td>
<td>The links $\alpha$ and $\beta$ are link-homotopic</td>
<td>88</td>
</tr>
<tr>
<td>$[\alpha]_L$</td>
<td>Set of links homotopic to $\alpha$</td>
<td>88</td>
</tr>
<tr>
<td>$\text{left}(\alpha)$</td>
<td>Left scrap of the simple link $\alpha$</td>
<td>99</td>
</tr>
<tr>
<td>$\text{right}(\alpha)$</td>
<td>Right scrap of the simple link $\alpha$</td>
<td>99</td>
</tr>
<tr>
<td>$\text{width}(X, \Omega)$</td>
<td>Width of the detail $X$ in the design $\Omega$</td>
<td>113</td>
</tr>
<tr>
<td>$\text{cap}(\chi, \Omega)$</td>
<td>Capacity of $\chi$ in the design $\Omega$</td>
<td>114</td>
</tr>
<tr>
<td>$\text{cross}(\alpha, \beta)$</td>
<td>Number of crossings between $\alpha$ and $\beta$</td>
<td>114</td>
</tr>
<tr>
<td>$\text{tangle}(\chi, \omega)$</td>
<td>Entanglement of the link $\omega$ with the cut $\chi$</td>
<td>114</td>
</tr>
<tr>
<td>$\text{cong}(\chi, \Omega)$</td>
<td>Congestion of the cut $\chi$ in the design $\Omega$</td>
<td>114</td>
</tr>
<tr>
<td>$\text{wind}(\chi, \omega)$</td>
<td>Winding of the links $\chi$ and $\omega$</td>
<td>118</td>
</tr>
<tr>
<td>$\text{flow}(\chi, \Omega)$</td>
<td>Flow across the link $\chi$ in the design $\Omega$</td>
<td>119</td>
</tr>
<tr>
<td>$\text{margin}(\chi, \Omega)$</td>
<td>The difference $\text{cap}(\chi, \Omega) - \text{flow}(\chi, \Omega)$</td>
<td>140</td>
</tr>
<tr>
<td>$P(\sigma)$</td>
<td>Polygon ${ x : |x - \sigma(0)| = |\sigma| }$</td>
<td>226</td>
</tr>
<tr>
<td>$\dot{\alpha}$</td>
<td>Angle at which the path $\alpha$ travels</td>
<td>227</td>
</tr>
<tr>
<td>$\delta^\perp$</td>
<td>Angle of the segment of the unit polygon ending at $\delta$</td>
<td>227</td>
</tr>
<tr>
<td>$\delta^\top$</td>
<td>Angle of the segment of the unit polygon starting at $\delta$</td>
<td>227</td>
</tr>
<tr>
<td>$[\delta, \theta]$</td>
<td>Angles lying between $\delta$ and $\theta$ clockwise</td>
<td>228</td>
</tr>
<tr>
<td>$R(\sigma)$</td>
<td>Interval $[\dot{\sigma}^\perp, \dot{\sigma}^\top]$</td>
<td>228</td>
</tr>
<tr>
<td>$L(\sigma)$</td>
<td>Interval $[-\dot{\sigma}^\perp, -\dot{\sigma}^\top] = R(-\sigma)$</td>
<td>228</td>
</tr>
<tr>
<td>$\beta^c$</td>
<td>Transformation from sketch model to design model</td>
<td>244</td>
</tr>
<tr>
<td>$\beta^c_e$</td>
<td>Transformation from design model to sketch model</td>
<td>244</td>
</tr>
<tr>
<td>$\theta^b$</td>
<td>Abbreviation for $\beta_e(\theta)$</td>
<td>244</td>
</tr>
</tbody>
</table>
# Table of Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega^i$</td>
<td>Abbreviation for $\xi^i(\theta)$</td>
<td>244</td>
</tr>
<tr>
<td>$\alpha^i$</td>
<td>Abbreviation for $\xi^i(\beta^i(\theta))$</td>
<td>244</td>
</tr>
<tr>
<td>$\mu(p)$</td>
<td>Number of the module that contains the point $p$</td>
<td>265</td>
</tr>
<tr>
<td>$x_p, y_p$</td>
<td>Coordinates of the point $p$</td>
<td>265</td>
</tr>
<tr>
<td>$p(d)$</td>
<td>Point to which configuration $d$ moves the point $p$</td>
<td>265</td>
</tr>
<tr>
<td>$\Delta p_q(d)$</td>
<td>The $x$-coordinate of $q(d)$ minus that of $p(d)$</td>
<td>265</td>
</tr>
<tr>
<td>$\Delta_{PQ}(d)$</td>
<td>Difference in displacements of the modules $\mu(Q)$ and $\mu(P)$</td>
<td>266</td>
</tr>
<tr>
<td>$C(S)$</td>
<td>Configuration space of the modular sketch $S$</td>
<td>266</td>
</tr>
<tr>
<td>$h \circ S$</td>
<td>Image of sketch $S$ under the homeomorphism $h: R^2 \rightarrow R^2$</td>
<td>266</td>
</tr>
<tr>
<td>$\chi_{PQ}$</td>
<td>Critical potential cut from feature endpoint $p$ to feature $Q$</td>
<td>275</td>
</tr>
<tr>
<td>$\phi_{pq}$</td>
<td>Potential cut $d \mapsto p(d) \triangleright q(d)$</td>
<td>276</td>
</tr>
</tbody>
</table>
Introduction

Single-Layer Wire Routing

A problem that frequently arises in the design of computer components is that of routing wires through some interconnection medium. Most wire-routing problems are computationally hard: determining whether an instance of a routing problem is even solvable is usually NP-complete. In this thesis I show that if the wires are restricted to a single planar layer, and if rough routings of the wires with respect to the routing obstacles are given, then the wires can be routed efficiently and optimally. 'Efficiently' means that the routing algorithm runs in polynomial time, and 'optimally' means that it simultaneously minimizes the length of every wire. To say it another way: Given the topology of a circuit layer, one can quickly produce a legal and nonwasteful geometry for that layer, or determine that no legal geometry is compatible with the given topology. Figure 1 illustrates this kind of routing problem.

Figure 1. An instance of a one-layer routing problem. The wires (grey paths) in the layout of panel (a) are rough routings. They are to be deformed into nonintersecting paths in the grid (dotted lines) shown in part (b), with their endpoints kept fixed and without moving them onto or across any features (dark points and lines). Panel (b) shows a solution with minimum wire length.

The fundamental fact about single-layer wire routing, which I prove, is that local routability conditions are necessary and sufficient for global routability. Consider
the layout in Figure 2. The wires cannot be routed: the topology forces too many wires to pass between the obstacles $A$ and $B$. In other words, the channel between $A$ and $B$ has greater congestion than capacity. The routability of layouts like those in Figures 1 and 2 is completely determined by the congestions and capacities of channels. This result leads to efficient algorithms for testing routability, and also to novel algorithms for layout compaction. I present these algorithms here.

**Figure 2.** An unroutable instance. This layout cannot be routed in the given topology because the channel between obstacles $A$ and $B$ is overfull. More precisely, the cut (striped segment) has space for only three crossings by wires, and all four crossings of this cut are necessary, despite the fact that only three distinct wires cross it.

What makes single-layer routing difficult and interesting is the possibility that different parts of the same wire may interact. As Figure 2 shows, a wire can pass through a channel more than once, and the different parts of the wire in that channel are constrained differently. Thus a wire behaves in some ways like several wires and in some ways like a single wire. To confront this issue I bring in ideas from homotopy theory and show how to analyze single-layer wiring by lifting wires and cuts from the routing region to its simply connected covering space. The covering space lets us formalize and work with the notions of the amount of wiring “forced” to pass across a cut, the regions that are “forbidden” to a wire, and the “necessary” crossings of a cut by wires, all of which play major roles in one-layer routing problems.

**A. Background**

This section puts my routing problem—which will be defined formally in Section 1A—into the context of other wire-routing problems, and it explains how that problem grew out of earlier work. It shows how single-layer routing with rough routings given generalizes the “river routing” problems previously studied, and how further generalizations lead to NP-complete problems. Considerations like these provide the theoretical impetus for my work. The following section offers an outline of the thesis itself and an introduction to its main ideas.
Types of wire-routing problems

Wire-routing problems abound, but they share some common characteristics. The wires, when routed, must connect certain points called terminals in a specified pattern, and they must satisfy some geometric constraints such as having a certain minimum thickness and separation from one another. Additional constraints may be imposed on the wires, e.g., that they be composed of rectilinear segments. The space in which wires are to be placed is called the routing region. In almost all practical problems, the routing region consists of one or more planes, or layers, with wires being allowed to pass between layers only at certain points.

The character of a wire-routing problem depends largely upon the topology of the routing region. Multilayer routing problems are usually NP-complete [51], even when the routing region has a simple shape. For this reason, much of the theoretical work on multilayer wire routing has concentrated on approximation algorithms [1, 4, 45]. These algorithms do not attempt to route within a fixed region, but instead they produce wirings that approach optimality in terms of the routing space or the number of wires they use. Single-layer routing problems are also NP-complete in the general case [20, 44]. Several restricted single-layer routing problems are known to be efficiently solvable, however, including those in which the routing region is simply connected [8, 22, 41, 49, 52] or annular [2] and the terminals lie on its boundary. One can also efficiently route edge-disjoint paths through a planar graph, provided that the terminals lie on a single face of the graph [3, 17, 32, 42]. Such routings are said to be in “knock-knee” mode. One can then convert the edge-disjoint paths into multilayer routings [5, 42].

The tractable routing problems are of three kinds. In a pure routing problem, the routing region and the terminals are fixed; the algorithm must determine whether the wires can be routed, and if so, find feasible realizations (or detailed routings) for them. In most single-layer routing problems, one can also minimize the length of every wire, which is desirable from a practical standpoint. Sometimes one asks only whether the wires can be routed at all; then one is concerned with a routability problem. The NP-completeness results mentioned above apply to routability problems. In a placement problem, one thinks of the terminals as being attached to modules which can move. As modules move, the shape of the routing region may change. The issue is to find placements for the modules and feasible realizations for the wires so as to minimize some geometric quantity like the area of the routing region.

Wiring models

When studying algorithms for wire-routing problems, one must work at a more abstract level than that of physical devices. One needs a mathematical wiring model
for the wires and the rules they must obey. For example, wires are usually represented as paths without thickness, but the minimum spacing between the (abstract) wires is increased to allow for the thickness of the actual (physical) wires. If one works with wires of differing thicknesses or materials, then the minimum separation between two wires will depend on which wires they are.

The wiring model most popular among theorists is what I call the grid-based model. It achieves simplicity and convenience without hiding any of the essential difficulties of placement and routing. In this model the routing region is overlaid with a rectilinear grid, and wires are required to be disjoint paths within the grid. The spacing between the gridlines corresponds to the minimum separation between wires. Other common models dispense with the grid and allow wires to contain diagonal segments or even circular arcs. Some models also permit different wires to have different separation requirements. My routing problems provide these options, but the examples in this Introduction stick to the grid model.

River routing

A single-layer routing problem that is well understood is the one-layer river routing problem [8] as refined by Leiserson and Pinter [22]. I state it for the grid-based wiring model, although other wiring models may be substituted [48]. The routing region is a rectangular channel, and the problem is to connect terminals \( A_1, \ldots, A_n \) on its bottom edge with corresponding terminals \( B_1, \ldots, B_n \) on its top edge. See Figure 3. Wires must be vertex-disjoint paths in the rectilinear grid with integer gridpoints; all the terminals are assumed to have integral coordinates. For technical convenience, the wires are allowed to run along the bottom gridline of the channel, but not along the top. The terminals \( B_1, \ldots, B_n \) must be in the same order as \( A_1, \ldots, A_n \), or else the wires would have to intersect, since the grid is planar.

![Figure 3. River routing. An instance of the problem of river routing in a rectangular channel: connect pairs of terminals (dark points) by nonintersecting wires in a grid (not shown). The grey lines show one feasible set of realizations for the wires. Dotted lines enclose the routing channel. The line from \( p \) to \( q \) is a cut of the channel; it has congestion 4 and capacity 4.](image)
When the wires can be legally routed, a simple, “greedy” algorithm suffices to find their minimum-length feasible realizations in time proportional to the size of the output. But to determine whether these realizations exist is even easier; one can test routability in time proportional to $n$, the number of wires.

An instance of the river routing problem can be solved if and only if it satisfies certain easily checked routability conditions. Consider a line segment, or cut, $\overline{pq}$ that runs from the bottom edge of the channel to the top. Suppose that the terminals to the left of $\overline{pq}$ are $A_1$ through $A_i$ and $B_1$ through $B_j$. Then $|j - i|$ different wires have terminals on opposite sides of $\overline{pq}$, and hence must cross $\overline{pq}$. I call the quantity $|j - i|$ the congestion of $\overline{pq}$. On the other hand, the number of wires that can cross $\overline{pq}$ without touching is equal to the horizontal or vertical separation between $p$ and $q$, whichever is larger. I call this quantity the capacity of $\overline{pq}$ and say that the cut $\overline{pq}$ is unsafe [6] if its congestion exceeds its capacity. If any cut in the channel is unsafe, then there is no legal way to route all the wires. Less obvious is the converse: if no cut in the channel is unsafe, then there is a legal way to route all the wires. In fact, as shown in [22], the wires can be routed unless one of $2n$ special cuts is unsafe. Thus to test routability, it suffices to check $2n$ inequalities of the form

$$\text{congestion of } \overline{pq} \leq \text{capacity of } \overline{pq}.$$  

Because the conditions for routability are so simple, one can efficiently solve various placement problems associated with river routing. For example, one can determine how close together the two rows of terminals may be placed while permitting the wires to be routed [8]. If the top row of terminals is free to move relative to the bottom row, then one can find the offset between the two rows that allows the minimum separation between them [34]. Finally, suppose that the terminals on each side of the channel are partitioned into contiguous modules, as in Figure 3, and that each module is free to move horizontally. Then one can position the modules and route the wires so as to minimize the width of the channel [22].

Rough routings

The tractable single-layer routing problems share the property that rough routings of the wires can be determined in advance. To have a rough routing $\rho$ of a wire $\omega$ means that every feasible realization of $\omega$ can be continuously deformed into $\rho$ within the routing region. In mathematical language, every realization of $\omega$ is path-homotopic to $\rho$. When the routing region is simply connected, any two paths between the terminals of $\omega$ are path-homotopic, and hence any such path serves as a rough routing for $\omega$. When the routing region is ring-shaped, rough routings cannot be chosen arbitrarily, but only a few sets of rough routings need consideration. A routing algorithm can simply try each set, and in fact the algorithm of [2] does just
that. In contrast, when the routing region has an arbitrary number of holes, as effectively happens in the NP-complete single-layer routing problems, the number of sets of rough routings that need consideration seems to be exponential.

One is naturally led to consider single-layer routing situations in which rough routings of wires are given. Pinter [41] proposed such a problem, called DRH ("Detailed Routing given a Homotopy"),\(^*\) which involved routing wires in a finite rectilinear grid. An instance of DRH comprises (1) rectangular modules within a bounding box, (2) terminals on the modules’ boundaries, and (3) nonintersecting rough routings that connect pairs of terminals. DRH is a routability problem: it asks whether the given rough routings can be continuously deformed, with their endpoints fixed and without touching any other modules, so that the resulting wires are disjoint paths in the grid.

![Diagram](image)

**Figure 4.** The problem called DRH. Part (i) shows an instance of Pinter’s problem DRH. Solid rectangles are modules, dark points are terminals, and gray curves are rough routings. This instance is routable, because the wires can be realized as shown in (ii). (The grid is not shown.) The dotted line in (i) is a cut. Although the rough routings cross it four times, two of those crossings are not necessary, as they can be removed by deforming the rough routing \(\rho\). Hence the congestion of the cut is 2.

Like the river routing problem, DRH can be analyzed in terms of the congestions and capacities of cuts. One defines a cut to be a line segment whose endpoints lie on modules and whose interior falls in the routing region. The capacity of a cut is the number of wires that can cross the cut without touching; it depends only on the number of gridlines the cut crosses. The congestion of a cut is, in essence, the number of times that wires are forced to cross the cut; it depends upon the topology.

\(^*\) By ‘homotopy’ he meant a set of rough routings, one for each wire to be routed. Technically, the term ‘homotopy’ refers to a continuous deformation of topological maps.
of the rough routings. As before, we say that a cut is unsafe if its congestion exceeds its capacity. Cole and Siegel [6] showed that an instance of DRH is unroutable if and only if it has an unsafe cut.

The characterization of routability has many applications. It was used in [6] to develop a fast algorithm for solving DRH, given a method of computing congestions of cuts. Leiserson and I presented such a method in a subsequent paper [21], thus showing that DRH is solvable in polynomial time. We also set forth routability conditions for a problem very similar to DRH and used them to construct a simplified routability testing algorithm. As in the case of river routing, the routability conditions can be used to solve placement problems as well [29].

Role of this thesis

The fact that DRH is tractable suggests that single-layer routing problems may also be efficiently solvable when rough routings are specified. In fact, Leiserson and I proposed a polynomial-time algorithm for such a problem in our paper [21]. Proving the correctness of our algorithm, however, turned out to be much more difficult than we expected. The problem was fundamental: we had almost no technical tools for working with wires in multiply connected regions. The results in [6] concerning DRH worked only for the grid-based wiring model, and even so, their mathematical foundations were unclear. The additional complexity that arises in continuous wiring models is considerable, as one can see by comparing the papers [52] and [22]. All told, the problem of converting rough routings to detailed routings was poorly understood.

In this dissertation I remedy that situation and show how single-layer routing problems can be efficiently solved. My main technical contribution is a mathematically rigorous theory of single-layer wiring. It gives necessary and sufficient routability conditions for DRH-like problems, and it applies to a variety of common wiring models, gridless models included. In addition, it characterizes the minimum-length feasible realizations of wires, thus shedding light on routing problems as well as routability problems. This theory allows me to justify and generalize the routability testing and routing algorithms given in my earlier paper [21].

The theory of single-layer wire routing has applications to placement problems as well. One placement problem of great practical importance is layout compaction with automatic jog insertion. This problem generalizes the problem of placing modules for river routing in a channel [22], and it can be solved similarly by means of routability conditions. In my master's thesis [29] I presented a polynomial-time algorithm for this problem, but it was restricted to the grid-based wiring model. Using the new theory of single-layer wiring, I extend this algorithm to many other wiring models.
**Figure 5.** An integrated circuit layout. This figure depicts a low-level representation of a portion of an integrated circuit. The layout comprises several layers; each layer is nothing more than a set of polygonal regions. The regions are shaded according to layer, and the shading of upper layers occludes that of lower layers.

**B. Thesis Overview**

This section outlines the structure of the thesis and describes the main ideas behind each chapter. It also provides some practical motivation for this research beyond the theoretical reasons just discussed. Because the problems I study are not easily defined, a precise statement of my main results must wait until Section 1A.

This thesis studies three problems of single-layer wire routing that arise when rough routings of wires are given. They concern an abstraction of a circuit layer called a sketch. The problems are *sketch routability, sketch routing,* and a placement problem: *(one-dimensional) sketch compaction.* I present polynomial-time algorithms for all three. These problems seem nearly as general as they can be and still remain efficiently solvable. On the one hand, they subsume most of the single-
layer routing and placement problems that have previously been proven tractable. On the other hand, natural variations on these problems that are less restrictive are also NP-complete [41, 48], and hence are unlikely to have polynomial-time algorithms.

**Wires as flexible objects**

One motivation for the sketch problems stems from the design of integrated circuits (ICs) and printed circuit boards (PCBs). Figure 5 depicts part of the layout for an integrated circuit: each grey tone corresponds to one layer of the chip. This layout contains no explicit information about the functions performed by different regions on a layer. The designer, on the other hand, considers some areas to be wires and other areas to be device components, as shown in Figure 6. He or she is often willing to let wires change in shape and length, but wants to control the shapes of
active devices and the widths of wires, as these parameters have the greatest effect on the performance of the circuit. This observation suggests that a design system for integrated circuits should distinguish between wires and other components, and should treat wires as flexible connections of fixed width. A sketch is an abstraction of an integrated circuit layer that allows wires to be treated this way.

The type of system I envision would free the designer from concern with the geometry of wires. The designer would provide the system with rough routings for wires, and the system would either route them to form a legal layout, or else show the designer why no routing was possible. When the designer wished to move some of the circuit devices, the system would automatically bend wires and move other components as necessary to keep the layout legal. The problems of sketch routability, routing, and compaction embody the main computational tasks that such a system should be able to execute.*

Nature of this research

Despite its practical roots, this dissertation is in essence a theoretical study. I have not implemented any of my algorithms, nor are the sketch problems themselves designed to model the complex rules that IC designs must obey. Instead, the thesis is primarily concerned with the mathematical foundations of single-layer wire routing. My approach allows me, in the analysis of sketch problems, to trade complexity in the algorithms for complexity in their proofs of correctness. Thus the algorithms I present are relatively simple, but their justification occupies the bulk of this document.

Central to both the algorithms and the theory is the concept of a cut: a path in the routing region that spans two obstacles. As with other single-layer wire-routing problems, the congestions and capacities of the cuts in a sketch determine its routability. This theorem informs the algorithms for the sketch routability and compaction problems; its application to routability testing is evident. Taking the idea farther, I show that a sketch can be compacted by transforming the routability conditions given by cuts into constraints—simple linear inequalities—on the positions of obstacles. Solving the resulting constraint system reveals the optimal locations for the obstacles, including the terminals of wires. The compacted sketch can then be routed to restore the wires.

The sketch routing algorithm requires a deeper result, and a new concept: that

* Such a system has recently been implemented [36, 37]. Called 'Bubbleman', it employs similar ideas to those presented here, but its algorithms are quite different. Rather than solving global routing and routability problems, it incrementally builds a layout with minimal wire lengths as one inputs the components. It also performs two-dimensional compaction with automatic jog insertion via simulated annealing [19].
of a half-cut. Whereas a cut measures the congestion between two obstacles, a half-cut measures the congestion between an obstacle and a wire. Each half-cut for a wire constrains the routing of that wire: if the half-cut becomes too short, the other wires will be unable to fit across it. Certain of these constraints suffice to establish the optimal detailed routing of a sketch—the feasible realization whose wires have minimum length.

To a large degree, then, the study of single-layer wire routing is the study of cuts and half-cuts, and their interactions with wires. The subject has two parts: a mathematical part, which establishes the theorems concerning routability and minimum-length feasible realizations; and an algorithmic part, which concerns the computation of congestion for cuts and half-cuts, and the integration of this information over an entire sketch. Besides the division between algorithms and mathematics, there is another. The sketch compaction problem demands rather different techniques from the sketch routing and routability problems, and so I treat it separately.

Algorithmic ideas

Chapter 1 defines the sketch model and presents efficient algorithms for the sketch routability and routing problems. The idea behind these algorithms is the conversion of topological conditions to geometric conditions. My theory of single-layer wiring reduces the sketch routing and routability problems to two simpler problems:

1. computing the congestion of a cut or half-cut, and
2. finding the shortest routing of a wire that passes through certain line segments in a certain order.

Problem (2) happens to be equivalent to the task of finding the shortest path that passes in order through a sequence of triangles, each one sharing an edge with the preceding one. In this form the problem is evidently geometrical, and can be solved in linear time by a short algorithm. Problem (1) is harder.

To compute the congestion of a cut in a sketch, I use a data structure called the rubber-band equivalent of the sketch. This structure is built by shrinking every wire in the sketch to its minimum length. The shrunken wires, or rubber bands, make no more crossings with cuts than their topology dictates. Leaving aside some technical difficulties, the congestion of a cut is derivable from crossings it makes with rubber bands. The same goes for half-cuts. Thus the rubber-band equivalent expresses a topological quantity, congestion, in terms of a geometric quantity, a crossing number. Computational geometry provides the means to accelerate the computation of these crossing numbers. The construction of the rubber-band equivalent, too, is an essentially geometric process, and fairly efficient. In sum, geometric methods
provide a conceptually uniform approach to sketch routing and routability testing, and lead to efficient algorithms for both problems.

Mathematical ideas

Chapter 2 begins a long technical development that culminates in correctness proofs for my sketch routing and routability testing algorithms. (Actually, some of the final steps are unfinished.) The theory revolves around a single concept: that of a simply connected covering space for the routing region. The covering space is a surface with infinitely many layers, each built from pieces of the routing region. The pieces are sewn together in such a way that every loop in the surface can be shrunk to a point. Paths in the covering space can be projected down to the routing region, and paths in the routing region can be lifted up into the covering space. Nearly every aspect of the theory exploits the special relationship between the multiply connected routing region and its simply connected covering space.

The covering space serves two primary functions. First, it provides a good definition of a necessary crossing between a cut and a wire. Informally, a necessary crossing is one that cannot be removed by rerouting the wire. The formal definition helps me to analyze the congestion of cuts, and to rigorously derive inequalities among the congestions of different cuts. Second, the covering space sorts out the interactions of different parts of the same wire. When a wire is lifted to the covering space, homotopically distinct parts of the wire fall on different layers. Thus the covering space transforms a problem with homotopy constraints (the rough routings) into a purely spatial problem. To show that a wire can be routed, I first find an appropriate routing within the covering space, and then project it to the routing region.

A second model

Unfortunately, the sketch model lends itself poorly to topological analysis: the covering space of the routing region does not permit lifting of wires and cuts. So my mathematical development employs a more elegant, but less practical, model, in which the analogue of a sketch is called a design. The design model supports a rich theory of routing that relates properties of cuts to the existence of various types of routings. It also identifies and characterizes the optimal, or ideal, routings of a routable design, and provides methods for computing the congestions of cuts and half-cuts. The design model differs sufficiently from the sketch model, however, that results in one cannot be applied directly to the other. Instead one must derive results concerning sketches by approximating the sketch with designs that, in some sense, converge to it.
I spend Chapters 4 through 7 exploring designs, Chapters 2 and 3 preparing for this exploration, and Chapter 8 applying the results to the sketch model. The chapter-by-chapter breakdown is as follows.

- Chapter 2 begins by stating many of the mathematical definitions and notations that will be used throughout the technical parts of the thesis. It also supplies a short introduction to homotopy theory and covering spaces, enough to appreciate the elementary ways in which I employ them. The rest of Chapter 2 claims, mostly without proof, theorems from topology that will be used sporadically in the following chapters.

- Chapter 3 defines the class of spaces, called sheets, that serve as the routing regions for designs. It then studies the topology of their simply connected covering spaces, which I call blankets, and of various sorts of paths in sheets and blankets. The central result is that when a link (e.g., a cut or wire) is lifted to a blanket, it separates the blanket into two pieces, a left side and a right side. This result allows us to recapture some of the simplicity of river routing in channels, where every cut and wire divides the channel. Thus Chapter 3 lays the real foundation for what is to come.

- Chapter 4 defines the design model and begins to develop the theory of cuts, half-cuts, and wires. It relates the congestion of a cut to the necessary crossings of the cut by wires, and it identifies congestion with a quantity called flow defined in terms of liftings to a blanket. Flow is a much more convenient and powerful concept than congestion, and much of Chapter 4 is concerned with relating the flows across different cuts. Not all the cuts we consider are straight; some even have self-intersections.

- Chapter 5 defines the ideal routings of the wires in a safe design, and proves that they form a valid routing of the design. The safety of a design is a function of its straight cuts, and primarily of the flows and capacities of those cuts. The result of the construction is that every safe design is routable. In more abstract terms, local routability implies global routability.

- Chapter 6 completes the proofs of the major theorems concerning designs. First it shows that unsafe designs are unroutable, providing a converse to the result of Chapter 5. It also shows that the arc length of ideal routings cannot be improved upon. Finally, it proves that the routability of a design depends only on the properties of a few straight cuts, not all of them. This observation makes it effective to test routability by testing safety.

- Chapter 7 goes on to consider techniques for routing and testing the routability of designs. The design model is ill-suited for the development of routing algorithms, but the techniques developed with reference to designs can later be applied to sketches. Much of Chapter 7 revolves around the
use of rubber bands in routing and testing routability. It explains how to construct them, why they can be used to compute flow, and how they give rise to the structures (called corridors or tunnels) that we using in routing wires.

- Chapter 8 develops a careful correspondence between sketches and designs. It then shows how to use results in the design model designs to obtain results in the sketch model. Due to lack of time and space, some proofs are omitted. The outcome includes two major theorems concerning sketches, and also justifies my main algorithms for routing testing the routability of sketches.

I make no claims about the simplicity, shortness, or elegance of my proof techniques. Indeed, they could surely be improved, particularly if more advanced results in algebraic topology were assumed. They testify nonetheless that the tool of lifting to a simply connected covering space is apposite to wire routing with homotopy constraints. As long as my current proofs are, they might be even longer in another approach. For any approach must ultimately be based on a solid understanding of the role of homotopy in the wiring problem, such as I have tried to give in Chapters 3 and 4.

Compaction with flexible wires

Chapter 9 presents and proves correct a polynomial-time algorithm for sketch compaction. The algorithm requires a new approach to the manipulation of cuts because the geometry of a sketch can change radically during compaction. The topology of the sketch, on the other hand, is invariant. For this reason I introduce a second technique for computing congestions: a graph-theoretic method that works directly from the topology of the sketch. It includes an interesting preprocessing phase that speeds up searches through the graph that represents the sketch.

The chief difficulty in sketch compaction, however, is not in computing the congestions of cuts, but in deciding which cuts require consideration. As the obstacles in a sketch move, the relevant cuts change, as do their congestions. What the compaction algorithm actually examines is a set of potential cuts—cuts whose positions are functions of the configuration of the obstacles—that give rise to routability conditions. It turns out that by considering the potential cuts in a certain order, one can find for each potential cut the configurations in which it constrains the layout; the potential cut has the same congestion in all such configurations. So the compaction algorithm builds its constraint system iteratively, at each step considering the effects of adding a single potential cut. The algorithm itself is far from transparent; only in the analysis does its rationale become clear.

The analysis of the compaction algorithm leans heavily on the notion of a configuration space. In sketch compaction the configuration space is the vector space of
possible displacements of the obstacles.* I relate the compaction algorithm to an abstract algorithm that manipulates subsets of the configuration space. (The actual, or concrete, algorithm represents these subsets by systems of linear inequalities.) I deduce the correctness of the abstract algorithm from four postulates concerning the sequence of potential cuts it evaluates. These postulates indeed hold for the potential cuts used by the compaction algorithm, and hence the correctness of that algorithm quickly follows. The advantage of this proof strategy is that changes in the model need not entail major changes in the proof; rather, it is enough to choose a sequence of potential cuts and check that they satisfy the postulates in the particular model one wishes to use.

Extensions and discussion

Chapter 10, the final chapter before the Conclusion, explores how far and how easily the sketch model can be extended in various ways. Among the possibilities it considers are these: allowing wires and obstacles to be made of different materials, each pair of materials with its own separation requirement; forcing wires to run in a grid; measuring the separation between wires with the euclidean metric; allowing wires to contain circular arcs; allowing obstacles to contain circular arcs; permitting the terminals of a wire to merge or pass through one another during compaction; routing with extended terminals, letting the points of connection move; and including wires with more than two terminals. In most cases the proposed changes in the sketch algorithms are relatively minor, but to justify them may be difficult (or even impossible, if one of my conjectures is false). The greatest problems arise in attempting to handle extended terminals and multiterminal nets. Chapter 10 proposes an alternative to the sketch model that, if it proves mathematically tractable, could eliminate these and several other problems.

The thesis concludes with a summary of its results, a comparison with some related work, a list of open problems, and several suggestions of directions for future research.

* The configuration space seems to be a natural tool for understanding compaction. It was by formulating the compaction problem in terms of configurations that I discovered a fact that could have significant practical consequences for compaction algorithms [30]. My observation, explained in Chapter 9, implies that Dijkstra's algorithm can be used to solve the standard one-dimensional compaction problem if the initial layout is legal.
Chapter 1

Sketch Algorithms

This chapter states precisely the major results of this dissertation. First it defines the sketch model and the problems of sketch routability, sketch routing, and sketch compaction that my algorithms solve. It then considers a data structure for a sketch, called its rubber-band equivalent, which supports computation involving the sketch topology, and thereby speeds up the algorithms for sketch routability and sketch routing. Next it presents algorithms for those two problems. Both algorithms have worst-case running time $O(n^2 \log n)$ on input of size $n$. Then, in Sections 1E and 1F, I show that the performance of these algorithms is limited mainly by the routability testing procedure, and I present several methods for improving its average-case running time. By exploiting the idea of shadowing [6] we obtain an algorithm for sketch routability that runs in time $O(n^{3/2} \log n)$ on the average. All proofs are deferred to Chapter 8. Sections 1A through 1D represent joint work with Charles Leiserson.

1A. The Sketch Model

We begin by defining sketches and the natural problems set in that model. This section also states the sketch routability theorem and the sketch routing theorem in which my algorithms are grounded.

A sketch is an abstraction of the wiring on a single layer of an integrated circuit or printed circuit board. It represents the topology and the geometry of that layer, but none of its electrical or functional characteristics. I have chosen the sketch model for its simplicity, its similarity to existing theoretical models, and its ease of implementation. Consequently, it deals only with piecewise linear objects. This limitation is not serious, for in practice one often approximates curved structures by polygons in order to avoid the problems of computing with irrational numbers (in whatever representation). A more serious drawback is that sketches, as described here, cannot satisfactorily represent wires with more than two terminals. These issues are discussed further in Chapter 10.
Since a sketch must distinguish between flexible and rigid objects, it has two types of components: traces, which represent either rough routings or detailed routings of wires, and features, which represent terminals, devices, and routing obstacles. A feature is a point or line segment in the plane, and a trace is a piecewise linear path with the following properties.

1. The path has no self-intersections.
2. The path touches no features except at its endpoints.
3. The endpoints of the path are features— the terminals of the trace.
4. Each terminal is a point, isolated from the other features.

A sketch is a finite set of features that intersect only at their endpoints, together with a finite set of nonintersecting traces. The connected groups of features in a sketch are called islands. By (3), every terminal is an island; the islands that are not terminals are called obstacles. The routing region of a sketch is the set of points that lie on no feature. Islands and traces are collectively called elements.

![Figure 1a-1. A typical sketch and the territories of its elements. Part (a) illustrates a simple sketch. Dark line segments are features, light paths are traces, and the number nearest to each element indicates its width. Part (b) shows each element's territory, which takes its width into account. The territory of a trace is not shown where it overlaps the territories of its terminals.]

The elements of a sketch actually represent the centerlines of regions in the wiring layer. Hence we associate with each element a positive number called its width that indicates how much space it actually requires. No trace may have greater width than either of its terminals. The territory of an element of width \( d \) is the set of points whose distance from that element is less than \( d/2 \).

We measure distance using a piecewise linear norm,* denoted \( \| \cdot \| \), that is the same for all elements. I call this norm the wiring norm, because different norms

---

* See the glossary for an explanation of wiring norms. The examples in this chapter
give rise to different wiring models. Terms like 'distance' and 'closest'—but not 'arc length'—refer to measurement in the wiring norm unless otherwise specified. The distance in the wiring norm between two points $p$ and $q$ is $\|p - q\|$, and the distance between two regions $P$ and $Q$ is

$$\|P - Q\| = \inf_{p \in P} \inf_{q \in Q} \|p - q\|.$$ 

Depending on the placement of its elements, a sketch may or may not represent a valid circuit layout. If it does, the sketch is called proper. In my model a sketch is proper if the elements that should not interact are properly separated. Two elements are assumed to interact if and only if their territories overlap. Sometimes this interaction is good, as when a trace connects to its terminals. Thus we consider a sketch to be improper if it has two elements with overlapping territories, unless those elements are a trace and one of its terminals.

There is one further constraint on proper sketches. It arises because a trace must be separated from itself, lest it form a loop in the layout. Let us say that a trace is self-avoiding if the set of points lying outside its territory and outside the territories of its terminals has only one connected component that includes islands of the sketch. In other words, the territory of a self-avoiding trace, together with those of its terminals, does not separate any two islands from one another. All the traces in a proper sketch must be self-avoiding. The sketch in Figure 1a-1 fails to be proper because the trace $\omega$ is not self-avoiding.

**Sketch routing problems**

The single-layer routing problems I consider take a sketch as input. This sketch is not expected to be proper. Instead, each trace in the input sketch represents a rough routing; it defines a set of possible realizations for that trace. A realization of a sketch is a sketch that results from routing each of its traces, that is, replacing them by realizations. We say that a sketch is routable if it has a proper realization. The sketch routability problem, then, is just the problem of determining whether a sketch is routable. It turns out that whenever a sketch is routable, it has a proper realization that simultaneously minimizes the length of every trace. The sketch routing problem is to find this realization if it exists.

To route a trace in a sketch, one deforms the trace in a continuous fashion. The notion of continuous deformation is made precise as follows. We define a bridge to be a piecewise linear path in a sketch that intersects features of the sketch at its endpoints only. Then all traces are bridges. We think of a bridge as a continuous use the $L^\infty$ norm, in which the distance between two points is the maximum of their horizontal and vertical separation.
function from the unit interval $I = [0, 1]$ to the plane $\mathbb{R}^2$. Two traces of the same width, say $\theta_0$ and $\theta_1$, are bridge-homotopic if they are part of some family of bridges $\{ \theta_t : t \in I \}$ such that the function $T: I \times I \to \mathbb{R}^2$ defined by $T(s, t) = \theta_t(s)$ is continuous and piecewise linear. The function $T$ is a homotopy or "continuous deformation" of bridges. If $\theta$ is a trace in a sketch $S$, then a route for $\theta$ is any bridge that is bridge-homotopic to $\theta$ in $S$. A realization of $\theta$ is a trace that is a route for $\theta$. The realization is feasible if it is part of a proper realization of the sketch $S$.

![Figure 1a-2. A proper realization of a sketch and a compacted version of it.](image)

Part (a) is a proper realization of the sketch in Figure 1a-1: the territories of its elements are disjoint, except where traces contact their terminals; and every trace is self-avoiding. (Territories are open sets; they do not include their boundaries.) Every trace in this realization has minimum length. Part (b) shows a compacted version of this sketch. If we allow the islands to move sideways independently, then among all the proper sketches that are reachable from the configuration at left, the sketch in (b) has minimum width.

The sketch compaction problem is a generalization of the sketch routing problem that involves moving features as well as traces. The input to this problem is a routable sketch with the islands grouped into modules; each module is allowed to move horizontally as a unit. Modules may not move vertically. As modules move, traces must move as well in order to remain connected to their terminals. Let us say that a sketch is reachable if it can be obtained from the input sketch by a continuous, piecewise linear motion that maintains the routability of the sketch. (The motion of each trace should be a piecewise linear homotopy, though not one that necessarily fixes its endpoints.) The sketch compaction problem is to find a proper, reachable sketch of minimum width. Solving this problem allows one to perform one-dimensional compaction of VLSI layouts, inserting jogs into wires automatically; a special case of this problem was considered in [29].
Major results

This thesis presents polynomial-time algorithms for the sketch routability, routing, and compaction problems. Given as input a sketch of size \( n \), the routing and routability testing algorithms run in time \( O(n^2 \log n) \), while the sketch compaction algorithm runs in time \( O(n^4) \). All are fairly easy to implement, and are efficient enough to be useful in practice.

The correctness of these algorithms rests on the theory of single-layer wiring. This theory gives necessary and sufficient conditions for a sketch to be routable, and provides methods for testing these conditions. For routable sketches, it also characterizes the minimum-length feasible realizations of traces. The tools of the theory are the techniques of point-set and algebraic topology; the objects it studies are traces and cuts.

One important result of the theory says, in essence, that a sketch is unroutable if and only if too many traces are forced to pass through the "channel" between some pair of islands. This statement may seem obvious, but it is far from trivial. We formalize it using the idea of a cut. A line segment is a cut of a sketch if it touches the features of the sketch at its endpoints only. (More properly, a cut is a linear path, and we write the cut \( \overline{pq} \) as \( p \to q \) if we wish to emphasize its orientation from \( p \) to \( q \).) Each cut has a capacity that represents the maximum total width of the traces that can cross it. If endpoints of the cut \( \overline{pq} \) lie on the islands \( P \) and \( Q \), then we define

\[
\text{capacity of } \overline{pq} = \text{length of } \overline{pq} - \left( \text{width of } P \right)/2 - \left( \text{width of } Q \right)/2.
\]

The length of \( \overline{pq} \) is measured in the norm used to define territories.

Each cut also has a congestion that measures the total width of the traces forced to pass across it. To define it, we first define the entanglement of a trace with a cut \( \overline{pq} \) to be the minimum number of crossings of \( \overline{pq} \) by any route for the trace. Crossings that occur at \( p \) or \( q \) do not count. The entanglement of a trace with a cut represents, in some sense, the number of necessary crossings of the cut by the trace. Intuitively, a necessary crossing is one that cannot be removed by applying a bridge homotopy to the trace. This intuitive notion is not easy to formalize, however, so we leave it informal until Section 4B. Congestion is defined in terms of entanglement. If \( \Theta \) denotes the set of traces in the sketch, then we define

\[
\text{congestion of } \overline{pq} = \sum_{\theta \in \Theta} (\text{width of } \theta) \cdot (\text{entanglement of } \theta \text{ with } \overline{pq}).
\]

If the congestion of a cut exceeds its capacity, then the traces will not be able to fit across the cut. We say a cut is unsafe if its congestion exceeds its capacity. This does not always mean the sketch is unroutable, however, because there may
The Rubber-Band Equivalent of a Sketch

Figure 1a-3. The attributes of cuts. In the sketch depicted here, the dashed line \( \overline{pq} \) is a nonempty cut. Three traces intersect this cut. The trace \( \alpha \) has width 1 but entanglement 0 with \( \overline{pq} \), as the crossings it makes with \( \overline{pq} \) are unnecessary. The trace \( \beta \) has width 2 and entanglement 2, and the trace \( \gamma \) has width 3 and entanglement 1. Hence the congestion of the cut \( \overline{pq} \) is 7. If its capacity is 7 or greater, this cut is safe.

not be any traces crossing that cut. A cut is empty if it has zero congestion and its endpoints lie on the same island. Empty cuts have no bearing on routability. A nonempty, unsafe cut, on the other hand, means that the channel it spans is congested. A sketch is safe if and only if all its nonempty cuts are safe.

Now the theorem concerning routability can be stated more precisely. If a sketch is safe, then it is routable. Conversely, every routable sketch is safe. I call this result the sketch routability theorem. It suggests that the routability of a sketch may be checked by testing whether certain cuts of the sketch are safe. One can easily find a small set of critical cuts with the property that if any nonempty cut is unsafe, some nonempty critical cut is unsafe. My algorithm for the sketch routability problem works by testing the safety and emptiness of these critical cuts.

More significant than the sketch routability theorem, however, is the sketch routing theorem, which yields minimum-length feasible realizations for the traces in a routable sketch. This theorem cannot be fully stated here, because it depends upon a complicated construction of traces called ideal realizations. Intuitively, an ideal realization of a trace in a routable sketch is a minimum-length route for that trace that stays far enough away from the islands to permit the other traces to be routed. Every trace in a routable sketch has a unique ideal realization. The sketch routability theorem states two things. First, if every trace in a routable sketch is replaced by its ideal realization, then the resulting sketch is proper. Second, no shorter feasible realizations exist for those traces. To solve the sketch routing problem, therefore, one need only be able to compute the ideal realization of each trace in a routable sketch. My routing algorithm does just this.

1B. The Rubber-Band Equivalent of a Sketch

My algorithms for sketch routability and routing both rely on a data structure called the rubber-band equivalent (RBE) of the sketch. This structure solves the central difficulty associated with the processing of sketches, namely the integration
of geometric and topological information. Methods from computational geometry can be applied to the RBE to compute the congestions of cuts and to find constraints on the positioning of traces. In this section I define the RBE, show how to construct it, and explain the operations that it supports.

I assume the input sketch is represented as a pair of data structures: a set $F$ of features, and a set $T$ of traces. Let us denote the size of a data structure $D$ by the symbol $|D|$. Each feature is a point or line segment, and hence requires constant space to represent. Each trace, being piecewise linear, is represented as a sequence of line segments. Thus $|F|$ is proportional to the number of features in $F$, and $|T|$ is proportional to the number of line segments that compose the traces in $T$. If $S$ is the sketch $(F, T)$, then we have $|S| = |F| + |T|$. The algorithm given in this section computes the rubber-band equivalent of a sketch $S = (F, T)$ in time $O(|F||T|\log |S|)$.

Motivation

Intuition suggests that if a trace crosses a cut more times than necessary, then it contains an unnecessary detour. If we could make each trace as short as possible, then the number of crossings between a cut and a trace would equal their entanglement. Unfortunately, most traces have no minimum-length routes, for a trace is not permitted to contact any features but its terminals. So we construct instead the rubber band of each trace: the shortest path, in euclidean arc length, that is the limit of a sequence of routes for that trace. Intuitively, we shrink the trace to its minimum length, allowing it to touch features but not to cross over them. The resulting path is a sequence of line segments whose endpoints are feature endpoints.

If we replace every trace in a sketch by its rubber band, the result is not, in general, a sketch. It nevertheless can be treated as a sketch in which features and traces have infinitesimal separation. Wherever a rubber band touches a feature, we consider it to leave the feature to its left, leave the feature to its right, or else connect to the feature (if the feature is one of its terminals). Similarly, wherever one rubber band touches another, the second rubber band falls either left or right of the first. No rubber band ever crosses over another one, and hence this adjacency information can be assigned in a consistent manner to all the features and rubber bands. The RBE of the sketch stores this information in a concise form.

The RBE helps one to compute, for any desired straight cut, the sequence of traces that necessarily cross it, in order along the cut. This sequence is called the content of the cut. (It may contain the same trace more than once.) The content of a cut nearly equals the sequence of rubber bands that cross the cut, the difference being that one sequence consists of traces while in the other one consists of the corresponding rubber bands. The tricky part, of course, is defining
Figure 1b-1. *The rubber-band equivalent of a sketch.* The sketch is on the left, its RBE on the right. In the RBE, features and rubber bands that are shown here as adjacent segments actually overlap. The strands have been artificially displaced to show the adjacency relations among the features and rubber bands.

which rubber bands cross a cut and in what order they do so. Here the adjacency information comes in. Some places where the cut intersects a rubber band should not be considered crossings. For example, if the cut intersects a feature from the top, and the rubber band runs along the bottom of the feature, one should think of them as being separated by an infinitesimal distance. If one filters out such intersections, the remaining ones correspond exactly with the traces in the cut’s content. Moreover, the cut can be considered to cross the rubber bands in a certain order, because even where the rubber bands overlap, their adjacency relation orders them totally. This ordering is irrelevant for computing flow but highly significant for wire routing, as explained in Section 1D.

**Definition and use of the RBE**

The RBE of a sketch is essentially a planar multigraph with some extra structure. Its nodes are feature endpoints; its arcs are features and cables, which are groups of rubber band segments. For each pair \{p, q\} of feature endpoints there can be up to three cables from p to q: one on each side of the cut or feature \(\overline{pq}\), and one that crosses over the cut \(\overline{pq}\). The rubber band segments within each cable are called its *strands*, and are totally ordered. We represent the ordering by means of a height-balanced tree. In addition, the features and cables radiating from each feature endpoint are circularly ordered as shown in Figures 1b-2 and 1b-3. We store this ordering in a pair of height-balanced trees by breaking it into total orderings as explained later. In effect, these orderings specify which features and strands would be adjacent if the rubber bands had infinitesimal thickness. The total orderings within cables, combined with the circular ordering on the cables that touch a feature endpoint, give rise to a circular ordering on the *strands* that touch a feature.
The RBE supports the following operation: given a ray emanating from the feature point $p$, report its crossing sequence at $p$: the sequence of rubber bands that cross over that ray at $p$. Rubber bands that end at $p$ are not part of the crossing sequence, nor are rubber bands that are parallel to the ray at $p$. The content of a cut $p \triangleright q$, converted into a sequence of rubber bands, is then the concatenation of three lists.

1. First is the crossing sequence of the ray $\overline{p\tilde{q}}$ at $p$.
2. Next come the rubber bands whose strands cross of the middle of $p \triangleright q$. The strands are sorted by distance from $p$, and ordered within each cable as well.
3. Last is the reverse of the crossing sequence of $\overline{q\tilde{p}}$ at $q$.

If $p$ is a point on a feature but not a feature endpoint, then the crossing sequence of a ray at $p$ can be computed without new data structures. Let $\overline{qr}$ be the feature containing $p$. The crossing sequence of a ray $\overline{ps}$ is just the sequence of strands in the cable (if any) lying along $\overline{qr}$ on the same side as $s$. This cable can be found by examining the circular order at $q$ or $r$, because if it exists, it must be adjacent to the feature $\overline{qr}$.

To compute crossing sequences at a feature endpoint $p$ we use one of two different data structures. If all the cables touching $p$ fall on a line $\ell$, as shown in Figure 1b-2, then it suffices to store four different crossing lists: two for the rays lying in $\ell$, and two for rays pointing into the half-planes of $\ell$. In this case, at most six arcs (features and cables) connect to the node $p$, so their circular ordering can be represented by
a constant-size data structure. If the cables touching $p$ are not all parallel, on the other hand, then we have the situation of Figure 1b-3. There is a ray $\overrightarrow{qr}$ whose crossing sequence at $p$ is longest, and a ray $\overrightarrow{qs}$ whose crossing sequence at $p$ is empty. Moreover, the crossing sequence at $p$ of an intermediate ray $\overrightarrow{pq}$ is obtained by enumerating the strands in the cables interior the angle $\angle spq$, where the interior of this angle is chosen not to include the ray $\overrightarrow{qr}$. To enumerate these cables quickly, we break the circular ordering of the arcs incident on $p$ into two total orderings. The arcs between $\overrightarrow{qr}$ and $\overrightarrow{qs}$ clockwise are stored in one height-balanced tree, and the arcs between $\overrightarrow{qr}$ and $\overrightarrow{qs}$ counterclockwise are stored in another height-balanced tree.

Constructing the RBE

The rubber-band equivalent of a sketch can be computed fairly efficiently. First one triangulates the routing region with cuts, which I call doorways or simply doors. This operation is efficient—it requires only $O(|F| \log |F|)$ time—and fairly standard in computational geometry [43], so I shall not dwell on it. Next one constructs the planar graph whose nodes are feature endpoints and whose arcs are feature segments. Nodes can be represented initially without using height-balanced trees, but when cables are added to the graph, some nodes will have to be converted to the more general data structure. Then comes the interesting part: for each trace in the sketch one computes its rubber band and inserts it into the data structure. I describe the insertion operation first.

Given the rubber band of a trace, one inserts its strands in order. To insert the strand $\overrightarrow{qr}$, first determine which cable it belongs in. There can be up to three cables from $q$ to $r$: one to the left of the ray $\overrightarrow{qr}$, one in the middle, which crosses over $\overrightarrow{qr}$; and one to the right of $\overrightarrow{qr}$. If the new strand leaves $q$ and $r$ to different sides, it goes in the middle cable. Otherwise if it leaves either $q$ or $r$ to the left or right, it goes in the left-hand or right-hand cable, respectively. If this strand is the entire rubber band, so that it has $q$ and $r$ as terminals, it goes in the left-hand cable by default. If the appropriate cable for the strand $\overrightarrow{qr}$ does not exist, create it and insert it into the circular orders at $q$ and $r$.

The case in which the cable exists is more difficult. If $\overrightarrow{qr}$ is the first strand in its rubber-band (i.e., $q$ is its terminal), then insert it at the right-hand edge of the left-hand cable or the left-hand edge of the right-hand cable, as appropriate. Otherwise let $\overrightarrow{pq}$ be the strand preceding $\overrightarrow{qr}$ in its rubber band, and find which strands are adjacent to $\overrightarrow{pq}$ in the circular order at $q$, ignoring those that connect to $q$ as a terminal. One of these is connected to a strand $X$ that goes to $r$. Insert $\overrightarrow{qr}$ adjacent to $X$. 

- 37 -
Making rubber bands

To find the rubber band for a trace $\theta$, we follow $\theta$ through the triangulation, and record the sequence of doorways that $\theta$ passes through. When $\theta$ crosses a doorway $\overline{pq}$ but immediately returns, the doorway $\overline{pq}$ may be removed from the sequence, because it represents an unnecessary detour. After eliminating such unnecessary doorways, which one can do in linear time, one is left with the sequence of doorways that the rubber band for $\theta$ passes through. Let us call this sequence of doorways a corridor. The shortest path through this corridor that connects the terminals of $\theta$ is the rubber band for $\theta$.

I now outline a linear-time algorithm to find the shortest path through a corridor. Each door in a corridor may share an endpoint with the previous door (or with the first terminal of the wire, if this is the first door), and hence has either one or two new vertices. We represent a corridor as the sequence of new vertices, together with an indication of which vertices lie to the left of the path, and which lie to the right. The algorithm examines the vertices one by one, keeping track of left and right boundaries for the shortest path. Suppose that a new vertex of the $n$th doorway, call it $l$, lies to the left of the path, and let $t$ denote the initial terminal of the trace. After examining $l$, the left boundary is the shortest path through the first $n-1$ doorways from $t$ to $l$. Similarly, after examining a right vertex $r$, the right boundary is the shortest path in the corridor from $t$ to $r$. The boundaries are piecewise linear paths, stored as sequences of vertices.

Figure 1b-4. A snapshot of Algorithm W. Only active vertices are shown. Dotted lines are the doors of the corridor, dark lines are boundaries for the optimal path, and light lines are rays of visibility. If $e$ is a left vertex, the algorithm will remove the points $c$ and $d$ from the left boundary in favor of $e$. If $e$ is a right vertex, it will replace the right boundary from $a$ to $x$ by the segments $ab$ and $be$.

Simple visibility tests are used to maintain the boundaries. The vertex at which the left and right boundaries diverge, and all vertices following it, are called active. When a left vertex $l$ is encountered, the algorithm finds the oldest active left vertex $v$ whose line of sight to $l$ falls right of the left boundary. Next, it removes the portion of the left boundary that follows $v$. If the right boundary blocks $v$ from seeing $l$, then the left boundary is extended along the right boundary until $l$ is visible from
the end of the left boundary. Finally, the point \( l \) is added to the left boundary. Symmetrical actions occur upon examination of a right vertex.

Algorithm \( W \), shown below, is a linear-time implementation of this path-finding procedure. It uses stacks to represent the boundaries, and employs two simple geometric tests to maintain them. The function \( \text{R-TURN}(p, q, r) \) determines whether the point \( r \) lies to the right of the ray \( \overrightarrow{pq} \); similarly, \( \text{L-TURN}(p, q, r) \) is true when \( r \) lies to the left of \( \overrightarrow{pq} \). The algorithm assumes that consecutive doors are not collinear, and that the corridor contains the final terminal of the trace as both left and right vertices.

**Algorithm \( W \).** (Finds a minimum-length path through a corridor.)

Input: Corridor vertices \( C[1..n] \); initial terminal \( t \).

Local variables: arrays of points \( L[1..n] \) and \( R[1..n] \); integers \( b, i, l, \) and \( r \).

Output: the vertices \( L[1..l] \) of a piecewise linear path.

1. \( l, r, b \leftarrow 1; \; L[l], R[r] \leftarrow t; \)
2. for \( i \leftarrow 1 \) to \( n \) do
3. \( \text{if } C[i] \text{ is a left vertex then} \)
4. \( \text{while } l > b \text{ and } \text{R-TURN}(L[l - 1], L[l], C[i]) \)
5. \( \text{do } l \leftarrow l - 1; \)
6. \( \text{while } r > b \text{ and not } \text{L-TURN}(R[b], R[b + 1], C[i]) \)
7. \( \text{do } b \leftarrow b + 1; \; l \leftarrow l + 1; \; L[b] \leftarrow R[b]; \)
8. \( l \leftarrow l + 1; \; L[l] \leftarrow C[i] \)
9. \( \text{else (copy lines 4–8, exchanging } L, l, \text{ and } L\text{-TURN for } R, r, \text{ and } R\text{-TURN).} \)

One can extend Algorithm \( W \) to determine, for each feature endpoint that the output path passes over, on which side of the path it lies. Proving the correctness of Algorithm \( W \) is straightforward.

**Complexity analysis**

The time and space performance of the RBE construction are dominated by the processing of strands. Each trace segment passes through \( O(|F|) \) triangles, and hence gives rise to \( O(|F|) \) strands. Hence the number of strands in the RBE is \( O(|F| |T|) \), and this bound is tight in the worst case. In practice the number should ordinarily be much smaller. Algorithm \( W \) generates each strand in \( O(1) \) time, and a strand can be inserted into the RBE in time \( O(\log |S|) \). (The log factor derives from the use of height-balanced trees.) Therefore the construction of the RBE requires time \( O(|F| |T| \log |S|) \) and space proportional to the size of the output, namely \( O(|F| |T|) \).

Given a feature and a ray beginning on that feature, the RBE can produce the crossing sequence of that ray in time proportional to its length. This performance is
optimal for the purposes of my routing algorithm, but not for my routability testing algorithm. To find the congestion of a cut $pq$ one need not compute its content or the necessary crossings of that cut by traces. One need only compute the sum of the widths of the traces in the content of $pq$. (Traces that appear more than once in the content are counted according to multiplicity.) Hence for the purpose of routability testing a condensed form of the RBE is needed. In this data structure, the strands within each cable are not distinguished; instead each cable is assigned a width that represents the sum of the widths of its strands. The condensed RBE also stores the width of every possible crossing sequence a ray could have; this requires storing 2 numbers per feature segment and at most $2n$ numbers at each vertex of degree $n$. These values can be computed in linear time from the widths of the cables, and the correct value for a ray can be found in $O(\log |S|)$ time.

Thus the condensed RBE is a planar multigraph whose vertices are feature endpoints and whose edges are cables. In this graph, at most three edges connect each pair of vertices—three cables, or one feature and two cables. Since the number of edges in a planar graph is at most linear in the number of vertices, the condensed RBE uses only $O(|F|)$ space. The workspace needed for its construction is also $O(|F|)$.

1C. Testing the Routability of a Sketch

A corollary to the sketch routability theorem shows that a sketch is routable if and only if its nonempty critical cuts are safe. We say that the critical cuts are decisive, because their safety and emptiness decide the routability of the sketch. A critical cut is a cut that begins at a feature endpoint and travels to the closest point on another feature. The distance is measured in the wiring norm; ties are broken using the euclidean metric. For any reasonable wiring norm, the critical cuts can be easily identified; there are $O(|F|^2)$ of them.

Thus the problem of routability testing is reduced to the problem of checking the safety and emptiness of a cut. For each critical cut, we need to know its congestion, its capacity, and whether its endpoints lie on the same island. The last condition is easy to test, because the islands of a sketch can be determined in $O(|F|^2)$ time. The capacity of a cut is also easy to compute, for it depends only on the distance between the cut’s endpoints. I assume that the wiring norm of a vector can be computed in constant time. The congestion, on the other hand, is relatively hard to compute; for this we use the rubber-band equivalent of the sketch. This section presents an algorithm to test the routability of a sketch in time $O(|F|^2 \log |F|)$, given its condensed RBE.
The scanning technique

To check each cut quickly, we use an idea from computational geometry called scanning. This technique involves sweeping a scan line across the plane, while keeping track of the objects that intersect the line. The data structure representing those objects can then facilitate the computation of geometric quantities such as the congestion of a cut. If that data structure can be updated and queried quickly, it speeds up the algorithm by eliminating repetitive access to the objects being examined. An event list drives the scanning process by specifying the order in which objects enter and leave the data structure, and when the structure should be queried. The algorithm constructs the event list before scanning, and simulates the motion of the scan line by processing the events in order.

![Figure 1c-1. A snapshot of Algorithm T. Here the algorithm is shown checking the safety of the critical cuts that begin at p. Algorithm T simulates the motion of a ray that sweeps around p like a radar beam. It uses data structures that support fast insertion, deletion, and search to represent the features and cables intersecting the scan ray. Whenever the scan ray includes a possible critical cut, Algorithm T can quickly determine whether this line segment is a cut, and if so, quickly compute its congestion.](image)

For testing routability, the scan line will be a ray emanating from a feature endpoint p. As the ray sweeps through the RBE, it will occasionally intersect another point q such that pq is a critical cut. When this happens, the algorithm computes the congestion and capacity of pq. If the cut is nonempty and the congestion is greater than the capacity, then the sketch is unroutable. The congestion is the sum of three quantities. The first is the total width of the cable strands that intersect the scan ray strictly between p and q; it is obtained from the data structure representing the scan ray. The other two are the total widths of the crossing sequences of the rays pq at p and qp at q; they are provided by the RBE. To construct the event list for scanning, the algorithm sorts all the relevant points in the RBE by angle as seen from p. Objects that touch p are left out. Both the sorting and the data structure accesses require only $O(\log |S|)$ time per object. Since there are $O(|F|)$ feature endpoints to scan from, the total execution time is $O(|F|^2 \log |S|)$. 

- 41 -
Details of Algorithm T

The condensed rubber-band equivalent of a sketch consists of two types of line segments: features and cables. To store the objects crossing the scan ray, therefore, our algorithm uses two dynamic sets, FS and WS. The FS data structure contains features, while WS is a set of cables, each of which is weighted according to the total width of the strands in the cable. The operations on FS are insertion, deletion, and

- \text{MIN}(s)$, which returns \textit{true} if $s$ is the nearest segment in FS to the origin $p$, and otherwise \textit{false}.

If the scan ray intersects $s$ at $q$, then \text{MIN}(s)$ determines whether $\overline{pq}$ is a cut. The set WS supports insertion and deletion of cables, plus

- \text{WIDTH}(s)$, which returns the total width of the cables in WS lying strictly between the query segment $s$ and the origin $p$.

If some cable stretches from $p$ to $q$, then \text{WIDTH}(s)$ returns the width of the cable (if any) that crosses over the cut $\overline{pq}$.

The set FS is easily implemented so that each operation runs in $O(\log |F|)$ time by using a height-balanced search tree, sorted by distance from $p$. When two segments touch at their endpoints, their order in FS is unimportant, and so the closest segment to $p$ can always be defined. Since features never cross, the order of segments within the set does not change. To execute \text{MIN}(s)$, first query the condensed RBE to determine whether $p$ is connected to a feature in the direction of the scan ray. Return \textit{false} if so, and otherwise return \textit{true} if and only if $s$ is the first (leftmost) element of FS.

The structure WS can also be a height-balanced search tree. Since the number of cables in the condensed RBE is $O(|F|)$, each operation on WS will take $O(\log |F|)$ time. The \text{WIDTH} operation can be implemented by storing in each node the total width of the cables in its left subtree, plus the width of the cable stored in the node itself. These values are easy to maintain under the standard tree-balancing operations. The value \text{WIDTH}(s)$ can then be found by searching the tree for the farthest segment that is strictly closer than $s$. Every time the search path branches right or stops at a node, accumulate the quantity in that node. If the result is positive, it is \text{WIDTH}(s)$. Otherwise let $q$ be the point at which the scan ray intersects $s$, and query the RBE for the width of the cable (if any) from $p$ to $q$ that crosses over the cut $\overline{pq}$. Take the result to be \text{WIDTH}(s)$.

The event list for scanning around a point $p$ consists of two types of events. First, there is an event for every endpoint of a feature or cable in the RBE, except those objects that intersect $p$. A point may correspond to more than one event if two or more objects intersect there. Second, for each feature $f$ there is an event corresponding to a point $q$ on $f$ such that $||p - q||$ is critical. The line segment $\overline{pq}$
Section 1C

Testing the Routability of a Sketch

is called a possible critical cut; it is a critical cut unless it crosses a feature. The points of the event list are sorted by angle as seen from $p$; angles are measured with respect to some reference ray. Events at the same angle may be sorted arbitrarily. Each point in the list is marked with a pointer to its segment, and whether it is the first point of its segment, the closest point, or the last point. If the segment is seen edge-on, either point can be "first".

Algorithm T. (Tests the routability of a sketch.)
Input: The condensed rubber-band equivalent of a sketch $(F,T)$.
Local variables: Data structures FS and WS; event list EL; points $p$ and $q$; feature $f$; cable $c$; congestion value $t$.
Output: Either true (routable) or false (unroutable).
1. Group the features into islands;
2. foreach feature endpoint $p$ do
3. Initialize FS and WS to represent the reference ray;
4. Clear EL;
5. foreach feature $f$ do
6. if $f$ does not touch $p$ then add events to EL for $f$;
7. foreach cable $c$ do
8. if $c$ does not touch $p$ then add events to EL for $c$;
9. Sort EL;
10. foreach event $e \in$ EL (in sorted order) do
11. Update FS and WS;
12. if $e$ is the possible cut $\overline{pq}$ to feature $f$ and MIN?(f) then
13. $c \leftarrow$ WIDTH($f$) + width of crossing sequences of $\overline{pq}$ at $p$ and $\overline{qp}$ at $q$;
14. if $c > 0$ or $p$ and $q$ lie in different islands then
15. if $c >$ capacity of $\overline{pq}$ then return false;
16. return true.

The operation of the algorithm is simple. It first finds the islands of the sketch by checking which features intersect which others. This takes at most $O(|F|^2)$ time. Then, for each feature endpoint $p$, it initializes the data structures FS and WS, constructs the event list for $p$, and simply scans through it, taking appropriate actions for each event. If the event is the first or last one involving that segment, the segment is inserted or removed, respectively, from the appropriate set. If the event is the closest point $q$ on a feature $f$, and MIN?(f) is true, then the algorithm computes the congestion of the cut $\overline{pq}$ and checks whether this cut is empty. If not, and if its congestion is greater than the capacity of $\overline{pq}$, the algorithm signals that the sketch is unroutable.
1D. Routing a Sketch

This section presents a polynomial-time algorithm for producing a proper routing of a sketch, given its rubber-band equivalent. The algorithm minimizes the length of every trace in the routing, so that total trace length and the length of the longest trace are simultaneously optimized. To process the sketch $S = (F,T)$ the algorithm uses $O(|F| |T|)$ space and $O(|F| |T| \log |S|)$ time; these bounds are nearly optimal, for the output sketch may contain $\Omega(|F| |T|)$ trace segments. The output is guaranteed to be a proper routing if one exists, but otherwise it need not even be a sketch; it may contain illegal intersections. Hence the sketch to be routed should first be tested for routability using Algorithm T of the previous section.

The routing strategy

The routing algorithm examines the necessary crossings of cuts to generate constraints on the output traces. Every cut has a content, the sequence of traces that it necessarily crosses. Suppose that the cut $p \triangleright q$ from the island $P$ to the island $Q$ has content $\{ \theta_1, \ldots, \theta_n \}$. Any realization $\theta'_k$ of $\theta_k$ makes a crossing with $p \triangleright q$ that, in some sense, has $i-1$ traces between it and $P$ and $n-i$ traces between it and $Q$. Suppose this crossing occurs at the point $x$. If $\theta'_k$ is to be part of a proper routing, $x$ must be separated from both $p$ and $q$ as follows:

$$ ||x - p|| \geq \frac{\text{width}(P)}{2} + \frac{\text{width}(\theta_k)}{2} + \sum_{i=1}^{k-1} \text{width}(\theta_i); $$

$$ ||x - q|| \geq \frac{\text{width}(Q)}{2} + \frac{\text{width}(\theta_k)}{2} + \sum_{i=k+1}^{n} \text{width}(\theta_i). $$

The set of points $x$ on $\overline{pq}$ satisfying these two inequalities is a *doorway* for the trace $\theta_i$. It is empty if and only if the cut $\overline{pq}$ is unsafe. If the doorway $\overline{xy}$ of $\overline{pq}$ is not empty, the segments $\overline{px}$ and $\overline{qy}$ of its complement $\overline{pq} - \overline{xy}$ are called *struts* for $\theta_k$.

Roughly speaking, we route each trace by finding the shortest route that passes through all of its doorways. To make this process finite, we consider only the doorways in certain special cuts.

Here we require that the wiring norm be piecewise linear. I assume that the routing algorithm is given the wiring norm in the form of its *unit polygon*, the set of vectors of norm 1. The unit polygon defines certain *diagonal slopes*, which are the slopes of the lines through the origin that contain vertices of the unit polygon. A cut is called *diagonal* if its slope is diagonal and one of its endpoints is a feature endpoint. The routing algorithm considers only the doorways in diagonal cuts.
Figure 1d-1. The doorway for a crossing of a diagonal cut. All the traces (grey paths) have unit width, and the tick marks divide $\overline{pq}$ into segments of unit length. The line segment $\overline{x'y}$ is the doorway for the crossing $c$ between the cut $\overline{pq}$ and the trace $\theta$.

(Hence its time and space complexity depend linearly on the number of vertices in the unit polygon.) In other words, the algorithm finds for each trace the shortest route that passes through its diagonal doorways.

To compute this route directly is difficult, so we do not consider all the doorways at once. Instead we consider only one diagonal slope at a time. The diagonal cuts of that slope split the routing region into trapezoidal strips; the rubber band for a trace passes through the strips in a particular order, and hence it has a particular sequence of doorways. These doorways form a corridor as shown in Figure 1d-2. The shortest path through this corridor is called a partial realization of the trace, though like rubber bands, it may touch features other than its terminals. The remarkable fact is that one can merge the partial realizations of a trace, one for each diagonal slope, to form the optimal, or ideal, realization of that trace. The complete routing algorithm is summarized below.

Figure 1d-2. The corridor formed by a sequence of doorways for a trace. Light lines represent diagonal cuts, and medium lines are features. Where consecutive doorways are collinear, an extra doorway is added to preserve the applicability of Algorithm W. A doorway may consist only of a single point, but it still contributes both left and right vertices to the corridor.
Algorithm R. (Produces a detailed routing of a routable sketch.)
Input: the RBE of a routable sketch S; the wiring norm’s unit polygon.
Output: the ideal realization of each trace in S.
Local variables: array of partial realizations P.
Subroutines: Algorithm W is used in line 6.
1. \textbf{foreach} diagonal slope s \textbf{do}
2. \hspace{1em} Scan over the RBE with a line of slope s, producing doorways for all traces;
3. \hspace{1em} \textbf{foreach} trace \theta \textbf{do}
4. \hspace{2em} \rho \leftarrow \text{rubber band of } \theta;
5. \hspace{2em} Sort the doorways of slope s for \rho, producing a corridor;
6. \hspace{2em} P[\theta,s] \leftarrow \text{shortest path through this corridor;}
7. \hspace{1em} \textbf{foreach} trace \theta \textbf{do}
8. \hspace{2em} Merge the paths \( P[\theta,s] \) to form the realization of \( \theta \).

Constructing doorways and partial realizations
For each diagonal slope s, Algorithm R finds the doorways of slope s by scanning over the RBE with a line of slope s. The scanning technique is very similar to that used by Algorithm T, so I discuss it only briefly. We maintain the features and cables that cross the scan line in a pair of height-balanced trees. A feature enters or leaves the scan line when the scan line intersects one of its endpoints, say p. When this occurs we find the diagonal cuts of slope s incident on p by searching the feature tree for closest features not containing p. Having found a diagonal cut \( p \triangleright q \), we determine the content of \( p \triangleright q \) as explained in Section 1B. To do so we must know the sequence of cables that cross \( p \triangleright q \) strictly between p and q; a search of the cable tree will provide this information quickly. Finally, from the content of \( p \triangleright q \) we can construct the doorways for these traces in linear time. (The doorways are defined by equations (1–1) and (1–2).)

Next Algorithm R combines the doorways of each rubber band into a corridor for the corresponding trace. No sorting is actually required. Consider a trace \( \theta \) with rubber band \( \rho \). Each crossing of the diagonal cut \( \overline{pq} \) by \( \rho \) can be associated with a particular strand of \( \rho \). Hence every doorway for \( \theta \) is associated with a point on a strand of \( \rho \). By placing the doorways for each strand in a simple queue, the queues for \( \rho \) can be concatenated to yield the correct ordering of doorways for \( \theta \), and thus form a corridor. The shortest path through this corridor, which Algorithm W produces in linear time, is the partial realization of \( \theta \) for the diagonal slope s.

Merging the partial realizations
In its final phase, Algorithm R combines the partial realizations of each trace to form the output traces. Let us define a joint of a piecewise linear path to be a
Figure 1d-3. The major steps in the routing of a sketch. The wiring model here is rectilinear; its diagonal slopes are +1 and -1. All features and traces have unit width. Dark segments and points are features; grey lines are traces and their partial and ideal realizations; dashed segments are struts; and circles mark vertices of partial realizations that appear in the ideal realizations. Part (vi) shows that the ideal realizations can be altered so that they run in a grid. Algorithm R does not implement this process, but I discuss it in Chapter 10.
point where two segments of the path meet. The desired realization of a trace \( \theta \) is a piecewise linear path whose joints are chosen from among the joints of the partial realizations of \( \theta \). Let \( \sigma \) denote the partial realization of \( \theta \) for the diagonal slope \( s \), and let \( \xi \) denote the ideal realization of \( \theta \).

There is a simple geometric procedure for determining whether a joint of \( \sigma \) is retained as a joint of \( \xi \). Let \( \overline{ax} \) and \( \overline{ce} \) be consecutive segments of \( \sigma \), with \( x \) the joint between them; then \( x \) is one endpoint of a doorway \( \overline{xq} \) in a cut \( \overline{pq} \). We say that \( \sigma \) turns toward \( p \) at \( x \) if \( p \) is not exterior to the angle \( \angle axc \). The path \( \sigma \) turns toward either \( p \) or \( q \) at \( x \), but not both. Assume \( \sigma \) turns toward \( p \). Then \( x \) is retained if and only if the segments \( \overline{ax} \) and \( \overline{ce} \) touch the polygon \( \{ z : \|z - q\| = \|x - q\| \} \) at \( x \) alone. To check this condition, it suffices to compare the slopes of \( \overline{ax} \) and \( \overline{ce} \) to the slopes of certain segments in the unit polygon of the wiring norm.

![Diagram](image)

**Figure 1d-4. Evaluating the joints of a partial realization.** Here the partial realization \( \sigma \) has a joint \( x \) on the diagonal cut \( \overline{pq} \), and \( \sigma \) turns toward \( p \) at \( x \). We associate with \( x \) the two polygons \( \{ z : \|z - p\| = \|x - p\| \} \) and \( \{ z : \|z - q\| = \|x - q\| \} \), shown here in grey. Part (i) shows the range of angles that \( \sigma \) may make at \( x \) if \( x \) is to appear as a joint in the full realization. Similarly, part (ii) shows the range of angles that \( \sigma \) may make at consecutive joints \( x \) and \( x' \) if the segment \( \overline{xx'} \) is to appear as a segment of the full realization.

It remains to find the correct ordering of the joints of \( \xi \). Three simple rules govern this process. First, the joints of \( \xi \) that come from a partial realization \( \sigma \) have the same order and orientation in \( \xi \) as in \( \sigma \). By the orientation of a joint I mean whether the path turns to the left or the right at the joint. Second, the joint of \( \xi \) that follows a joint \( x \) of \( \sigma \) is either another joint of \( \sigma \), or else it comes from a partial realization \( \sigma' \) chosen as follows. If \( \xi \) turns left at \( x \), then \( \sigma' \) corresponds to the next diagonal slope counterclockwise from \( s \). Otherwise, if \( \sigma \) turns right
at \( x \), then \( \sigma' \) corresponds to the next diagonal slope clockwise from \( s \). The third rule determines when two consecutive joints of \( \sigma \) are consecutive in \( \xi \). Let \( x \) and \( x' \) denote these two joints, and let \( p \) and \( p' \) denote the corresponding feature endpoints toward which \( \sigma \) turns. The joints \( x \) and \( x' \) are consecutive in \( \xi \) if and only if the line segment between them intersects the polygons \( \{ z : \|z - p\| = \|x - p\| \} \) and \( \{ z' : \|z' - p'\| = \|x' - p'\| \} \) on their boundaries only. Again, this can be checked by comparing the slope of \( xx' \) to the slopes of certain segments in the unit polygon of the wiring norm.

An extension of the third rule allows the merging process to start and finish. Let \( t \) be the first (or last) terminal of \( \Theta \). The first (or last) joint \( x \) of \( \xi \) has the property that the line segment \( tx \) intersects the polygon \( \{ z : \|z - p\| = \|x - p\| \} \) only on its boundary, where \( p \) is the feature endpoint toward which \( \sigma \) turns at \( x \). Together these rules determine a unique piecewise linear path \( \xi \). It can be produced in linear time from the partial realizations of \( \Theta \), provided that the input sketch is routable.

**Attempting to route an unroutable sketch**

If the input to Algorithm R is the RBE of an unroutable sketch, then one of two things can happen. One possibility is that the process of merging partial realizations gets stuck: either it reaches a point where none of the available joints can be added, or it reaches the final terminal of the trace without having used all the available joints. The other possibility is that the merge completes successfully, but that the traces it has produced form an improper sketch. I conjecture that the latter possibility never arises, so that Algorithm R can always determine whether its input sketch is routable. If this conjecture proves true, then one need not apply Algorithm R to the input of Algorithm T to test for routability. One advantage of Algorithm T, however, is that it identifies the unsafe cuts that make the sketch unroutable. Algorithm R does not have this ability, and it can consume far more space than Algorithm T.

**Complexity analysis**

Algorithm R uses at most \( O(|F| |T|) \) space to route a sketch \((F, T)\). We mentioned in Section 1B that the detailed RBE is no larger than this. The number of doorways generated by phase one is also \( O(|F| |T|) \), because each wire segment in the original sketch can cause at most one crossing of each diagonal cut. The output sketch fits in the same amount of space because its wire segments have endpoints on distinct doorways. On the other hand, sketches exist whose only detailed routings occupy \( O(|F| |T|) \) space, so the space bound of Algorithm R is asymptotically optimal.
All the operations performed by Algorithm R take time linear in the size of its data structures, except the sorting that precedes the scanning operations, which requires logarithmic time per object. The time taken by Algorithm R is therefore at most $O(|F||T|\log |S|)$. In practice, the number of crossings between diagonal cuts and wires should be much less than $|F||T|$, and the algorithm should correspondingly faster. I have no experimental data to this effect, however.

1E. Efficiency Concerns

Seen from a theoretical standpoint, the algorithms for sketch routing and routability testing are quite efficient. Their worst-case running times are similar and seemingly close to optimal: $\Theta(n^2 \log n)$ on input of size $n$. In particular, the resource bounds of Algorithm R are optimal to within logarithmic factors on some inputs; there exist sketches of size $n$, like that in Figure 1e-1, whose only proper realizations have size $\Omega(n^2)$. From a practical standpoint, however, the sketch algorithms do not seem as good. Most programs that operate on integrated circuit designs have empirical running times close to linear, or at worst $O(n^{3/2})$ on input of size $n$. Since VLSI circuits are so huge, slower algorithms cannot be tolerated except when applied to small cells within a larger design.

Figure 1e-1. A small sketch whose proper realization is large. If the distance between adjacent dotted lines is 1 unit and the unit polygon is square, then the only proper realization of the sketch on the left is the sketch on the right.

This section suggests two ways of speeding up routing and testing routability of sketches. One approach concerns worst-case performance. Algorithm R can be modified, without changing its underlying strategy, to eliminate the logarithmic factor in its time bound. Probably the running time of Algorithm T can also be improved to $O(n^2)$, at the cost of using $\Theta(n^2)$ space in every instance. The other
approach concerns average-case performance. It begins with an investigation of the expected performance of Algorithms T and R on practical circuits, and then explores three methods for speeding up the performance bottleneck, which turns out to be Algorithm T. Two are described here; Section 1F is devoted to the third. All three methods involve paring down the number of critical cuts whose congestion needs to be computed. The result is a set of algorithms for the sketch routability problem whose average-case performance ranges from $\Theta(n \log n)$ to $\Theta(n^{3/2}\log n)$, depending upon assumptions concerning the placement of traces and features in a typical sketch. I should emphasize that these results are not based on any experimental evidence; I have not implemented any sketch algorithms. Instead I derive estimates of running time and space usage from models of the distribution of features and traces in the sketches input to Algorithms T and R.

Eliminating logarithmic factors

Recently I realized that the rubber-band equivalent is not the best data structure for Algorithm R to use in computing doorways. A faster method is to compute for each diagonal slope $s$ a realization (not proper) whose traces cross the diagonal cuts of slope $s$ as seldom as possible. Section 9B explains how to compute in time and space $O(|F||T|)$ a structure called a reduced intersection graph that represents the necessary crossings of those traces and cuts. The content of a diagonal cut of slope $s$ can be read off directly from the reduced intersection graph, as can the sequence of such cuts that each trace passes through. As a result the corridors for partial realizations can be computed in time $O(|F||T|)$ per diagonal slope. Since the only logarithmic factors in Algorithm R came from constructing and scanning over the RBE, the result is an algorithm for sketch routing that runs in time $O(|F||T|)$ plus $O(|F|\log |F|)$ to scan for diagonal cuts. In fact, its running time is essentially proportional to the number of crossings between traces in the input sketch and diagonal cuts. The only reason not to adopt this approach is that my sketch compaction algorithm, currently the only client of Algorithm R, applies Algorithm R to a rubber-band equivalent rather than a sketch.

A similar improvement may be possible in Algorithm T. Leo Guibas [11] has suggested that the scanning in Algorithm T can be replaced by a topological sweep [11], reducing the worst-case running time from $O(n^2 \log n)$ to $O(n^2)$. To take advantage of this speedup, and to obtain a similar speedup in the construction of the condensed RBE, that structure must be represented in the form of an adjacency matrix. Hence $\Theta(n^2)$ space is required in every instance, as opposed to $\Theta(n)$ space for Algorithm T as it stands. Consequently this improvement is of academic interest only.
The proper measure of input size

The sketch algorithms so far described—for constructing the RBE, testing routability, and routing—all run in essentially quadratic time, but this running time arises from different causes in each case. When constructing the RBE, the number of crossings between trace segments and doors determines the running time to within a logarithmic factor. When testing routability, the time complexity is determined by the number of pairs of features in the condensed RBE, again with an added logarithmic factor. And when routing a sketch, the relevant quantities are the number of strands in the RBE and the number of crossings between trace segments and diagonal cuts. Of these quantities, only the number of pairs of features is generally $\Theta(n^2)$. Likewise, though the space usage of Algorithm R is $\Theta(n^2)$ in the worst case, it is actually proportional to the number of crossings between traces and certain cuts (doors and diagonal cuts).

I argue that a sketch algorithm whose resource usage is nearly proportional to the number of crossings between traces and $O(n)$ cuts is really quite efficient. Whether the expected number of such crossings is close to linear in $n$ depends on one's source of sketches. But in any case, that quantity is a more reasonable measure of sketch complexity than $n$, the number of feature and trace segments in the sketch. The reason is that one can encode quite complicated sketches with just a few segments. Each trace segment in the input sketch can, in principle, span the width of the sketch. One would prefer a measure of sketch size that accounted for the lengths of trace segments. Probably no such measure is convenient, but if one adopts this viewpoint, the complexity of Algorithm R, in particular, seems much smaller. So the only algorithm that could really stand improvement is Algorithm T. The bottleneck is the repeated scanning around feature endpoints for critical cuts, which takes $O(n \log n)$ time per feature whether such critical cuts are found or not.

Typical sketches

What properties of practical sketches can we exploit to speed up Algorithm T? I submit that there are at least three: density, uniformity, and locality. By density I mean that the components in typical circuit layouts are tightly packed. Depending on how the layout components are represented in sketches, the only features visible from a given feature may be a few of its nearest neighbors. (If almost all features are points, then visibility is essentially unlimited. But if many features are line segments, then expected visibility is bounded independently of sketch size. This fact is independent of density, but the constant—the expected number of features visible from a given feature—does depend on density.) In this case the expected number of critical cuts is $\Theta(n)$. The second principle, uniformity, says that the elements of a sketch are distributed nearly uniformly in a rectangular region of small aspect
ratio. Applied to the rubber-band equivalent, it implies that the average number of cables crossed by a critical cut is $O(1)$ if visibility is restricted, and $O(\sqrt{n})$ if visibility is unrestricted. Finally, locality suggests that local constraints almost always dominate over nonlocal ones. In other words, it is highly unlikely that a long nonempty critical cut is unsafe without some shorter nonempty critical cut being unsafe also. None of these principles can be justified formally, but I think that programmers of circuit design systems will agree that they are reasonable assumptions.

Checking critical cuts: two approaches

The locality principle has immediate application to routability testing. Rather than scanning the entire sketch from each feature endpoint, one could scan only part of the sketch each time. For example, one might first divide the components of the RBE into bins, each bin corresponding to a square region of the sketch. For each feature endpoint, one could then include only the components in its bin and the adjacent bins in the scanning operation. This technique should be fast, but it has the drawback of relying on the locality principle for correctness, not just performance. If an unsafe, nonempty, critical cut is found, the sketch is proven to be unroutable. But if no such cut is found, the sketch is not necessarily routable. Finding a good tradeoff between speed and risk of error would probably require extensive experimentation.

A less risky approach to routability testing relies instead on the assumption of density. Rather than locating the critical cuts by scanning, we obtain them from the visibility graph of the sketch. The visibility graph $(V, E)$ of a sketch $(F, T)$ is a graph whose vertices are the endpoints of features in $F$ and whose edges are the line segments in $F$ and all straight cuts between endpoints of features in $F$. The edges emanating from each feature endpoint are sorted in clockwise order. One can compute the visibility in time $\Theta(|E| + |F| \log |F'|)$ and space $\Theta(|E|)$ by the methods of [14]. The running time averages $\Theta(|F| \log |F'|)$ if our sketch is dense, meaning that the expected number of cuts between feature endpoints is $\Theta(|F'|)$, and is $\Theta(|F|^2)$ in the worst case. Given the visibility graph $(V, E)$ of a sketch, the critical cuts can be enumerated in time $\Theta(|E|)$. For each feature endpoint $p$, one can list the portions of features visible from that endpoint, and check which of those portions contain the closest points to $p$ on their respective features.

Having identified the critical cuts, one must compute their congestions without scanning. The simplest way to do so, based on what we already know, is to make separate queries to the condensed RBE for each critical cut. The condensed rubber-band equivalent of a sketch is like an embedded planar multigraph. We may consider it one since although some of its arcs overlap, they are circularly ordered at the
nodes they connect. Some of the faces of this graph are polygonal, and some are degenerate (where two edges connect the same feature endpoints). In $O(|F| \log |F|)$ time we can add edges so as to triangulate the polygonal faces, keeping the size of the whole graph linear in $|F|$. Now we compute the dual graph: the graph whose nodes are the faces of the original graph and whose edges represent adjacency across the original edges. This computation takes linear time.

Every cut now corresponds to a path in the dual graph whose length is the number of cables crossed over by the cut. One can find this path in time proportional to its length, simply by walking through the dual graph. The congestion of the cut $\overline{pq}$ is the sum of the weights of the cables that $\overline{pq}$ crosses over, plus the widths of the crossing sequences of $\overline{pq}$ and $\overline{qp}$. (As usual, when a cable lies within $\overline{pq}$, it may or may not contribute to $\text{cong}(\overline{pq})$, depending on which of the three possible cables from $p$ to $q$ it is.) The crossing sequence terms are provided by the condensed RBE at a cost of $O(\log n)$ time per cut. We conclude that after $O(n \log n)$ preprocessing operations on the condensed RBE, the congestion of a cut can be computed in time $O(\log n)$ plus $O(1)$ per cable it crosses over.

Using both data structures—the visibility graph and dual of the condensed RBE—we obtain an algorithm for testing sketch routability whose performance is potentially far superior to that of Algorithm T. Under the most optimistic assumptions of density and uniformity, the expected running time is $\Theta(n \log n)$. In the very worst case, $\Theta(n^3)$ time might be consumed. One drawback to this approach is the complexity of implementing the algorithm that constructs the visibility graph.

1F. Faster Routability Testing

This section describes a very powerful technique for speeding up routability testing in piecewise wiring norms, the kind we use. It results in a routability testing algorithm that runs in time $O(n^{3/2} \log n)$ on typical sketches of size $n$, without needing any more than linear space.

The key to routability testing is finding a small set of decisive cuts. So far we have considered methods for identifying and checking the critical cuts in a sketch. Critical cuts are decisive but difficult to enumerate, since every pair of features can potentially generate a critical cut. To determine which of the minimum-length paths between features are actually critical cuts, one must either consider all pairs of features (as Algorithm T does by scanning) or construct something like a visibility graph. By exploiting a property of cuts under piecewise linear norms, we can eliminate many pairs of features from consideration, whether or not they generate a critical cut. This property, called shadowing, was discovered by Cole and Siegel [6] and independently by me. Some line segments in the sketch are shadowed by
other features, and even if they are critical cuts, they need not be checked. If
the arrangement of features in the sketch is close to uniform, then most cuts are
shadowed, and one can quickly generate a decisive set of unshadowed cuts.

Definition of shadowing

The principle of shadowing is that no cut $\overline{pq}$ need be checked if there is a point $r$
on a feature such that

$$||p - q|| = ||p - r|| + ||q - r||.$$  \hspace{1cm} (1.3)

If the point $p$ is considered fixed, the point $r$ casts a shadow consisting of all points $q$
which, together with $p$ and $r$, satisfy (1.3). We say that the cut $\overline{pq}$ is shadowed
(by $r$). Typical shadows for the rectilinear ($L^\infty$) wiring norm are pictured in Figure
1f-1. If the norm $|| \cdot ||$ were the euclidean norm, this shadow would be nothing
more than the ray starting at $r$ and pointing away from $p$. But since the wiring
norm is piecewise linear, shadows can have substantial size. More to the point,
if the features in a sketch are evenly distributed, then the number of unshadowed
features, as seen from a given feature endpoint, is likely to be small: $O(\log |F|)$ on
the average. Later in this section I justify this bound and explain why shadowed
cuts may be ignored.

\begin{figure}[h]
  \centering
  \includegraphics[width=0.5\textwidth]{shadowing.png}
  \caption{Shadows in the rectilinear norm. With respect to the central point,
each of the other points casts a shadow, shown as a shaded region. Shadows
include their frontiers. Darker shades represent multiple overlapping shadows. The
dashed lines are the lines of diagonal slope passing through the central point. Points on
these lines are shadowed only by other such points.}
\end{figure}

We use shadowing to find a small decisive set of cuts for a sketch. This set
contains the diagonal cuts in the sketch, of which there are $O(|F|)$, and the unshadowed
cuts between feature endpoints, of which there are typically $O(|F|\log |F|)$. The expected time needed to find these cuts is also $O(|F|\log |F|)$. One can compute
the congestion of the decisive cuts from the dual of the condensed RBE, as
described in Section 1E, at an average cost of $O(\sqrt{|F|})$ time per cut. If the cuts
are checked as they are produced, none of our data structures grows larger than
$O(|F|)$. The result is an algorithm for sketch routability that consumes only linear
space and $O(|F|^{3/2} \log |F|)$ time, plus that needed to construct the condensed RBE, for typical sketches.

**Scanning for unshadowed cuts**

Simple scanning algorithms suffice for finding the decisive cut set. Diagonal cuts, in particular, are easy to find by scanning with lines of diagonal slope as Algorithm R does. I now present an algorithm that enumerates the other desired cuts. Shadowing works best when there are only two diagonal slopes, as when the wiring norm is rectilinear. In this case, scanning for unshadowed cuts takes time $O(|F| \log |F|)$ plus $O(1)$ per cut found. For simplicity I illustrate the algorithm using the taxicab ($L^1$) norm, defined by $||(x, y)|| = |x| + |y|$, which is the rectilinear norm rotated through $\pi/4$ radians and rescaled by $\sqrt{2}$. In the taxicab norm the points that can shadow a cut $\overline{pq}$ are the points in the rectangle whose sides are aligned with the axes and which has $p$ and $q$ at opposite corners. I also simplify matters by assuming that all features are points. It matters little if the algorithm outputs a line segment that is not really a cut, because the fact that it is not a cut will be discovered when trying to compute its congestion.

We compute all the unshadowed cuts between feature endpoints by scanning over the sketch from left to right with a vertical line. Actually, we skip some unshadowed cuts that are diagonal and produce some cuts that are just on the boundary of being shadowed, but these discrepancies cause no problems. The scan considers only feature endpoints. At all times during the scan, we maintain a data structure that contains every feature endpoint lying left of the scan line, except that where two or more feature endpoints have the same $y$-coordinate, only the rightmost is kept. These feature endpoints are kept sorted by $y$-coordinate, presumably in some height-balanced tree to enable fast insertion. Each point in the structure also has two links to other points in the structure, an **upward** link and a **downward** link. The upward link of $p$ points to the feature endpoint above it and strictly to its right that is closest to $p$ in $y$-coordinate. If no such point exists, then the upward link is **nil**. The downward link of $p$ is similar, but points to the closest feature endpoint below it and strictly to its right. See Figure 1f-2(a).

Adding a new feature endpoint to the structure is simple. Figure 1f-2(b) illustrates the process. Call the new endpoint $q$. One first finds the feature endpoints left of $q$ that lie just above and below $q$ in $y$-coordinate. Denote them by $p^+$ and $p^-$, respectively. They are identical if $q$ has the same $y$-coordinate as a point already considered. Next one finds the unshadowed cuts incident on $q$ from the left, while at the same time updating the up and down links of the existing points. Beginning at $p^+$, follow the upward links until reaching nil or a point on the scan line. All the points in this chain have unshadowed cuts to $q$, and their downward links must be
modified to point to \( q \). Next, starting at \( p^- \), follow the down links until reaching \( \text{nil} \) or a point on the scan line. All the points in this chain have unshadowed cuts to \( q \), and their upward links should now connect them to \( q \). If \( p^+ = p^- \), one deletes this point. Finally one sets the upward and downward links of \( q \) to \( \text{nil} \).

The correctness and complexity analysis of this method are straightforward. Processing one feature endpoint takes time \( O(\log |F|) \) per feature endpoint plus \( O(1) \) time per cut produced. Applied to each feature endpoint in turn, it produces all the unshadowed cuts between them except those that are vertical. In the taxicab norm, vertical cuts between feature endpoints are diagonal cuts. Because the diagonal cuts are gathered separately, there is no harm in ignoring them here. Similarly, the horizontal cuts generated in the scan for unshadowed cuts may be dropped to avoid duplication.

**More complicated wiring norms**

No fundamental changes are needed in the scanning algorithm if the unit polygon of the wiring norm is a parallelogram and not a square. The unit polygon is a parallelogram if and only if the wiring norm has exactly two diagonal slopes. One simply redefines one diagonal slope to be "vertical" and the other to be "horizontal", and reinterprets the terms ‘above’, ‘below’, ‘left’, and ‘right’ accordingly. Equivalently, one may rotate and skew the sketch and its wiring norm so that one diagonal slope actually is vertical and the other is horizontal, and apply the inverse transformation to each cut generated.

Scanning for unshadowed cuts is somewhat more complicated when the unit
polygon of the wiring norm has more than four sides. In this case several scans are needed to produce all the cuts. Each scan produces the unshadowed cuts whose slopes lie in a certain range. For each diagonal slope \( s \), we need a scan that generates the unshadowed cuts whose slopes lie between \( s \) and the diagonal slope \( t \) immediately clockwise from \( s \). In this scan we pretend that \( s \) and \( t \) are the only diagonal slopes, and throw away the generated cuts whose slopes do not lie clockwise between \( s \) and \( t \). (Which cuts are shadowed is independent of all properties of the wiring norm except the diagonal slopes.) The remaining unshadowed cuts are, in fact, unshadowed in the original wiring norm. Since some of the generated cuts have to be thrown away, we can no longer claim that the scanning algorithm runs in time \( O(|F| \log |F|) \) plus \( O(1) \) per unshadowed cut. Nevertheless, the average number of cuts thrown away, as well as the average number retained, is only \( O(|F| \log |F|) \) per diagonal slope. We now justify this figure.

The number of unshadowed cuts

The average-case time complexity of the new routability testing method depends foremost on the number of unshadowed cuts. More accurately, it depends on the number of line segments output by the scanning procedure, which is approximately the number of unshadowed cuts that would exist if each feature endpoint were a feature unto itself, i.e., if all features were points. We now estimate this quantity; it turns out to be \( O(n \log n) \) where \( n = \Theta(|F|) \) is the number of feature endpoints. As a model of the distribution of feature endpoints, we assume that \( n \) points are independently and uniformly distributed in the unit square \( I \times I \). The size of the square is irrelevant to the present analysis. We wish to estimate the expected number of pairs \((p, q)\) of these points for which the cut \( pq \) unshadowed. Since expectation is additive, regardless of independence, it suffices to determine the chance that a particular cut \( pq \) is unshadowed, and multiply this chance by \( \binom{n}{2} \).

An approximate analysis shows that the probability of a cut being unshadowed is \( O(\ln(n)/n) \). Let \( p \) and \( q \) be two of the randomly placed points, and let \( \Box pq \) denote the rectangular region with diagonal \( pq \) whose sides are aligned with the axes. Define a random variable \( A \) whose value is the area of \( \Box pq \). The cut \( pq \) is output by the scanning procedure only if the inside of \( \Box pq \) contains none of the \( n \) points except \( p \) and \( q \). This event has probability \( (1 - A)^{n-2} \). Define the random variables \( X \) and \( Y \) to be the horizontal and vertical separations, respectively, of \( p \) and \( q \). Almost all the contribution to the chance that \( pq \) is empty comes from small \( X \) and \( Y \). We are willing to ignore constant factors, so there is no harm in pretending that \( X \) and \( Y \) are uniformly distributed in \([0, 1]\). If this were true, then
for \( a \in [0, 1] \) we would have

\[
\Pr[A \leq a] = \iint_{xy \leq a} 1 \, dy \, dx \\
= \int_0^a \int_0^1 1 \, dy \, dx + \int_a^1 \int_0^{a/x} 1 \, dy \, dx \\
= a + \int_a^1 a/x \, dx = a(1 - \ln a).
\]

Differentiating with respect to \( a \) gives \(-\ln a\), so \( a \mapsto -\ln a \) is a good estimate of the density function for \( A \). Hence the probability that \( \overline{pq} \) is unshadowed is on the order of

\[
\int_0^1 (1 - a)^{n-2}(-\ln a) \, da = -\int_0^1 u^{n-2} \ln(1 - u) \, du.
\]

Now we integrate by parts, choosing the antiderivative \((u^{n-1} - 1)/(n - 1)\) for \( u^{n-2} \), and thus obtain

\[
-\int_0^1 u^{n-2} \ln(1 - u) = -\left[ \frac{u^{n-1} - 1}{n - 1} \ln(1 - u) \right]_0^1 + \int_0^1 \frac{u^{n-1} - 1}{(n - 1)} \cdot \frac{-1}{1 - u} \, du.
\]

The bracketed term vanishes, and we are left with the integral

\[
\frac{1}{n - 1} \int_0^1 \frac{u^{n-1} - 1}{u - 1} \, du = \frac{1}{n - 1} \int_0^1 (1 + u + u^2 + \cdots + u^{n-2}) \, du
\]

\[
= \frac{1}{n - 1} \sum_{k=1}^{n-1} \frac{1}{k} \approx \frac{\ln n}{n}.
\]

The expected number of unshadowed cuts among the \( n \) points is therefore \( \binom{n}{2} \) times \( O(\ln(n)/n) \), which is \( O(n \log n) \).

This analysis changes only quantitatively, not qualitatively, if the wiring norm is not the taxicab norm. Because of the way we break down a complicated wiring norm into wiring norms with two diagonal slopes each, it suffices to consider a wiring norm whose unit polygon is an arbitrary parallelogram. Then the points that can shadow a cut \( \overline{pq} \) all lie in a parallelogram \( \diamond pq \) with \( p \) and \( q \) at opposite corners. This parallelogram takes the place of \( \square pq \). The distribution of the area of \( \diamond pq \) is asymptotic to that of \( \square pq \) as area approaches 0, and hence the chance that \( \overline{pq} \) is unshadowed remains \( O(\ln(n)/n) \). Therefore the expected number of cuts generated while scanning is \( O(n \log n) \) per diagonal slope.
An explanation of shadowing

Why should the unshadowed cuts between feature endpoints and the diagonal cuts form a decisive set? The answer to this question has two parts. The first part notes that the set of all cuts between feature endpoints, together with the diagonal cuts, constitute a decisive set. For lack of a better word, let us call these cuts pivotal. Pivotal cuts are strongly related to critical cuts. Recall that a critical cut is a cut from a feature endpoint to the closest point on another feature as measured in the wiring norm, with ties broken using the euclidean norm. As you might guess, the method of tiebreaking is arbitrary. Instead one can use a tiebreaker that always picks a diagonal cut or a cut between feature endpoints. For if \( p \) is any point and \( Q \) is a feature not containing \( p \), there is a point \( q \in Q \) minimizing \( ||q - p|| \) such that either \( q \) is an endpoint of \( Q \) or the slope of \( \overline{pq} \) is diagonal. Hence the pivotal cuts form a decisive set for the same reason that critical cuts do. (See Proposition 8b.4.)

The second part of the answer explains why shadowed cuts need not be checked. Shadowing derives its power from the following lemma.

**Lemma:** Let \( \overline{pq}, \overline{pr}, \text{ and } \overline{qr} \) be cuts in a sketch. Assume that \( r \) shadows \( \overline{pq} \) and that the inside of the triangle \( \triangle pqr \) is free of features. If \( \overline{pq} \) is unsafe and nonempty, then either \( \overline{pr} \) or \( \overline{qr} \) is unsafe and nonempty.

The idea behind the lemma is the following. Let \( P, Q, \text{ and } R \) denote the islands containing \( p, q, \text{ and } r \), respectively. Then by equation (1–3) and the definition of capacity, we have

\[
\text{capacity of } \overline{pq} = \text{capacity of } \overline{pr} + \text{capacity of } \overline{qr} + \text{width of } R. \quad (1–4)
\]

On the other hand, Figure 1f.3 suggests that

\[
\text{congestion of } \overline{pq} \leq \text{congestion of } \overline{pr} + \text{congestion of } \overline{qr} + \text{width of } R, \quad (1–5)
\]

since the trace \( \rho \) is no wider than its terminal \( R \). This inequality can be proven using the machinery of Section 4F (Proposition 4f.1) and Chapter 8. Subtracting the relation (1–5) from equation (1–4), we infer that the margin of safety of \( \overline{pq} \), the difference between its capacity and congestion, is at most the sum of the margins of safety of \( \overline{pr} \) and \( \overline{qr} \). Hence if \( \overline{pq} \) is unsafe—if its margin of safety is negative—then either \( \overline{pr} \) or \( \overline{qr} \) is unsafe. Moreover, if \( \overline{pq} \) is also nonempty, one of \( \overline{pr} \) and \( \overline{qr} \) is unsafe and nonempty. I leave this last deduction as an exercise.

The lemma gives us a condition under which a shadowed cut \( \overline{pq} \) need not be checked. Suppose the cut \( \overline{pq} \) is shadowed by a point \( s \). Let \( r \) be the closest point on a feature to \( \overline{pq} \), excluding \( p \) and \( q \), in the closed region bounded by the triangle \( \triangle pqs \). Then \( \overline{pr} \) and \( \overline{qr} \) are cuts, the inside of \( \triangle pqr \) is empty, and \( r \) shadows \( \overline{pq} \).
Figure 1f.3. An inequality concerning congestion. If the inside of $\Delta pqr$ is free of features, then all traces (darker shaded paths) that necessarily cross $\overline{pq}$ also necessarily cross either $\overline{pr}$ or $\overline{qr}$, with at most one exception: one trace (lightly shaded path) can have $r$ as a terminal.

Hence the lemma applies to $p$, $q$, and $r$. It shows that $\overline{pq}$ cannot be unsafe and nonempty unless either $\overline{pr}$ or $\overline{qr}$ has the same properties. So we can avoid checking $\overline{pq}$ if we can determine that neither $\overline{pr}$ nor $\overline{qr}$ is both unsafe and nonempty.

Now we invoke a special property of pivotal cuts, which can be traced all the way to Lemma 7c.3. If any cut in the sketch is unsafe and nonempty, then the sketch has a pivotal cut that is also unsafe and nonempty, simply because the pivotal cuts are decisive. The special property is this: the unsafe and nonempty pivotal cut is no longer than the original cut, as measured by the wiring norm. To ensure that $\overline{pr}$ and $\overline{qr}$ are safe, it therefore suffices to check pivotal cuts that are shorter than $\overline{pr}$ and $\overline{qr}$, and are therefore shorter than $\overline{pq}$. In other words, we may remove $\overline{pq}$ from our decisive cut set, which consisted originally of the pivotal cuts. The same principle applies to all shadowed pivotal cuts, and hence the unshadowed pivotal cuts form a decisive set. This set contains precisely all the unshadowed cuts between feature endpoints and all the diagonal cuts, since diagonal cuts are never shadowed.
Chapter 2

Topological Preliminaries

Point-set topology and elementary homotopy theory form the basis for all the mathematical work in this thesis. Since point-set topology is more widely known, and too large a subject to be covered here, I assume familiarity with its basic concepts and the relationships among them. The reader should know the definition of the terms basis, component, embedding, homeomorphism, path, and quotient space, the concept of a local property, and what it means for a space to be compact, connected, Hausdorff, metric, normal, or path-connected. For those readers who wish to refresh their memories, I have provided definitions of these terms in the glossary. An excellent reference, both for point-set topology and for an introduction to homotopy theory, is the text by Munkres [38].

Unlike the other chapters, this chapter contains little or no original material; it merely encapsulates known results for future reference. As the nomenclature of topology is not entirely standardized, the first part of the chapter describes the terms and symbols I have adopted. Everyone who wishes to study the mathematical parts of this thesis should read these definitions, because Chapters 3 through 8 depend on them. The rest of the chapter reviews some elementary results from different branches of topology. Sections 2A and 2B introduce homotopy theory at an elementary level. I have provided proofs for the easier results to help the reader assimilate the definitions. Section 2C discusses some facts about plane curves that will be used from time to time. Lastly, Section 2D distills the results we will need concerning topological manifolds. The proofs in the final section rely on the machinery of homology theory, but no homology theory is used elsewhere in this thesis.

Topological spaces and maps

A space always means a topological space, and a map on topological spaces always means a continuous function. The following are standard topological spaces:

- the unit interval $I = [0, 1]$,
- the euclidean spaces $\mathbb{R}^n$, for $n \geq 1$, 
the euclidean half-spaces \( H^n = \{ (x_1, \ldots, x_n) \in \mathbb{R}^n : x_n \geq 0 \} \), for \( n \geq 1 \),
the spheres \( S^n = \{ x \in \mathbb{R}^{n+1} : |x| = 1 \} \), for \( n \geq 0 \).

In every case, the superscript denotes the topological dimension of the space itself, and not the dimension of the space in which it is embedded. I reserve the right to use each of the symbols \( R, H \), and \( S \) without a superscript to mean something other than the spaces listed above. In particular, \( R^1 \) should be distinguished from \( R \), which need not denote the real line. When a space such as \( \{ x \} \) contains only one element, I frequently omit the braces and write simply \( x \).

A subspace \( A \subseteq X \) is a retract of \( X \) if there is a map \( r : X \rightarrow A \), called a retraction, such that \( r(a) = a \) for all \( a \in A \). The spaces \( I \) and \( R^1 \) are absolute retracts in the following sense. If \( I \) or \( R^1 \) is embedded in a normal space \( X \) as a closed subspace \( A \), then \( A \) is a retract of \( X \).

For every subspace \( A \) of a topological space \( X \), the following subspaces of \( X \) are defined.
- Its interior \( \text{Int} A \), the union of the open sets contained in \( A \).
- Its closure \( \text{Cl} A \), the intersection of the closed sets that contain \( A \).
- Its frontier, or "topological boundary", which is \( \text{Fr} A = \text{Cl} A - \text{Int} A \).

The term 'boundary' is reserved for use with manifolds.

I employ a few convenient devices for describing maps. If \( E(t) \) is any expression, then \( t \mapsto E(t) \) denotes the function whose value at \( t \) is \( E(t) \). The domain and range of this function should be inferred from context. If \( F : X \times Y \rightarrow Z \) is a function with two arguments, then \( F(x_0, \cdot) : Y \rightarrow Z \) is the function \( y \mapsto F(x_0, y) \), and \( F(\cdot, y_0) : X \rightarrow Z \) is the function \( x \mapsto F(x, y_0) \). This "dot" notation generalizes to more complicated expressions. If \( f : X \rightarrow Y \) and \( U \subseteq X \), then \( f|_U \) denotes the restriction of \( f \) to \( U \). We write \( f(U) \) for the set \( \text{Im} f|_U = \{ f(u) : u \in U \} \). The symbol \( \text{id}_X \) denotes the identity map on the space \( X \).

Paths and their images

A path \( \alpha \) is always a continuous function with domain \( I \), and should not be confused with its image \( \text{Im} \alpha \). When we speak of a path intersecting a set or another path, however, we are implicitly referring to the image of that path. The endpoints of a path \( \alpha \) are the points \( \alpha(0) \) and \( \alpha(1) \), and are considered as an ordered pair. Thus \( \alpha \) and \( \beta \) have the same endpoints if \( \alpha(0) = \beta(0) \) and \( \alpha(1) = \beta(1) \). I write \( \alpha : A \rightarrow B \) to mean that \( \alpha \) is a path with \( \alpha(0) \in A \) and \( \alpha(1) \in B \). The middle of the path \( \alpha \) is the set \( \text{Mid} \alpha = \alpha((0,1)) \).

The distinction between paths and their images is reflected by the distinction between linear paths and line segments. In a space such as \( \mathbb{R}^n \) that has a linear structure, the linear path \( x \rightarrow y \) is the path \( t \mapsto (1-t)x + ty \), whereas the line segment \( \overline{x y} \) is the set \( \text{Im}(x \rightarrow y) \). We say \( \alpha \) is piecewise linear if there is a partition...
0 = t_0 < t_1 < \cdots < t_n = 1 of I such that \alpha is linear on each interval [t_{i-1}, t_i]. If this partition is minimal, so that \alpha is not linear on any interval [t_{i-2}, t_i], then we call the points \alpha(t_i) the vertices of \alpha. A piecewise linear, injective path is called simple. A loop is a path whose endpoints coincide; the loop \alpha is simple if \alpha is piecewise linear and \alpha(s) = \alpha(t) implies s = t or \{s, t\} = \{0, 1\}. A polygon is either a simple loop in \mathbb{R}^2 or the image of such a loop, depending on context. A subset of \mathbb{R}^2 is polygonal if it comprises the inside of a polygon together with some or all of its frontier.

I now define three important operations on paths. If \alpha is a path in X, and a, b \in I, then the path obtained by varying the argument of \alpha from a to b is the path \alpha_{a:b}: I \to X given by

$$\alpha_{a:b}(t) = \alpha((1 - t)a + tb).$$

We call \alpha_{a:b} a subpath of \alpha. If \alpha and \beta are paths in X satisfying \alpha(1) = \beta(0), then their concatenation is the path \alpha \ast \beta: I \to X equaling

$$t \mapsto \begin{cases} 
\alpha(2t), & \text{if } t \leq \frac{1}{2}; \\
\beta(2t - 1), & \text{if } t \geq \frac{1}{2}.
\end{cases}$$

Note that \((\alpha \ast \beta)_{0:1/2} = \alpha\) and \((\alpha \ast \beta)_{1/2:1} = \beta\). The reverse of a path \alpha, denoted \overset{*}{\alpha}, is \alpha_{1:0}.

Given a way to measure the length of a linear path, I define the arc length of a path \alpha to be the least upper bound of the lengths of piecewise linear approximations to \alpha. (A piecewise linear path \beta approximates \alpha if \beta(t) = \alpha(t) for each vertex \beta(t) of \beta.) If \alpha is a path in \mathbb{R}^2, the euclidean arc length of \alpha is denoted |\alpha|. The arc length of \alpha in an arbitrary norm \| \cdot \| is denoted \|\alpha\|. One reason for using norms rather than arbitrary metrics is to make the arc length of a linear path equal the distance between its endpoints: in any norm \| \cdot \| we have \|p \cdot q\| = \|p - q\|. A path \alpha of finite arc length is canonical if \|\alpha_{0:t}\| = t \cdot |\alpha| for every t \in (0, 1].

Geometric primitives

Because piecewise linear paths are central to this work, we need a few more definitions relating to them. Some of these definitions supersede less precise definitions given in Section 1D. Let \alpha be a piecewise linear path. A joint of \alpha is a point s \in (0, 1) such that for no open interval (x, y) containing s is the subpath \alpha_{x:y} linear. A segment of \alpha is a subpath \alpha_{s:t} with s < t such that each of s and t is a joint of \alpha or a point in \{0, 1\}. Now let \alpha_{r:s} and \alpha_{s:t} be consecutive segments of a piecewise linear path \alpha: I \to \mathbb{R}^2. We say that \alpha turns at s if neither \alpha_{r:s} nor \alpha_{s:t} is constant, and \alpha(s) does not lie on the linear path \alpha(r) \cdot \alpha(t). If \alpha turns at s,
then the rays from $\alpha(s)$ through $\alpha(r)$ and $\alpha(t)$ form an angle of measure less than $\pi$ (and perhaps of measure 0). The path $\sigma$ turns away from a point $x \in \mathbb{R}^2$ at $s$ if $x$ is exterior to this angle, and otherwise $\sigma$ turns toward $x$ at $s$.

Whenever a path has two "sides", it makes sense to talk about another path crossing over it. And at least in the neighborhood of any point on a nonconstant segment, every piecewise linear path does have two sides. We say that $\alpha$ crosses over $\beta$ at a point $x \in I$ if there is an interval $[s, t]$ containing $x$ such that $\text{Im} \, \alpha_{s,t} \subset \beta$ but the paths $\alpha_{0,s}$ and $\alpha_{1,t}$ approach $\beta$ from opposite sides.

2A. Homotopies and the Fundamental Group

The notion of a rough routing for a wire is rooted in the mathematical idea of path homotopy. Hence in the of study planar wiring problems involving rough routings, we look first at the homotopy theory of paths. This section defines the appropriate notions of homotopy for paths and general maps, gives a precise definition of simple connectivity, and provides several methods for proving that a space is simply connected.

Path homotopy

Roughly speaking, two paths are path homotopic if one can be continuously deformed into the other without moving its endpoints. One can make this notion precise by expressing the continuous deformation as a continuous function.

**Definition 2a.1.** Two paths $\alpha, \beta : I \to Y$ are path homotopic, denoted $\alpha \simeq_P \beta$, if there is a map $F : I \times I \to Y$ such that $F(\cdot, 0) = \alpha$, $F(\cdot, 1) = \beta$, and the maps $F(0, \cdot)$ and $F(1, \cdot)$ are constant. The map $F$ is called a path homotopy between $\alpha$ and $\beta$.

A path homotopy $F$ defines a family of paths $\{F(\cdot, t) : t \in I\}$ with the same endpoints. As $t$ varies from 0 to 1, the path $F(\cdot, t)$ varies in a continuous manner. A good example of a path homotopy is given by the following lemma.

**Lemma 2a.2.** For every path $\alpha$ and all points $a, b, c \in I$ we have

$$\alpha_{a:b} \ast \alpha_{b:c} \simeq_P \alpha_{a:c}.$$  

**Proof.** A path homotopy between $\alpha_{a:b} \ast \alpha_{b:c}$ and $\alpha_{a:c}$ is the map $H$ defined by $H(\cdot, t) = \alpha_{a:h(t)} \ast \alpha_{h(t):c}$ where $h(t) = (1 - t)b + t(a + c)/2$.  

\[ -65 - \]
The relation of path homotopy is an equivalence relation, as one can check directly. To prove that \( \alpha \simeq_P \beta \) implies \( \beta \simeq_P \alpha \), for example, it suffices to note that if \( F \) is a path homotopy between \( \alpha \) and \( \beta \), then the map \( (s, t) \mapsto F(s, 1 - t) \) is a path homotopy between \( \beta \) and \( \alpha \). The equivalence class of a path \( \alpha \) under path homotopy is denoted \([\alpha]_P\), and is called the path class of \( \alpha \).

**The fundamental group**

We now define a concatenation operation for path classes. Path concatenation respects path homotopy, in the sense that if \( \alpha \simeq_P \gamma \) and \( \beta \simeq_P \delta \), then \( \alpha \ast \beta \simeq_P \gamma \ast \delta \). Thus the concatenation \([\alpha]_P \ast [\beta]_P\) of two path classes is well defined by setting \([\alpha]_P \ast [\beta]_P = [\alpha \ast \beta]_P\). The important properties of this operation are listed below; they can be derived from Lemma 2a.2.

1. **Associativity:** \([\alpha]_P \ast ([\beta]_P \ast [\gamma]_P) = ([\alpha]_P \ast [\beta]_P) \ast [\gamma]_P\) whenever these expressions are defined.

2. **Existence of identities:** \([\alpha]_P \ast [\alpha_{1,1}]_P = [\alpha]_P = [\alpha_{0,0}]_P \ast [\alpha]_P\) for any path class \([\alpha]_P\). Thus \([\alpha_{1,1}]_P\) and \([\alpha_{0,0}]_P\) are right and left identities, respectively, for \([\alpha]_P\).

3. **Existence of inverses:** \([\alpha]_P \ast [\hat{\alpha}]_P = [\alpha_{0,0}]_P\) and \([\hat{\alpha}]_P \ast [\alpha]_P = [\alpha_{1,1}]_P\) for any path class \([\alpha]_P\). Thus \([\hat{\alpha}]_P\) is both a left and right inverse for \([\alpha]_P\); its own inverse is \([\alpha]_P\), since the reverse of \(\hat{\alpha}\) is \(\alpha\).

Equations (1) through (3) are called the groupoid properties of concatenation. They would make concatenation a group operation, except that the concatenation of two paths is not always defined. To obtain a group we need only restrict ourselves to paths that begin and end at a specific point.

**Definition 2a.3.** Let \( x_0 \) be a point of the space \( X \). A path in \( X \) whose endpoints coincide at \( x_0 \) is called a loop at \( x_0 \). The fundamental group of \( X \) at \( x_0 \), denoted \( \pi_1(X, x_0) \), is the set of path classes of loops at \( x_0 \), under the operation of concatenation.

The identity element of \( \pi_1(X, x_0) \) is the class \([t \mapsto x_0]_P\) of the constant loop at \( x_0 \). A loop at \( x_0 \) is called inessential if it falls in this class, and essential otherwise.

A natural question to ask is how the fundamental group \( \pi_1(X, x_0) \) depends on the choice of base point \( x_0 \). For a path-connected space, the answer is that the fundamental groups at different base points are isomorphic. To see why, let \( \alpha \) be a path in \( X \) from \( x \) to \( y \), and consider the map \( h_\alpha: \pi_1(X, x) \to \pi_1(X, y) \) defined by

\[
h_\alpha([\gamma]_P) = [\hat{\alpha} \ast \gamma \ast \alpha]_P.
\]
This map is a group homomorphism; one simply computes, using the groupoid properties of concatenation, that

\[ h_\alpha([\gamma]_P * [\delta]_P) = [\widehat{\alpha} * (\gamma * \delta) * \alpha]_P \]
\[ = [(\widehat{\alpha} * \gamma * \alpha) * (\widehat{\alpha} * \delta * \alpha)]_P \]
\[ = h_\alpha([\gamma]_P) * h_\alpha([\delta]_P). \]

Furthermore, if \( \beta = \widehat{\alpha} \), then \( h_\beta \) and \( h_\alpha \) are inverses, so \( h_\alpha \) is actually an isomorphism.

**Definition 2a.4.** A space \( X \) is simply connected if \( X \) is path-connected and \( \pi_1(X, x_0) \) equals 0, the trivial group, for some point \( x_0 \in X \).

Because all the fundamental groups of a path-connected space are isomorphic, a simply connected space has trivial fundamental group at every point. In other words, every loop in a simply connected space is inessential. As a consequence we deduce a very useful property of simply connected spaces.

**Lemma 2a.5.** In a simply connected space, any two paths having the same initial and final points are path-homotopic.

**Proof.** Let \( X \) be a simply connected space, and let \( \alpha \) and \( \beta \) be two paths in \( X \) from \( x \) to \( y \). Then \( \beta * \widehat{\alpha} \) is a loop at \( x \), and because \( \pi_1(X, x) \) is trivial, we have \( \beta * \widehat{\alpha} \simeq_P \beta_{0,0} \). Hence by the groupoid properties of concatenation,

\[ [\beta]_P = [\beta * \beta_{1,1}]_P = [\beta * (\widehat{\alpha} * \alpha)]_P = [(\beta * \widehat{\alpha}) * \alpha]_P = [\beta_{0,0} * \alpha]_P = [\alpha]_P. \]

Therefore \( \beta \) is path-homotopic to \( \alpha \). \( \square \)

**Induced homomorphisms**

Not only can we associate with each space a fundamental group, but to each map between spaces we can associate a homomorphism between the corresponding fundamental groups. Suppose \( f: X \to Y \) is a map of topological spaces, and suppose \( f(x_0) = y_0 \). We usually express this fact by writing \( f: (X, x_0) \to (Y, y_0) \). Then \( f \) induces a homomorphism

\[ f_*: \pi_1(X, x_0) \to \pi_1(Y, y_0), \]

defined by \( f_*([\alpha]_P) = [f \circ \alpha]_P \). This map is well defined, because if \( \alpha \) is a loop at \( x_0 \), then \( f \circ \alpha \) is a loop at \( y_0 \); if \( \beta \) is another path in \( X \), and \( H \) is a path homotopy between \( \alpha \) and \( \beta \), then \( f \circ H \) is a path homotopy between the paths \( f \circ \alpha \) and \( f \circ \beta \).

The map \( f_* \) is a group homomorphism because \( f \circ (\alpha * \beta) = (f \circ \alpha) * (f \circ \beta) \), which implies

\[ f_*([\alpha * \beta]_P) = f_*([\alpha]_P) * f_*([\beta]_P). \]
The correspondence between maps of spaces and homomorphisms of fundamental groups is actually a "functor", which means that it has the following functorial properties:

1. It commutes with composition, that is, \((g \circ f)_* = g_* \circ f_*\).
2. It takes identity maps to identity maps, so \(id_* = id\).

One important consequence of these properties is that the fundamental group of a space is a topological invariant, meaning that homomorphisms preserve it. For suppose that \(f: (X, x_0) \to (Y, y_0)\) is a homeomorphism with inverse \(g: (Y, y_0) \to (X, x_0)\). Then the maps \(g_*\) and \(f_*\) are inverses: by property (1), we have \(f_* \circ g_* = (f \circ g)_* = id_*\), which by property (2) is the identity homomorphism on \(\pi_1(Y, y_0)\); similarly \(g_* \circ f_* = (g \circ f)_*\) is the identity on \(\pi_1(X, x_0)\). Therefore \(f_*\) gives an isomorphism between the fundamental groups of \(X\) (at \(x_0\)) and \(Y\) (at \(y_0\)).

Similar reasoning shows that if \(A\) is a retract of \(X\), and \(x_0 \in A\), then the map \(i_*: \pi_1(A, x_0) \to \pi_1(X, x_0)\) induced by the inclusion \(i: (A, x_0) \to (X, x_0)\) is a monomorphism (one-to-one). For if \(r: (X, x_0) \to (A, x_0)\) is the retraction, then \(r \circ i = id_A\), whence \(r \circ i_*\) is the identity on \(\pi_1(A, x_0)\). Since \(\text{Ker} i_* \subseteq \text{Ker}(r_* \circ i_*) = 0\), the kernel of \(i_*\) is trivial. As a corollary, every retract of a simply connected space is simply connected.

**Homotopy of general maps**

There are many types of homotopy relations, path homotopy being only one of them. As we are concerned primarily with homotopy among paths and loops, the following results will be used mainly for proving spaces to be simply connected.

**Definition 2a.6.** Let \(X\) and \(Y\) be topological spaces, and let \(A \subseteq X\). Two maps \(f, g: X \to Y\) are homotopic relative to \(A\), written \(f \simeq g \text{ rel } A\), if there is a map \(F: X \times I \to Y\) such that \(F(\cdot, 0) = f\), \(F(\cdot, 1) = g\), and \(F|_{AXI} = id_{AXI}\). If \(A = \emptyset\), we simply write \(f \simeq g\). The map \(F\) is a homotopy between \(f\) and \(g\).

Though the concept of homotopy seems to apply only to maps, it can tell us something about a space when applied to the identity map on the space. A subspace \(A\) of a space \(X\) is a deformation retract if there is a retraction \(r: X \to A\) such that \(id_X \simeq i \circ r \text{ rel } A\), where \(i: A \to X\) is the inclusion map. The homotopy between \(r\) and \(id_X\) is called a deformation retraction. The fundamental group of a deformation retract satisfies an even stronger property than that of a retract.

**Lemma 2a.7.** If \(A\) is a deformation retract of \(X\), then the inclusion \(i: (A, x_0) \to (X, x_0)\) induces an isomorphism of fundamental groups.

**Proof.** Because \(A\) is a retract of \(X\), the map \(i_*: \pi_1(A, x_0) \to \pi_1(X, x_0)\) is a monomorphism. It remains to show that \(i_*\) is an epimorphism (onto). Let \(\beta\) be a loop at
Section 2B  

Covering Spaces

We prove that \([\beta]_P\) is in the image of \(i_*\) by applying the deformation retraction to \(\beta\). Let \(F: X \times I \to X\) be a deformation retraction of \(X\) to \(A\), and define a map \(G: I \times I \to X\) by \(G(s, t) = F(\beta(s), t)\). Then \(G\) is a path homotopy, since for \(e \in \{0, 1\}\), the point \(G(e, t) = F(\beta(0), t) = F(x_0, t) = x_0\) (because \(F\) is the identity on \(A \times I\)). Moreover, \(F\) is a homotopy between \(\beta\) and a loop \(\alpha: I \to X\) whose image lies in \(A\): we have \(G(\cdot, 0) = F(\alpha(\cdot), 0) = \beta\), and \(G(\cdot, 1) \subseteq F(X, 1) \subseteq A\). Therefore \(\beta \simeq_P \alpha\). Let \(\alpha': I \to A\) be the path \(t \mapsto \alpha(t)\) in \(A\). Then \([\alpha']_P \in \pi_1(A, x_0)\), and 

\[i_*([\alpha']_P) = [i \circ \alpha']_P = [\alpha]_P.\]

Since \([\alpha]_P = [\beta]_P\), this means \([\beta]_P \in \text{Im} i_*\). □

Lemma 2a.7 gives us one way to show that a space is connected. Say that a space \(X\) is contractible if some point of \(X\) is a deformation retract of \(X\). As an example, any starlike or convex subset of \(\mathbb{R}^n\) is contractible. For if \(X \subseteq \mathbb{R}^n\) contains a point \(z\) such that the line segment \(\overline{zx}\) lies in \(X\) whenever \(x\) does, then the map \(F: X \times I \to X\) defined by \(F(x, \cdot) = x \circ z\) is a deformation retraction of \(X\) to \(z\). We call it a contraction of \(X\) to \(z\).

Lemma 2a.8. Every contractible space is simply connected.

Proof. Let \(F: X \times I \to X\) be a contraction of \(X\) to the point \(z \in X\). Then \(X\) is path-connected because any point \(x \in X\) can be joined to \(z\) by the path \(\rho_x = F(x, \cdot)\); for any two points \(x, y \in X\), the concatenation \(\rho_x \ast \rho_y\) is a path between \(x\) and \(y\). Because \(z\) is a deformation retract of \(x\), the previous lemma shows that \(X\) and \(z\) have isomorphic fundamental groups. There is only one path in \(z\), so \(\pi_1(z, z)\) is trivial. Hence \(\pi_1(X, z) = 0\) also. □

Extension lemma

We conclude the section with a criterion for a loop to be inessential.

Lemma 2a.9. Let \(f: Fr(I \times I) \to X\) be any map, and let \(\delta\) be the loop

\[\delta = (\cdot, 0) \ast (1, \cdot) \ast (\cdot, 1) \ast (0, \cdot): I \to I \times I.\]

The loop \(f \circ \delta\) is inessential if and only if \(f\) has an extension \(F: I \times I \to X\). □

2B. Covering Spaces

In order to compute the fundamental groups of spaces that are not simply connected, one usually introduces the notion of a covering space. As we shall see, the fundamental group of the circle \(S^1\) is easily determined using this device. But I introduce covering spaces for a different reason. In essence, a simply connected covering space provides a spatial representation of path homotopy classes. It thereby
Topological Preliminaries

converts problems involving homotopy constraints, such as my single-layer wire routing problems, into problems without homotopy constraints.

This section provides a very brief introduction to the theory of covering spaces. It defines covering spaces and the notion of lifting to a covering space, and proves the important theorem that the lifting of a map is unique if the lifting is determined at a single point. It then gives conditions for a map to be liftable, and notes that paths and homotopies of paths can always be lifted. Some applications of lifting are also presented. Finally, it states some fairly mild conditions under which a space has a simply connected covering space, and shows that in the presence of those conditions, that covering space is essentially unique.

Definition of covering space

**Definition 2b.1.** Let \( p: M \to X \) be a surjective map. An open set \( U \) in \( X \) is **evenly covered** by \( p \) if \( p^{-1}(U) \) can be partitioned into disjoint open sets, each of which is mapped homeomorphically onto \( U \) by \( p \). If every point of \( X \) has a neighborhood that is evenly covered by \( p \), then \( p \) is called a **covering map**, and \( M \) a **covering space** of \( X \).

A covering space is often called simply a **cover**; the space it covers is called the **base** space. Locally, a covering space looks like a union of disjoint copies of the base. It follows immediately that a covering map is a local homeomorphism. For suppose that \( p: M \to X \) is a covering map, and let \( v \) be a point of \( M \). Take \( U \) to be a neighborhood of \( p(v) \) that is evenly covered by \( p \), and partition \( p^{-1}(U) \) into disjoint open sets that are mapped homeomorphically onto \( U \) by \( p \). One of these open sets, call it \( V \), contains \( v \). Then \( V \) is a neighborhood of \( v \), and \( p|_V \) is a homeomorphism onto its image, which is open. This makes \( p \) a local homeomorphism. As a consequence, \( M \) has all the local properties that \( X \) has.

Perhaps the simplest interesting covering map is \( \theta: \mathbb{R}^1 \to S^1 \) given by

\[
\theta(t) = (\cos 2\pi t, \sin 2\pi t),
\]

which maps the real line onto the circle. We show that every point of \( S^1 \) is evenly covered by \( \theta \). Let \( s_0 \) represent the point \((1,0)\) of \( S^1 \). Then \( S^1 - s_0 \) is a neighborhood of every point of \( S^1 \) but \( s_0 \), and is evenly covered by \( \theta \). For \( \theta^{-1}(S^1 - s_0) \) is \( \mathbb{R}^1 - Z \), which is the disjoint union of the open intervals \( \{(n,n+1): n \in \mathbb{Z}\} \), and each of these intervals is mapped homeomorphically onto \( S^1 - s_0 \) by \( \theta \). Similarly, the neighborhood \( S^1 - (-1,0) \) of \( s_0 \) is evenly covered by \( \theta \). Therefore \( \theta \) is a covering map. A related covering map, pictured in Figure 2b-1, is \( \theta \times id_I: \mathbb{R}^1 \times I \to S^1 \times I \). This map can be thought of as compressing an infinite helical surface in \( \mathbb{R}^3 \) onto an annular region in \( \mathbb{R}^2 \).
Figure 2b-1. Lifting to a covering space. The helical surface, which extends infinitely in both directions, is a simply connected covering space for the annular region below. The covering map is downward projection. Also shown is a path $\alpha$ in the annulus and one of its liftings $\tilde{\alpha}$ to the covering space. There is one such lifting for each point in the inverse image of $\alpha(0)$.

Lifting

One can study objects in a base space by transporting those objects to some covering space. If $p: M \to X$ is a covering map, and $g: Y \to X$ is a map into $X$, a lifting, or lift, of $g$ is a map $\tilde{g}: Y \to M$ satisfying $p \circ \tilde{g} = g$. For example, if $g$ is a path in $X$, then $\tilde{g}$ is a path in $M$ that "sits over" $g$. (See Figure 2b-1.)

**Theorem 2b.2.** (Uniqueness of Liftings) Two liftings of a map from a connected space are equal if they agree at one point.

**Proof.** Let $p: M \to X$ be a covering map, and let $g, g': Y \to M$ be two liftings of a map $f: Y \to X$. Let $Y_\subseteq$ be the set of points in $Y$ at which $g$ and $g'$ agree, and let $Y_\neq$ be its complement. To prove the theorem, it suffices to show that $Y_\subseteq$ and $Y_\neq$ are both open in $Y$. For if $Y$ is connected, it follows that either $Y_\subseteq$ or $Y_\neq$ is empty.
Figure 2b-2. A complicated covering space. The surface pictured above is part of the simply connected covering space for a disk with two circular holes removed. The "layers" of this covering space are indexed by the free group on two generators (which is also the fundamental group of the base space). Only a few layers are shown.
Let \( y \) be a point of \( Y \), and choose a neighborhood \( U \subseteq X \) of \( f(y) \) that is evenly covered by \( p \). Because \( f \) is continuous, there is a neighborhood \( V \) of \( y \) such that \( f(V) \subseteq U \). We may assume that \( V \) is connected. Now let \( W \) be the component of \( p^{-1}(U) \) that contains \( g(y) \). Because \( g(V) \) is connected, and \( g(V) \subseteq p^{-1} \circ f(V) \subseteq p^{-1}(U) \), the set \( g(V) \) must lie entirely within \( W \). Similarly, if \( W' \) is the component of \( p^{-1}(U) \) that contains \( g'(y) \), then \( g'(V) \subseteq W' \). If \( y \in Y_\neq \), then \( W \cap W' = \emptyset \), so \( g(V) \cap g'(V) = \emptyset \) and therefore \( V \) is a neighborhood of \( y \) in \( Y_\neq \). If instead \( y \in Y_\eq \), then \( W = W' \), whence \( g(v) = g'(v) \) for all \( v \in V \) because \( p \circ g = p \circ g' \) and \( p|_W \) is injective. In this case, \( V \) is a neighborhood of \( y \) in \( Y_\eq \). Thus \( Y_\eq \) and \( Y_\neq \) are both open in \( Y \). \( \square \)

A natural question is: When can a map be lifted to a covering space? The following theorem gives a complete answer to this problem for a large class of spaces. Recall that a space is locally path-connected if it has a basis of path-connected sets. For example, any convex subspace \( X \subseteq \mathbb{R}^n \) is locally path-connected, because every open ball in \( X \) is convex and hence path-connected.

**Theorem 2b.3.** (Lifting Theorem) Let \( p: (M, m_0) \rightarrow (X, x_0) \) be a covering map, and let \( Y \) be connected and locally path-connected. The map \( g: (Y, y_0) \rightarrow (X, x_0) \) has a lifting \( \tilde{g}: (Y, y_0) \rightarrow (M, m_0) \) if and only if

\[
g_* (\pi_1(Y, y_0)) \subseteq p_* (\pi_1(M, m_0)). \quad \square
\]

In particular, the conclusion always holds if \( Y \) is simply connected and locally path-connected, for then \( Y \) is connected and \( \text{Im} \ g_* \) is trivial. For each point \( m_0 \in p^{-1}(x_0) \), the map \( g: (Y, y_0) \rightarrow (X, x_0) \) can then be lifted in such a way that \( \tilde{g}(y_0) = m_0 \). In particular, every path \( \alpha: I \rightarrow X \) and every path homotopy \( F: I \times I \rightarrow X \) can be so lifted: the spaces \( I \) and \( I \times I \) are convex, and hence locally path-connected, contractible, and simply connected (Lemma 2a.8). Actually, the proof of the lifting theorem requires these facts, and the following proposition as well. The proof of the proposition, although it relies on the lifting of path homotopies, is nevertheless instructive.

**Proposition 2b.4.** Liftings of homotopic paths are path-homotopic if they agree at one endpoint.

**Proof.** Let \( p: M \rightarrow X \) be a covering map, and let \( \tilde{\alpha} \) and \( \tilde{\gamma} \) be paths in \( M \) whose projections \( \alpha = p \circ \tilde{\alpha} \) and \( \gamma = p \circ \tilde{\gamma} \) are path homotopic. Let \( F: I \times I \rightarrow X \) be a path homotopy between \( \alpha \) and \( \gamma \), and suppose that \( \tilde{\alpha}(e) = \tilde{\gamma}(e) \) where \( e \) is either 0 or 1. Choose the base points \((e, 0), (e, 0), \) and \( (e, 0) \) for \( I \times I, X, \) and \( M \) respectively. Then \( F \) lifts to a map \( \tilde{F}: I \times I \rightarrow M \) satisfying \( \tilde{F}(e, 0) = \tilde{\alpha}(e) \). I claim that \( \tilde{F} \) is a path homotopy between \( \tilde{\alpha} \) and \( \tilde{\gamma} \).
• \( \tilde{F}(e, \cdot) \) is constant. Both \( t \mapsto \tilde{F}(e, t) \) and \( t \mapsto \tilde{\alpha}(e) \) are liftings of the constant path \( t \mapsto F(e, t) \), and they agree at \( t = 0 \). The interval \( I \) is connected, so by uniqueness of liftings (Theorem 2b.2), the liftings are identical. In particular, \( \tilde{F}(e, 1) = \tilde{\alpha}(e) = \tilde{\gamma}(e) \).

• \( \tilde{F}(\cdot, 0) = \tilde{\alpha} \). Both \( \tilde{\alpha} \) and \( \tilde{F}(\cdot, 0) \) lift \( \alpha \), because \( p \circ \tilde{F}(\cdot, 0) = F(\cdot, 0) = \alpha \), and the two liftings agree at \( e \). By (2b.2) again, they must be the same map.

• \( \tilde{F}(\cdot, 1) = \tilde{\gamma} \). Both \( \tilde{\gamma} \) and \( \tilde{F}(\cdot, 1) \) lift \( \gamma \), and they agree at \( e \).

• \( \tilde{F}(1 - e, \cdot) \) is constant. Both \( t \mapsto \tilde{F}(1 - e, t) \) and \( t \mapsto \tilde{F}(1 - e, 0) \) lift the constant path \( t \mapsto F(1 - e, \cdot) \). Because the liftings agree when \( t = 0 \), they must be equal.

Armed with this lemma and the covering map \( \theta : (R^1, 0) \to (S^1, s_0) \), the reader should be able to prove the following result. (Hint: lift each loop \( \alpha \) at \( s_0 \) to a path \( \tilde{\alpha} \) beginning at \( 0 \), and consider \( \tilde{\alpha}(1) \).)

**Proposition 2b.5.** The fundamental group of the circle is isomorphic to the integers under addition.

### Existence and uniqueness of covering spaces

What makes the proof of Proposition 2b.5 work is that the circle has a simply connected covering space (the real line) with a natural group operation (addition). Most spaces do not come with as nice a covering space as \( R^1 \). Nevertheless, simply connected covering spaces can often be constructed out of the space of paths in the base space. The following theorem gives sufficient conditions for this construction to work. The conditions may look scary, but in fact they are satisfied by almost every decent space. We say that a space \( X \) is **semilocally simply connected** if every point \( x \in X \) has a neighborhood \( U \) such that the map \( i_* : \pi_1(U, x) \to \pi_1(X, x) \) induced by the inclusion \( i : U \to X \) is trivial. Of course, this condition holds if \( U \) is simply connected.

**Theorem 2b.6.** Every connected, locally path-connected, semilocally simply connected space has a simply connected covering space. □

There is a notion of equivalence among covering spaces of a fixed base space. This notion is stronger than that of homeomorphism, because it also requires that the correspondence between the covering spaces respect the covering maps. More specifically, if \( p : M \to X \) and \( q : N \to X \) are covering maps, then \( M \) and \( N \) are **equivalent** if there are inverse maps \( f : M \to N \) and \( g : N \to M \) such that \( q \circ f = p \) and \( p \circ g = q \). Equivalent covering spaces are topologically indistinguishable. The following proposition shows that the simply connected cover of a decent space is essentially unique.
Proposition 2b.7. All simply connected covering spaces of a connected, locally path-connected space are equivalent.

Proof. (Lift the covering maps.) Let $X$ be connected and locally path-connected, let $p: (M, m_0) \to (X, x_0)$ and $q: (N, n_0) \to (X, x_0)$ be covering maps, and suppose that $M$ and $N$ are simply connected. The maps $p$ and $q$ are local homeomorphisms, so $M$ and $N$ have all the local properties that $X$ has. In particular, $M$ and $N$ are locally path-connected. We now apply the Lifting Theorem (2b.3), lifting $p$ to a map $\tilde{p}: (M, m_0) \to (N, n_0)$, and also lifting $q$ to a map $\tilde{q}: (N, n_0) \to (M, m_0)$. By the definition of lifting, $p \circ \tilde{q} = q$ and $q \circ \tilde{p} = p$. I claim that $\tilde{p}$ and $\tilde{q}$ are inverses, making $M$ and $N$ homeomorphic. Because $p \circ \tilde{q} \circ \tilde{p} = q \circ \tilde{p} = p$, we find that $\tilde{q} \circ \tilde{p}$ is a lift of the map $p$. But $id_M$ also lifts $p$, and because $\tilde{q} \circ \tilde{p}(x_0) = \tilde{q}(y_0) = x_0$, the two maps agree at the point $x_0$. Since $M$ is connected, they must be identical, by Theorem 2b.2. Entirely symmetrical reasoning shows that $id_Y = \tilde{p} \circ \tilde{q}$. □

Covering transformations

Proposition 2b.7 not only shows that the simply connected cover of a decent space is unique, but also implies that this cover must be highly symmetrical. If $p: M \to X$ is a covering map, a homeomorphism $h: M \to M$ that lifts $p$ is called a covering transformation of $M$. Suppose that $M$ is simply connected and $X$ is locally path-connected. For any two points $m_0, m_1 \in M$ that have the same image $x_0$ under $p$, Proposition 2b.7 gives us a covering transformation $h: M \to M$ such that $h(m_0) = m_1$. (Consider the covering maps $p: (M, m_0) \to (X, x_0)$ and $p: (M, m_1) \to (X, x_0)$.) Hence different lifts of the same path or homotopy are related by a covering transformation. This fact allows us to ignore the base point of the covering space; all base points are equivalent.

We conclude this section with another simple application of the Lifting Theorem. It shows how one can lift subspaces as well as maps.

Lemma 2b.8. Let $p: M \to X$ be a covering map, and let $C$ be a simply connected, locally path-connected subspace of $X$. For every path component $A$ of $p^{-1}(C)$, the map $p: A \to C$ is a homeomorphism.

Proof. Let $c$ be a point of $C$, and pick $a \in A \subseteq p^{-1}(C)$. Lift the identity map on $C$ to a map $i: (C, c) \to (p^{-1}(C), a)$. Because $C$ is path-connected, so is $i(C)$, and hence $i(C) \subseteq A$. We have $p \circ i = id_C$, and it remains to show $i \circ p = id_A$. The map $i \circ p: (A, a) \to (A, a)$ lifts $p: (A, a) \to (C, c)$, because $p \circ (i \circ p) = (p \circ i) \circ p = id_C \circ p$. Hence $id_A$ and $i \circ p$ are two liftings of $p: A \to C$, and they satisfy $id_A(a) = a = i \circ p(a)$. Therefore $id_A = i \circ p$ by uniqueness of liftings. □


2C. Paths and Loops in the Plane

This section collects miscellaneous results concerning the topology of subsets of the plane. Most are of the "obvious but nontrivial" variety, like the fact that a simple loop is inessential in a subspace of the plane if and only if that subspace includes the inside of the loop. These results can be justified using standard topological methods. One result, however, we derive from a theorem of real analysis. Our result says: Within any nonempty family of canonical paths in a bounded subspace of the plane, there exists a sequence of paths converging to a path whose euclidean arc length is no greater than that of any path in the family. We use this result in proving the existence of things like rubber bands and ideal routes, which are defined as the minimum-length paths in certain families.

**Facts about polygons**

Geometric topology, the study of topology within euclidean spaces, is another source of mathematical insight into single-layer wire routing problems. In these problems the routing region—a subspace of the plane—has important topological properties that one often takes for granted. Perhaps the most famous of these is the Jordan Curve Theorem, stated below for the case of piecewise linear loops.

**Theorem 2c.1. (Jordan Curve Theorem)** If \( \lambda \) is a simple loop in \( \mathbb{R}^2 \), then \( \mathbb{R}^2 - \im \lambda \) has two components, one bounded and one unbounded, whose common frontier is \( \im \lambda \).

The bounded component is called the inside of the loop, and the other component is called the outside. A kind of converse to Theorem 2c.1 is the following.

**Lemma 2c.2.** Let \( X \subset \mathbb{R}^2 \) be a finite union of polygonal regions. If \( A \) is a bounded component of \( \mathbb{R}^2 - X \), then there is a simple loop in \( X \) whose inside contains \( A \).

We shall also need the following results.

**Proposition 2c.3.** If \( \lambda \) is a simple loop in \( \mathbb{R}^2 \), then \( \im \lambda \) is a retract of \( \mathbb{R}^2 - \text{inside}(\lambda) \).

**Proposition 2c.4.** Let \( \lambda \) and \( \mu \) be simple loops in \( \mathbb{R}^2 \). If \( \im \mu \subset \text{inside}(\lambda) \), then \( \im \mu \) is a deformation retract of \( \mathbb{R}^2 - \text{inside}(\mu) - \text{outside}(\lambda) \).

Theorems like these belong to geometric topology, and are somewhat messy to prove rigorously, even when stated (as here) for piecewise linear objects only. Unfortunately, I have no reference for these particular results, though they follow from well-known properties of polygons. One reference for geometric topology is [35].
Enclosure

Another intuitive property of planar loops is this: A simple loop $\lambda$ whose inside contains a point $p$ cannot be deformed so that $p$ ends up outside, except by crossing over $p$. More formally, if $p$ lies inside $\lambda$, and if the simple loop $\mu$ is loop-homotopic to $\lambda$ in $R^2 - p$, then $p$ lies inside $\mu$.

We can obtain a more general result by extending the notion of “inside” to loops that are not simple. Say a loop $\lambda$ in $R^2$ encloses a connected subset $F$ of $R^2 - Im \lambda$ if $\lambda$ is essential in $R^2 - F$. Now suppose $\lambda \simeq_F \mu$ as paths in $R^2 - F$. Then either both loops are essential or both are inessential in $R^2 - F$, so $\lambda$ encloses $F$ if and only if $\mu$ does. The next proposition shows that the definition of enclosure agrees with the definition of inside for simple loops.

**Proposition 2c.5.** Let $\lambda$ be a loop in $R^2$, and let $S$ denote the space $R^2 - Im \lambda$. If $\lambda$ encloses a connected subset $F$ of $S$, then $F$ lies in a bounded component of $S$. The converse holds if $\lambda$ is simple. 

As a consequence we obtain a very intuitive result concerning simple loops.

**Corollary 2c.6.** A simple loop $\lambda$ is inessential in a subspace $S$ of $R^2$ if and only if $S$ includes the inside of $\lambda$.

**Proof.** Suppose that $S$ includes inside($\lambda$), and let $F$ denote the outside of $\lambda$. By Proposition 2c.5, $\lambda$ does not enclose $F$, which means $\lambda$ is inessential in $R^2 - F$. Hence $\lambda$ is inessential in the larger space $S$. Now suppose that $S$ does not include inside($\lambda$), and let $F$ be a component of $R^2 - S$ lying inside $\lambda$. By Proposition 2c.5 again, $\lambda$ encloses $F$, which means that $\lambda$ is essential in $R^2 - F$. Hence $\lambda$ is essential in the smaller space $S$.

An elementary property of enclosure is the following: If $\lambda$ and $\mu$ are loops based at the same point, and $\lambda$ does not enclose $F$, then $\lambda * \mu$ encloses $F$ if and only if $\mu$ does. More generally, suppose $\lambda_1, \ldots, \lambda_n$ are loops based at the same point, and let $F$ be a connected set that intersects none of them. The concatenated loop $\lambda_1 * \cdots * \lambda_n$ can enclose $F$ only if some $\lambda_i$ does, and it does enclose $F$ if exactly one $\lambda_i$ does.

**Piecewise linearity**

When dealing with piecewise linear paths, we shall often want our homotopies to be piecewise linear also. A map $F$ from $I \times I$ into a linear space is piecewise linear if $I \times I$ can be divided into triangles so that $F$ is linear on each triangle. If $F$ is piecewise linear and $\alpha$ is any linear path in $I \times I$, then $F \circ \alpha$ is a piecewise linear path. The following lemma allows us to assume, in many cases, that our homotopies are piecewise linear.

\begin{center}
- 77 -
\end{center}
Theorem 2c.7. Let $S \subset R^2$ be the union of finitely many triangles, and let $F: I \times I \to S$ be a homotopy. If the paths $F(0, \cdot)$, $F(1, \cdot)$, $F(\cdot, 0)$, and $F(\cdot, 1)$ are piecewise linear, then there is a piecewise linear map $G: I \times I \to S$ that agrees with $F$ on $Fr I \times I$.

Outline of proof. (For those familiar with simplicial complexes.) One first constructs a triangulation of $S$ and a triangulation of $I \times I$ such that the map $F|_{Fr I \times I}$ is simplicial. Then one applies the Simplicial Approximation Theorem to $F$. The result is a piecewise linear map $G$ that agrees with $F$ on $Fr I \times I$. □

Minimization of arc length

One idea that was already put to use in Chapter 1 is the construction of the minimum-length path satisfying some condition. The following proposition gives us a tool for constructing a minimum-length path as a limit of other paths. It relies on a classical theorem from topology and real analysis called Ascoli's Theorem.

Proposition 2c.8. Let $\Lambda$ be a nonempty family of canonical paths in a bounded subspace $S$ of $R^2$, and put $l = \inf \{ |\lambda| : \lambda \in \Lambda \}$. Then $\Lambda$ includes a uniformly convergent sequence of paths whose limit has euclidean arc length at most $l$.

Proof. Let $\Delta = (\delta_n)_{n=1}^\infty$ be a sequence of paths in $\Lambda$ whose euclidean arc lengths converge to $l$. We use Ascoli's theorem, taken from [46], to show that the sequence $(\delta_n)$ has a convergent subsequence.

Definition: Let $\Phi$ be a family of functions from a space $X$ to a metric space $Y$ with metric $\sigma$. The family $\Phi$ is equicontinuous if for every point $x \in X$ and every $\epsilon > 0$, there is a neighborhood $N$ of $x$ such that $\sigma[f(x), f(y)] < \epsilon$ for all $y \in N$ and all $f \in \Phi$.

Theorem: (Ascoli's Theorem) Let $\Phi$ be an equicontinuous family of functions from a separable space $X$ to a metric space $Y$. Let $(f_n)$ be a sequence in $\Phi$ such that for each $x \in X$ the closure of the set $\{ f_n(x) : n > 0 \}$ is compact. Then there is a subsequence $(f_{n_k})$ that converges pointwise to a continuous function $f$, and the convergence is uniform on each compact set of $X$.

In our case, the family of functions is $\Delta$, the space $X$ is the unit interval $I$, and the space $Y$ is $S$ with the euclidean metric. We check the conditions of Ascoli's Theorem in order. Let $u$ be a bound on the arc lengths of the paths $\delta_i$. The family $\Delta$ is equicontinuous, because if $x \in I$ and $\epsilon > 0$ and $\delta \in \Delta$, every point $y$ in the open set $I \cap (x - \epsilon/2, x + \epsilon/2)$ satisfies

$$|\delta(x) - \delta(y)| \leq |\delta_{x,y}| = |y - x| \cdot |\delta| < (\epsilon/u) |\delta| \leq \epsilon.$$

The space $I$ is separable because the set of rationals in $I$ is countable and dense in $I$. Finally, the set $\{ \delta_n(x) : n > 0 \}$ lies in the bounded set $S$, which implies that its closure is compact.
We conclude that Ascoli's Theorem is applicable to the sequence \( (\delta_n) \). It yields a subsequence \( (\alpha_k) \) of \( (\delta_n) \) that converges to a path \( \alpha \). Because \( I \) is compact, the convergence is uniform.

It remains to show that \(|\alpha| \leq l\). Let \( \gamma \) be any piecewise linear approximation to \( \alpha \), and let \( \epsilon \) be any positive real number. The lengths of the paths \( \alpha_k \) converge to \( l \), so there is number \( K \) such that for all \( k > K \), we have \(|\alpha_k| < l + \epsilon\). Suppose \( \gamma \) has \( m \) segments. Because the functions \( \alpha_k \) converge uniformly to \( \alpha \), for all sufficiently large \( k \) we have \(|\alpha_k(t) - \alpha(t)| < \epsilon/m \) for all \( t \in I \). In particular, if the \( i \)th vertex of \( \gamma \) is \( \gamma(t_i) = \alpha(t_i) \), we may choose \( k \) so that \(|\alpha_k(t_i) - \alpha(t_i)| < \epsilon/m \) for all \( i \). The points \( \alpha_k(t_i) \) divide \( \alpha_k \) into small pieces that correspond to the segments of \( \gamma \). The length of the piece from \( \alpha_k(t_i) \) to \( \alpha_k(t_{i+1}) \) is at least the length of the corresponding segment of \( \gamma \), less \( 2\epsilon/m \). Summing these inequalities, we find that

\[
|\gamma| \leq |\alpha_k| + 2\epsilon < l + 3\epsilon.
\]

Because \( \epsilon \) was arbitrary, it follows that every piecewise linear approximation to \( \alpha \) has length \( l \) or less. Therefore the arc length of \( \alpha \) is at most \( l \). □

2D. Topological Manifolds

Almost all the spaces dealt with in this paper are manifolds, with or without boundary. A manifold is a very nice kind of topological space; it looks locally like \( R^n \) or \( H^n \). This section establishes the properties of manifolds that will be needed later on.

**Definition 2d.1.** If \( x \) is a point in a space \( X \), a patch about \( x \) is a homeomorphism of a neighborhood of \( x \) with an open set of \( H^n \). An **m-manifold with boundary** is a nonempty Hausdorff space in which every point has a patch. The **boundary** of an \( m \)-manifold with boundary is the set of points \( x \) having a patch \( h \) such that \( h(x) \in R^{m-1} \subset H^n \). Such a patch is called a **boundary patch**.

I will always use the term **m-manifold** to mean \( m \)-manifold with boundary. The boundary of an \( m \)-manifold \( M \), which is an \((m-1)\)-manifold, is denoted \( Bd M \). A classical theorem [39, p. 207] shows that if \( x \in M \) has a boundary patch, then every patch about \( x \) is a boundary patch.

**Theorem 2d.2.** (Invariance of Domain) Let \( U \subseteq R^n \) be open, and let \( f: U \to R^n \) be continuous and injective. Then \( f(U) \) is open in \( R^n \) and \( f \) is an embedding. □

To see how this theorem applies to manifolds, let \( h: U \to V \) and \( h': U' \to V' \) be two patches about the same point \( x \) in an \( m \)-manifold. If \( h \) is not a boundary patch, then there is a neighborhood \( W \) of \( h(x) \) in \( V \) that is open in \( R^m \). The map \( h' \circ h^{-1}|_W \)
that sends $W$ into $H^m$ is continuous and injective, and hence by Theorem 2d.2, its image is open in $R^m$. But this image contains $h'(x)$, and therefore $h'(x)$ does not lie in $R^{m-1}$. Thus $h'$ is not a boundary patch.

One can infer that $M - Bd M$ is open in $M$, for any manifold $M$. For if $x \in M - Bd M$, there is a patch $h: U \to V$ about $x$ whose image does not intersect $R^{m-1}$. This homeomorphism $h$ is a nonboundary patch for every point in $U$, and hence no point of $U$ lies in $Bd M$. But $U$ is a neighborhood of $x$. Therefore $M - Bd M$ is open. Note that $H^m$ itself is an $m$-manifold, and $Bd H^m = R^{m-1}$.

**Facts about manifolds**

Our first lemma concerns covering spaces of manifolds. If $f: M \to X$ is a map from a manifold $M$, then $Bd f$ denotes the restriction of $f$ to $Bd M$.

**Lemma 2d.3.** If $p: M \to X$ is a covering map and $X$ is an $m$-manifold, then $M$ is an $m$-manifold and $Bd p: Bd M \to Bd X$ is also a covering map.

**Proof.** Consider any point $v$ of $M$. Because $p$ is a local homeomorphism, $v$ has a neighborhood $V$ that is mapped homeomorphically by $p$ onto a neighborhood $U$ of $p(v)$. Choose a patch $h: U' \to h(U')$ about $p(v)$. Then $p$ maps the neighborhood $V \cap p^{-1}(U')$ of $v$ homeomorphically onto $U \cap U'$, which itself is homeomorphic under $h$ to the open set $h(U \cap U')$ of $H^m$. Hence $h \circ p$, when restricted to $V \cap p^{-1}(U)$, is a patch about $v$. Therefore $M$ is an $m$-manifold. Furthermore, if $p(v) \in Bd X$, then $h \circ p$ maps $v$ to a point of $Bd H^m$, so $v$ lies in $Bd M$; conversely, if $p(v) \notin Bd X$, then $h \circ p(v) \notin Bd H^m$, so $v \notin Bd M$. Therefore $p$ maps $Bd M$ onto $Bd X$.

It remains to show that the surjection $Bd p: Bd M \to Bd X$ is a covering map. Let $x$ be a point of $Bd X$, and choose a neighborhood $U$ of $x$ in $X$ that is evenly covered by $p$. Then $U \cap Bd X$ is a neighborhood of $x$ in $Bd X$, and $p^{-1}(U \cap Bd X) = p^{-1}(U) \cap Bd M$. Say $p^{-1}(U) = \bigoplus \alpha V_{\alpha}$, where the $V_{\alpha}$ are open in $M$ and $p: V_{\alpha} \to U$ is a homeomorphism for each $\alpha$. (Direct summation denotes disjoint union.) Then $p^{-1}(U \cap Bd X) = \bigoplus \alpha (V_{\alpha} \cap Bd M)$; each set $V_{\alpha} \cap Bd M$ is open in $Bd M$, and is carried to $U \cap Bd X$ by $Bd p$; and $Bd p$ restricted to $V_{\alpha} \cap Bd M$ is an embedding, because it is the restriction of the embedding $p|_{V_{\alpha}}$ to a closed subset of $V_{\alpha}$. Therefore $U \cap Bd X$ is evenly covered by $Bd p$. □

Manifolds have many wonderful properties. We shall have occasion to use only a few. The next two results are well known.

**Lemma 2d.4.** Let $M$ be a connected manifold. For every pair of points $x$ and $y$ in $M - Bd M$, there is a homeomorphism $h: M \to M$ such that $h(x) = y$, $h|_{Bd M} = id_{Bd M}$, and $h \simeq id_M$ rel $Bd M$. □

**Proposition 2d.5.** Every connected manifold has a simply connected cover.
Proof. In view of Theorem 2b.6, it suffices to show that every manifold is locally path-connected and semilocally simply connected. Let \( X \) be an \( m \)-manifold, and let \( U \) be a neighborhood of an arbitrary point \( x \in X \). We find a simply connected neighborhood of \( x \) within \( U \), which will show that \( X \) has a basis of simply connected sets. Consequently \( X \) locally simply connected, and hence both locally path-connected and semilocally simply connected.

Let \( h \) be a homeomorphism of a neighborhood \( V \) of \( x \) with an open subset of \( H^m \). Then \( h(U \cap V) \) is open in \( H^m \), so choose within this set an open ball \( B \) around \( h(x) \). Since \( B \) is convex, it is simply connected, and \( h^{-1}(B) \) is homeomorphic to \( B \). Therefore \( h^{-1}(B) \) is also simply connected, because path-connectivity and fundamental groups are topological invariants. Furthermore, \( h^{-1}(B) \) is open in \( U \cap V \), and hence in \( X \). Therefore \( h^{-1}(B) \) is a simply connected neighborhood of \( x \), contained within \( U \).

The next two lemmas are my own inventions, and although they rely on more advanced topics in algebraic topology, their proofs follow directly from standard results.

**Lemma 2d.6.** Let \( M \) be simply connected, and let \( U \) be a path-connected neighborhood of a closed subset \( X \subseteq M \). Then each path component of \( M - X \) contains exactly one path component of \( U - X \).

**Proof.** (For those who know singular homology theory.) If \( X = U \), then \( X \) is both open and closed in \( M \), whence either \( X = M \) or \( X = \emptyset \) by the connectivity of \( M \). In either case the lemma is trivial. Hence we assume \( X \subset U \), and choose a point \( u \) of \( U - X \). The couple \( \{ M - X, U \} \) is excisive because \( M - X \) and \( U \) are open sets that cover \( M \). Hence there is a relative Mayer-Vietoris sequence

\[
\cdots \to H_1(M, u) \to H_0(U - X, u) \to H_0(M - X, u) \oplus H_0(U, u) \to H_0(M, u) \to \cdots
\]

Because \( U \) and \( M \) are path-connected, the groups \( H_0(M, u) \) and \( H_0(U, u) \) are trivial. Furthermore, \( H_1(M, u) \approx H_1(M) \) is trivial because \( M \) is simply connected. Hence the sequence above takes the form

\[
0 \to H_0(U - X, u) \xrightarrow{i_*} H_0(M - X, u) \to 0.
\]

Thus the map \( i_* \), which is induced by the inclusion \( i : (U - X, u) \to (M - X, u) \), is an isomorphism. The groups \( H_0(U - X, u) \) and \( H_0(M - X, u) \) are free abelian, generated by the path components of \( U - X \) and \( M - X \), respectively, that do not contain \( u \). For each path component \( C \) of \( U - X \) with \( u \notin C \), there is a path component \( D \) of \( M - X \) that contains \( C \), and \( i_* \) maps the generator of \( H_0(U - X, u) \) corresponding to \( C \) into the generator of \( H_0(M - X, u) \) corresponding to \( D \). For \( i_* \) to be an isomorphism means that \( D \) does not contain \( u \), and no two path components
of $U - X$ are carried by $i$ into the same path component of $M - X$. Hence every path component of $U - X$, including the one that contains $u$, lies in a unique path component of $M - X$. □

**Lemma 2d.7.** Let $M$ be a simply connected, noncompact 2-manifold, and let $U$ be a neighborhood of $x \in M - Bd M$ that is homeomorphic to an open ball in $R^2$. Then every essential loop in $U - x$ is essential in $M - x$.

**Proof.** (Uses singular homology and a little homotopy theory.) The set $M - x$ is open because manifolds are Hausdorff, so $\{M - x, U\}$ is an excisive couple in $M$. Hence there is a Mayer-Vietoris sequence

$$\cdots \to H_2(M) \to H_1(U - x) \to H_1(M - x) \oplus H_1(U) \to H_1(M) \to \cdots.$$ 

Because $M$ is simply connected, $H_1(M)$ is trivial, and $H_1(U)$ is zero because $U$ is homeomorphic to a contractible space. We can also infer that $H_2(M) = 0$ from the theorem [53] that every connected, noncompact $n$-manifold satisfies $H_n(M) = 0$. (This theorem is usually stated for manifolds without boundary, but it can be extended to manifolds with boundary as follows. Let $M$ be a connected, noncompact $n$-manifold with boundary, and let $N$ be the space obtained from the disjoint union $M \oplus (Bd M \times [0,1])$ by identifying $p$ with $(p,0)$ for all $p \in Bd M$. Then $N$ is a connected $n$-manifold without boundary, and $N$ cannot be compact because $M$ is a closed subspace of $N$ that fails to be compact. Hence $H_n(N) = 0$, and $H_n(M) \approx H_n(N)$ because $M$ is a deformation retract of $N$.) Hence if $i: (U - x) \to (M - x)$ denotes the inclusion, then the Mayer-Vietoris sequence takes the form

$$0 \to H_1(U - x) \xrightarrow{i_*} H_1(M - x) \to 0.$$ 

We conclude that $i_*$ is an isomorphism.

From this we can show that the inclusion $i$ induces a monomorphism of fundamental groups. For any base point $y$ in a space $Y$, the function that sends a loop $\alpha: I \to Y$ at $y$ to the singular 1-cycle $\alpha$ (identifying $I$ with the standard 1-simplex) induces an epimorphism

$$\phi: \pi_1(Y,y) \longrightarrow H_1(Y).$$ 

The kernel of this homomorphism is the commutator subgroup of $\pi_1(Y,y)$, which vanishes if $\pi_1(Y,y)$ is abelian. Let $y$ be any point of $U - x$. The diagram

$$\begin{array}{ccc}
\pi_1(U - x, y) & \phi & H_1(U - x) \\
\downarrow i_* & & \downarrow i_* \\
\pi_1(M - x, y) & \phi & H_1(M - x)
\end{array}$$

- 82 -
commutes, as one may easily verify. Furthermore, \( U - x \) is homeomorphic to an open ball of \( \mathbb{R}^2 \) with one point removed. It follows that \( U - x \) has the homotopy type of a circle, whence \( \pi_1(U - x, y) \approx \mathbb{Z} \) is abelian. Therefore the top map in the diagram is an isomorphism. Hence \( i_* \circ \phi = \phi \circ i_* \) is an isomorphism, which makes \( i_* \) a monomorphism.

To complete the proof, suppose that \( \alpha \) is an essential loop in \( U - x \), and let \( y \) be \( \alpha(0) \). Then \( [\alpha]_P \neq 0 \) in \( \pi_1(U - x, y) \), and hence \( i_*([\alpha]_P) \neq 0 \) in \( \pi_1(M - x, y) \). But \( i_*([\alpha]_P) = [i \circ \alpha]_P \), so \( i \circ \alpha \) is essential in \( M - x \). \( \square \)
Chapter 3

The Topology of Blankets

The main tool in my analysis of single-layer wire routing is the lifting of cuts and wires to a simply connected covering space of the routing region. For this purpose the sketch model is not adequate because cuts and traces in a sketch have their endpoints outside the routing region, and hence cannot be lifted. To avoid this difficulty I retreat to a cleaner model, called the design model in which all entities of interest lie wholly within the routing region. The routing region in the design model is a 2-manifold with boundary called a sheet, and its simply connected covering space is called a blanket. The cuts and wires in the design model are paths called links that begin and end on the boundary of the sheet. Since the boundary is part of the sheet, they can always be lifted to the blanket.

The present chapter studies the topological properties of the elements of the design model. (I describe the design model itself at the beginning of Chapter 4, and do not take up the sketch model again until Chapter 8.) Its principal goal is to recapturing some of the simplicity of routing in channels. For example, every cut in a channel divides the channel into two pieces, and one can determine whether a wire is forced to cross the cut by checking whether its endpoints fall on opposite sides of the cut. A cut in a sheet does not separate the sheet, but we prove that every lifting of that cut to the blanket separates the blanket. This fact leads to a good definition of a necessary crossing of a cut by a wire and of the flow across a cut; see Definition 4b.2.

Looking further at separation properties, we consider how collections of cut liftings separate the blanket. There are two important results in this direction. One says that if a collection of cut liftings in a blanket forms a loop, then that loop has an inside and an outside, and no part of the blanket's boundary lies inside the loop. Consequently, a wire lifting, which must begin and end on the boundary, cannot cross into the loop without also crossing out of it. We use this fact in Chapter 4 to relate the flows across cuts. A second result says that if two cuts are homotopic as links (a concept we will define shortly), then one can choose homotopic liftings of those cuts, and they separate the components of the blanket's boundary in the same way. This fact leads to Proposition 4b.3, which states that homotopic cuts
have equal flow.

The chapter concludes with a look at the analogues of rubber bands in the design model. First we show that every path in $\mathbb{R}^n$ can be reparameterized to make it canonical without affecting its image, path class, or arc length. Then we prove that every path class of paths in a sheet contains a unique minimum-length canonical path. Such minimum-length paths, called elastic chains, will be used for several purposes later on.

Sheets and blankets

Let us begin by defining the elements of the design model. The routing region is a subspace of the plane called a sheet: a compact, connected 2-manifold whose boundary consists of two or more disjoint polygons. To make a sheet, start with a polygon $P_0$, and remove from $P_0 \cup \text{inside}(P_0)$ the insides of finitely many disjoint polygons $P_1, \ldots, P_n$ that lie inside $P_0$. If $n \geq 1$, the resulting space is a sheet whose boundary has connected components $P_0, P_1, \ldots, P_n$. These subspaces are the fringes of the sheet. The insides of the polygons $P_1, \ldots, P_n$ and the outside of $P_0$ form the routing obstacles. Because sheets are connected manifolds, Proposition 2d.5 shows that every sheet has a simply connected cover, which we call a blanket. And since sheets are connected and locally path-connected, Theorem 2b.7 shows that all blankets of a sheet are equivalent. Hence we can speak of "the" blanket of a sheet. By Proposition 2d.3, every blanket is a 2-manifold with boundary.

The simplest sort of blanket is depicted in Figure 2b-1, if one takes the borders of the annulus to be polygons. In this example the sheet has only two fringes. The blanket for a sheet with 3 or more fringes is harder to visualize and to draw, though I have made an attempt in Figure 2b-2. If one is concerned with the covering map, then one should envision the blanket as infinitely many layers lying above the sheet, connected so as to satisfy the following condition: a path in the blanket is a loop if and only if its projection to the sheet is an inessential loop. If one is concerned only with the intrinsic properties of the blanket, however, then the representation of Figure 3-1 is helpful; it embeds the blanket in a bounded region of the plane.

Our primary objects of interest are paths of various kinds. We study paths in a sheet by lifting them to the sheet's blanket. (By Theorem 2b.3, the Lifting Theorem, paths can always be lifted.) "Lifting" will always mean lifting from the sheet to its blanket. For instance, if $\Phi$ is a set of paths in a sheet, then a $\Phi$-lifting is any path in the sheet's blanket whose projection to the sheet is a member of $\Phi$.

Flat manifolds

Sheets, blankets, and all their submanifolds have a very special property: they are flat. A flat $m$-manifold $M$ is one that comes equipped with a local embedding
Figure 3-1. One way to visualize a blanket. This figure forms the basis for many subsequent pictures of blankets. (Later figures show only part of the blanket and parts of a few fringes.) All blankets are homeomorphic either to $\mathbb{R} \times I$ (as in Figure 2b-1) or to the shaded subspace of the plane. Specifically, every sheet with 3 or more fringes has this space as a blanket, though the covering map varies. Part of this surface is shown in Figure 2b-2.
Any sheet $S$ is flat, because it comes with an inclusion $i: S \to R^2$. (Inclusions are always embeddings, and hence local embeddings as well.) Every blanket is flat, for if $M$ is a blanket with covering map $p: M \to S$, then $p$ is a local homeomorphism, and hence $i \circ p: M \to R^2$ is a local embedding. A submanifold $N$ of a flat manifold $M$ is naturally a flat manifold, for if $h: M \to R^m$ is a local embedding, so is $h|_N$. Of particular importance to us are the scraps of a blanket: its simply connected, open submanifolds. All scraps are flat manifolds.

Flat manifolds inherit many nice properties of Euclidean space, such as a notion of linearity for paths. Let $M$ be flat, with local embedding $h: M \to R^m$. A path $\alpha$ in $M$ is linear if $h \circ \alpha$ is linear, straight if $h \circ \alpha$ is linear and nonconstant, and bent if there is a point $t \in I$ such that $\alpha_{0:t}$ and $\alpha_{t:1}$ are straight and intersect at $\alpha(t)$ alone. More generally, $\alpha$ is piecewise linear, abbreviated PL, if $h \circ \alpha$ is piecewise linear, and piecewise straight if $h \circ \alpha$ is piecewise linear and none of its segments is constant. A simple path is piecewise linear and injective; a simple loop is the same except that its endpoints coincide. Straight and bent paths are always simple. If $x$ is a point in $I$, we say $\alpha$ is linear at $x$, straight at $x$ or bent at $x$ if there is an interval $[s, t]$ containing a neighborhood of $x$ such that $\alpha_{s:t}$ is linear, straight, or bent, respectively. All of these properties except simplicity are preserved by lifting from a sheet to its blanket and by projecting from a blanket to its sheet. For example, if $\alpha$ is a lift of a path $\beta$, then $\alpha$ is straight at $x$ if and only if $\beta$ is straight at $x$. If $\beta$ is simple then so is $\alpha$, but the converse is false.

Another notion that flat manifolds inherit is that of arc length. Given a norm $|\cdot|$ on $R^m$, one can define the arc length of a path $\alpha$ in $M$ as $|h \circ \alpha|$. If $M$ is path-connected, one thereby obtains a metric for $M$: define the distance between two points in $M$ to be the infimum of the arc lengths of all paths between those points. You may check that this distance function is a topological metric on $M$.

Links and link homotopy

The paths in a manifold fall into different categories depending on where they touch the boundary of the manifold. A path $\alpha$ in a manifold $M$ is a link if $\alpha^{-1}(Bd M) = \{0, 1\}$, a half-link if $\alpha^{-1}(Bd M) = \{0\}$, and a mid-link if $\alpha^{-1}(Bd M)$ is empty. Links, half-links, the reverses of half-links, and mid-links are collectively called sublinks. A chain for a path $\alpha$, so called because it may contain one or more links, is any path in $[\alpha]_P$. For any manifold $M$, we call the components of $Bd M$ the fringes of $M$. The fringes that contain the endpoints of a sublink are the terminals of the sublink. A link has either one or two terminals, a half-link has one, and a mid-link has none.

The notion of homotopy for links is very important because it applies to all cuts and wires. Two links $\alpha$ and $\beta$ in a manifold $M$ are link-homotopic, written $\alpha \simeq_L \beta$, 

- 87 -
The Topology of Blankets

if there is a homotopy $H: I \times I \to M$ between $\alpha$ and $\beta$ such that $H(\{0,1\} \times I) \subseteq Bd M$. In other words, as $\alpha$ is deformed into $\beta$, its endpoints must stay on their respective fringes. Thus link-homotopic links have the same terminals. The map $H$ is called a link homotopy. One may check that the relation of being link-homotopic (also called link homotopy) is an equivalence relation; the set of links that are link-homotopic to $\alpha$ is denoted $[\alpha]_L$. Two links that are path-homotopic are also link-homotopic, so the path-homotopy class $[\alpha]_P$ is always a subset of $[\alpha]_L$.

**Liftings of links**

Because the majority of the lemmas and propositions in the next four chapters involve lifting paths from a sheet to its blanket, some further remarks about lifting are in order. Suppose the blanket $M$ covers the sheet $S$ via the map $p$. By Lemma 2d.3, the boundary of the blanket lies over the boundary of the sheet. Hence a lifting of a link is a link, a lifting of a half-link is a half-link, and a lifting of a mid-link is a mid-link. An elementary but important fact is that the liftings of a simple path are disjoint. For let $\alpha$ and $\beta$ lift the simple path $\gamma$, and suppose $\alpha(s) = \beta(t)$. Then $\gamma(s) = \gamma(t)$, whence $s = t$. Hence by uniqueness of liftings (Theorem 2b.2) we have $\alpha = \beta$.

Another useful fact is that all the lifts of a path in $S$ have the same topological properties: if $\alpha$ and $\beta$ lift the same path, then there is a covering transformation $T: M \to M$ such that $T \circ \alpha = \beta$. For by Proposition 2b.7, the covering spaces $(M, \alpha(0))$ and $(M, \beta(0))$ are equivalent; there is a covering transformation $T: M \to M$ that carries $\alpha(0)$ to $\beta(0)$. So $T \circ \alpha$ lifts the same path as $\alpha$, and it agrees with $\beta$ at 0; hence $T \circ \alpha = \beta$ by uniqueness of liftings. Moreover, $T$ is a homeomorphism, and therefore $\alpha$ and $\beta$ are topologically indistinguishable.

## 3A. Constructing Paths in Blankets

One drawback of working with blankets is that their geometry and topology are unfamiliar. Whereas in the plane one can take for granted many theorems of Euclidean geometry and geometric topology, the analogous facts about blankets are far less intuitive. Hence the need for the present chapter, which collects basic results about blankets.

We begin with several methods for constructing paths and links in blankets. First we show there exists a simple link, half-link, or mid-link between every two distinct points in a blanket. Then we characterize link homotopy in terms of path homotopy, and we prove that two links in a blanket are link-homotopic if and only if they begin and end on the same fringes. Most importantly, we relate link homotopy
in a sheet to link homotopy in its blanket. Homotopic links in a blanket, when projected to the sheet, remain homotopic; homotopic links in a sheet can always be lifted so that their liftings are homotopic.

Existence of simple paths

Because blankets are connected manifolds, they are path-connected. That is, for every two points in a blanket, there is a path that connects them. Since I work only with piecewise linear paths, I need to know that the path can always be made piecewise linear. We can prove something stronger: the path can always be made simple, and its middle need never intersect a fringe. Two lemmas are helpful in proving this claim. The first says that one can remove all self-intersections from a piecewise linear path.

**Lemma 3a.1.** For any PL path $\alpha$ in a flat manifold, there is a simple path $\beta$ in $\text{Im} \alpha$ with the same endpoints as $\alpha$, and $\|\beta\| \leq \|\alpha\|$ in any norm $\|\cdot\|$.

The proof of this lemma is an induction on the number of pairs of segments of $\alpha$ that intersect. I omit the details.

The second lemma states that one cannot disconnect a manifold by removing all or part of its boundary.

**Lemma 3a.2.** If $M$ is a connected manifold and $X \subseteq Bd M$, then $M - X$ is connected.

**Proof.** Let $N$ denote $M - X$, and suppose that $N$ is not connected. Then there are nonempty open sets $U$ and $V$ that partition $N$. Let $Cl U$ and $Cl V$ denote the closures of $U$ and $V$ in $M$. Because $M$ is connected, $Cl U$ and $Cl V$ must intersect, or else their complements, which are nonempty open sets, would partition $M$. Let $x$ be a point of $(Cl U) \cap (Cl V)$; it cannot lie in $N$, and hence must lie in $X$. Take a boundary patch $h: W \to H^n$ around $x$ whose image is an open ball. Then $h(W \cap N)$ is the connected set $h(W) - Bd H^n$ with perhaps some points of closure added, so $h(W \cap N)$ is also connected. Hence $W \cap N$ is connected since $h$ is a homeomorphism. But $W \cap N$ is the union of its disjoint open subsets $W \cap U$ and $W \cap V$, neither of which is empty, because $x$ is a point of closure of both $U$ and $V$. Thus $W \cap N$ is not connected, a contradiction.

Armed with Lemmas 3a.1 and 3a.2, we show that for any two points in a blanket, there is a simple link, half-link, or mid-link connecting them.

**Proposition 3a.3.** Every pair of points in a scrap $M$ can be connected by a simple path whose middle lies in $M - Bd M$.

**Proof.** Set $N$ equal to $M - Bd M$. We first prove the proposition in the case where both points lie in $N$. Being an open subset of a flat manifold, $N$ itself is a flat
manifold. The preceding lemma shows that $N$ is connected. Let $p: M \to R^m$ be the local embedding associated with $M$. Say that an open set $V \subseteq N$ is nice if $h$ maps $V$ homeomorphically onto a convex subset of $R^m$. Every point of $N$ has a nice neighborhood. Define an equivalence relation $\sim$ on the points of $N$ by setting $x \sim y$ if there is a finite sequence of nice sets $V_1, \ldots, V_n$ such that:

1. $x \in V_1$ and $y \in V_n$; and
2. $V_i$ meets $V_{i+1}$ whenever $1 \leq i < n$.

The equivalence classes of $\sim$ are open, and form a partition of $N$; since $N$ is connected, there can be only one equivalence class.

So for any two points $x, y \in N$, there is a finite sequence of nice sets $V_1, \ldots, V_n$ satisfying (1) and (2) above. We use this sequence to construct a simple path in $M$ from $x$ to $y$. Choose points $x = x_0, x_1, \ldots, x_n = y$ such that $x_i \in V_i \cap V_{i+1}$ for all $i$ satisfying $1 \leq i < n$. Because each set $p(V_i)$ is convex, the linear path $\lambda_i$ from $p(x_{i-1})$ to $p(x_i)$ lies in $p(V_i)$ for each $i$. Let $\alpha_i$ be $(p|_{V_i})^{-1} \circ \lambda_i$, and let $\alpha$ be the concatenated path $\alpha_1 \ast \cdots \ast \alpha_n$. Then $\alpha$ is piecewise linear, and runs from $x$ to $y$. Lemma 3a.1 reduces $\alpha$ to a simple path from $x$ to $y$.

To complete the proof, suppose that one of the points to be connected, say $x$, lies on $Bd M$. There is a path $h \circ p$, defined on a neighborhood $U$ of $x$, such that $p(U)$ is polygonal. Take any straight path from $p(x)$ whose middle lies in $Int p(U)$, and lift it to a path $\alpha$ in $U$ starting at $x$. Then $\alpha$ is a straight half-link in $M$. The previous lemma proves the existence of a simple path $\gamma$ in $N$ from $\alpha(1)$ to $y$, and the concatenated path $\alpha \ast \gamma$ is a PL half-link in $M$ from $x$ to $y$. Lemma 3a.1 reduces this path to a simple half-link from $x$ to $y$. The same technique handles the case in which both $x$ and $y$ lie on $Bd M$. □

Link homotopy

One can characterize link homotopy in terms of path homotopy, as the next lemma shows.

**Lemma 3a.4.** Two links $\alpha$ and $\beta$ in a manifold $M$ are link-homotopic if and only if there exist paths $\kappa$ and $\nu$ in $Bd M$ such that $\alpha \ast \kappa \ast \beta \ast \nu$ is an inessential loop in $M$.

**Proof.** This is a consequence of Lemma 2a.9. For there to be a link homotopy between $\alpha$ and $\beta$ means that there is a map $f: Fr(I \times I) \to M$ with an extension $F$ over $I \times I$ such that $f(\cdot, 0) = \alpha$, $f(\cdot, 1) = \beta$, and the paths $\nu = f(0, \cdot)$ and $\kappa = f(1, \cdot)$ run in $Bd M$. By Lemma 2a.9, the existence of the extension $F$ is equivalent to $f \circ \delta$ being inessential, where $\delta$ is the loop

$$\delta = (\cdot, 0) \ast (1, \cdot) \ast (\cdot, 1) \ast (0, \cdot): I \to I \times I.$$
But $f \circ \delta$ is just $\alpha \ast \kappa \ast \tilde{\beta} \ast \tilde{\nu}$, so the proof is complete. \qed

In a simply connected manifold the condition that the loop be inessential is redundant. Since fringes are path-connected, we obtain the following important corollary.

**Corollary 3a.5.** Two links in a blanket are link-homotopic if and only if they have the same terminals. \qed

Here we use the convention that $\alpha$ and $\beta$ have the same terminals if $\alpha(0)$ lies on the same fringe as $\beta(0)$, and $\alpha(1)$ lies on the same fringe as $\beta(1)$.

If link-homotopic links in a blanket are projected to the sheet, they remain link-homotopic. For if $F$ is a link homotopy between $\alpha$ and $\beta$, and if $p: M \to S$ is the covering map, then $p \circ F$ is a link homotopy between $p \circ \alpha$ and $p \circ \beta$. The next lemma is a partial converse: given link-homotopic links in a sheet, we can lift them to obtain link-homotopic links in the blanket.

**Proposition 3a.6.** Let $\alpha$ and $\beta$ be link-homotopic links in a sheet $S$, and let $M$ be a blanket of $S$. There is a bijective correspondence between the lifts of $\alpha$ to $M$ and the lifts of $\beta$ to $M$, and corresponding lifts are link-homotopic.

**Proof.** Let $p: M \to S$ be the covering map. Choose a link homotopy $F: I \times I \to S$ between $\alpha$ and $\beta$, and let $\tilde{\alpha}$ be any lift of $\alpha$. We say that $\tilde{\alpha}$ corresponds to a lift $\tilde{\beta}$ of $\beta$ if there is a link homotopy between $\tilde{\alpha}$ and $\tilde{\beta}$ that lifts $F$.

By symmetry, it suffices to show that to each lift $\tilde{\alpha}$ of $\alpha$ there corresponds a unique lift $\tilde{\beta}$ of $\beta$. Let $\tilde{\alpha}$ be given. Because $I \times I$ is a convex subset of $\mathbb{R}^2$, it is locally path-connected and simply connected, and hence by the Lifting Theorem (2b.3), $F$ has a lift $\tilde{F}: I \times I \to M$ such that $\tilde{F}(0, 0) = \tilde{\alpha}(0)$. Theorem 2b.2 shows this lift to be unique, so there is only one choice for $\tilde{\beta}$, namely $\tilde{\beta} = \tilde{F}(\cdot, 1)$. We see that $\tilde{\beta}$ is a lift of $\beta$, because

$$p \circ \tilde{\beta} = p \circ \tilde{F}(\cdot, 1) = F(\cdot, 1) = \beta.$$ 

I claim that $\tilde{F}$ is a link homotopy between $\tilde{\alpha}$ and $\tilde{\beta}$. Two things must be shown: that $\tilde{F}(\cdot, 0) = \tilde{\alpha}$, and that $\tilde{F}([0, 1] \times I)$ is contained in $\text{Bd} M$. The second is easy. Because $F$ is a link homotopy, we have

$$p \circ \tilde{F}([0, 1] \times I) = F([0, 1] \times I) \subseteq \text{Bd} S.$$ 

Now $p^{-1}(\text{Bd} S) = \text{Bd} M$ by Lemma 2d.3, and hence $\tilde{F}([0, 1] \times I) \subseteq \text{Bd} M$. To show that $\tilde{F}(\cdot, 0)$ and $\tilde{\alpha}$ coincide, note that they are lifts of $\alpha$ that agree at one point (namely 0), and apply uniqueness of liftings (Theorem 2b.2). \qed
Lifting of convergent sequences

The last result in this section concerns the lifting of another relation among paths: uniform convergence. Given a uniformly convergent sequence of paths in the sheet, we can lift them to the blanket so that the limit of the lifts is a lift of the limit.

**Lemma 3a.7.** Let $M$ be a blanket of a sheet $S$. Let $(\alpha_n)$ be a sequence of paths in $S$ that converges uniformly to a path $\alpha$, and let $\beta$ be a lift of $\alpha$ to $M$. There is a sequence of paths $(\beta_n)$ that converges uniformly to $\beta$, and $\beta_n$ lifts $\alpha_n$ for each $n$.

**Proof.** Let $p: M \to S$ be the covering map. Choose $\epsilon$ smaller than the minimum distance between fringes of $S$, and small enough that whenever two points on a fringe $V$ of $S$ are separated by a distance $\epsilon$ or less, they lie on adjacent segments of the polygon $V$. Let $P \subset S \times S$ be the set $\{(p, q) : |p - q| < \epsilon\}$, and define a function $L: P \times I \to S$ as follows. The path $L(p, q, \cdot)$ is the linear path from $p$ to $q$ if this path lies in $S$. Otherwise, let $V$ be the unique fringe of $S$ that $p \triangleright q$ intersects. Because $V$ is a convex polygon, $p \triangleright q$ crosses exactly two segments of $V$. These segments are adjacent. Let $v$ be their common vertex, and define $L(p, q, \cdot)$ to be the path $(p \triangleright v) \star (v \triangleright q)$, parameterized according to arc length. Then $L$ is a continuous function on $P \times I$. In addition, there is a constant $K$ such that the arc length of $L(p, q, \cdot)$ is at most $K |p - q|$.

We construct the sequence $(\beta_n)$ as follows. Let $\epsilon_n = \sup_{t \in I} |\alpha_n(t) - \alpha(t)|$; we have $\epsilon_n \to 0$ as $n \to \infty$. When $\epsilon_n \geq \epsilon$, let $\beta_n$ be an arbitrary lift of $\alpha_n$. Otherwise, let $F_n$ be the homotopy between $\alpha_n$ and $\alpha$ given by

$$F_n(s, t) = L(\alpha_n(s), \alpha(s), t).$$

Because $L$ is continuous, so is $F_n$. Let $G_n$ be a lift of $F_n$ satisfying $G_n(\cdot, 1) = \beta$, and set $\beta_n = G_n(\cdot, 0)$. Then $\beta_n$ lifts $\alpha_n$, and the distance between $\beta_n$ and $\beta$ is

$$\sup_{t \in I} \inf_{\sigma \in \mathcal{S}} |p \circ \sigma| : \beta_n(t) \to \beta(t) \leq \sup_{t \in I} |p \circ G_n(t, \cdot)| = \sup_{t \in I} |L(\alpha_n(t), \alpha(t), \cdot)|.$$

Since the distance between $\alpha_n(t)$ and $\alpha(t)$ is at most $\epsilon_n$, this quantity is bounded by $K \epsilon_n$. Therefore the paths $(\beta_n)$ converge uniformly to $\beta$. \qed
3B. Separation Results

In this section we consider more of the global topological properties of blankets. The main result, which is fundamental to my entire approach to wire routing, is that every simple link in a blanket splits it into two scraps. We build on this result to show that any simple loop of $k$ links splits a blanket into $k + 1$ scraps, one of which contains no fringes. If these properties seem obvious in view of Figure 3-1, you may consider this section as providing formal evidence that Figure 3-1 is an accurate representation of a blanket.

The topology of fringes

Every fringe of a sheet is a polygon, and hence homeomorphic to the circle $S^1$. It should come as no surprise, therefore, that every fringe of a blanket is homeomorphic to the real line $R^1$. To prove this fact we first need one lemma.

**Lemma 3b.1.** Every fringe of a sheet is a retract of the sheet.

**Proof.** Let $F$ be a fringe of the sheet $S$. Suppose first that $S$ lies outside $F$, by which I mean $S \subseteq F \cup \text{outside}(F)$. By Lemma 2c.3 there is a retraction of $F \cup \text{outside}(F)$ onto $F$, which when restricted to $S$ gives a retraction of $S$ onto $F$.

The other possibility is that $S$ lies inside $F$. Because $S$ has two or more fringes, there is a point $x$ in $\text{inside}(F) - S$. Let the map $h$ be inversion with respect to the unit circle centered at $x$. Since $h$ is its own inverse, it is a homeomorphism of $R^2 - x$ with itself. Now $h(S)$ is a sheet that lies outside the fringe $h(F)$. Hence there is a retraction $r$ from $h(S)$ onto $h(F)$, and the map $h \circ r \circ h$ is a retraction of $S$ onto $F$. □

Let $F$ be a fringe of a sheet $S$. As shown in Section 2A, the fact that $F$ is a retract of $S$ implies that the inclusion $i: F \to S$ induces a monomorphism of fundamental groups: every essential path in $F$ is essential in $S$. We use this fact in the following lemma and elsewhere.

**Lemma 3b.2.** Every fringe of a blanket is homeomorphic to $R^1$.

**Proof.** Let $A$ be a fringe of the blanket $M$, and let $p: M \to S$ be the covering map. By Lemma 2d.3, the fringe $A$ covers a fringe $F$ of $S$ via the map $p|_A$. We show that $A$ is simply connected, and thence Proposition 2b.7 shows that $A$ is homeomorphic to any other simply connected covering space of $F$. Since $F$ is homeomorphic to the circle $S^1$, the real line $R^1$ is one such covering space.

Because $Bd M$ is a manifold, its component $A$ is a connected manifold and hence path-connected. It remains to show that every loop $\alpha$ in $A$ is inessential in $A$. Certainly $\alpha$ is inessential in $M$, because $M$ is simply connected. Hence $p \circ \alpha$ is
inefficient in $S$. Because $F$ is a retract of $S$, by Lemma 3b.1, the loop $p \circ \alpha$ is inefficient in $F$. But $p|_A: A \to F$ is a covering map, so any lift of $p \circ \alpha$ to $A$ is inefficient in $A$. In particular, $\alpha$ is inefficient in $A$. □

**Neighborhoods of sublinks**

To determine how a set $X$ separates a blanket, we apply Proposition 2d.6 to a neighborhood $U$ of $X$ whose properties we know. Here $X$ is the image of a simple path in a flat 2-manifold. The neighborhoods we use are called tubular because they look like thin tubes about the simple path in question. A tubular neighborhood of $X$ has no holes: it separates the manifold essentially as $X$ does.

**Definition 3b.3.** Let $\alpha$ be a simple sublink in a flat 2-manifold $M$. A neighborhood $N$ of $\text{Im } \alpha$ is **tubular** if there is a piecewise linear homeomorphism $h:I \times I \to \text{Cl } N$ whose inverse $k$ has the properties shown in table 3b-1.

<table>
<thead>
<tr>
<th>$\alpha^{-1}(\text{Bd } M)$</th>
<th>$k(\text{Fr } N)$</th>
<th>$k(\text{Bd } M)$</th>
<th>$k \circ \alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\emptyset$</td>
<td>$\bigcup e_i$</td>
<td>$\emptyset$</td>
<td>$p_1 \triangleright p_2$</td>
</tr>
<tr>
<td>${0}$ or ${1}$</td>
<td>$e_1 \cup e_3 \cup e_4$</td>
<td>$e_2$</td>
<td>$p_0 \triangleright p_2$</td>
</tr>
<tr>
<td>${0, 1}$</td>
<td>$e_1 \cup e_4$</td>
<td>$e_2 \cup e_3$</td>
<td>$p_0 \triangleright p_3$</td>
</tr>
</tbody>
</table>

**Table 3b-1. Requirements for a tubular neighborhood.** For $N$ to be a tubular neighborhood of $\text{Im } \alpha$, the homeomorphism $k$ must carry the path $\alpha$ and the sets $\text{Fr } N$ and $\text{Cl } N \cap \text{Bd } M$ onto certain parts of $I \times I$, which differ depending on whether $\alpha$ is a mid-link, half-link, or link. The points $p_0, \ldots, p_3$ are given by $p_i = (\frac{i}{2}, \frac{1}{2})$, while the line segments $e_1, \ldots, e_4$ are $e_1 = I \times 1$, $e_2 = 0 \times I$, $e_3 = 1 \times I$, and $e_4 = I \times 0$.

Of course, we need to know that every simple sublink has tubular neighborhoods. To prove this rigorously would be very tedious. The following lemma shows only how to construct the neighborhoods; Figure 3b-2 suggests how one might prove that they are, in fact, tubular.

**Lemma 3b.4.** Let $\alpha$ be a simple sublink in scrap. Every neighborhood of $\text{Im } \alpha$ contains a tubular neighborhood of $\text{Im } \alpha$.

**Proof.** Let $M$ be a scrap of the blanket $B$, and let $p: B \to S$ be the covering map. Write $\alpha$ as the concatenation of finitely many paths $\alpha_i$ such that $p \circ \alpha_i$ is a line segment for each $i$. By subdividing these line segments if necessary, we may assume that for each $i$, the image of $\alpha_i$ sits inside a neighborhood $V_i$ such that $p|_{V_i}$ is an embedding. Choose a positive number $\varepsilon$ smaller than the following quantities:
(1) The minimum distance between a path $\alpha_i$ and the complement $B - V_i$ of the corresponding neighborhood.

(2) The minimum distance between paths $\alpha_i$ and $\alpha_j$, over all $i$ and $j$ with $|i - j| > 1$.

(3) The minimum distance between the compact set $\text{Im} \alpha$ and the fringe edges of $B$ that do not intersect $\text{Im} \alpha$.

(4) The distance from $\text{Im} \alpha$ to the closed set $B - M$, if the latter is nonempty.

Let $N$ be the set of points whose distance from $\text{Im} \alpha$ is less than $\epsilon/2$. Then $N$ has the desired properties; see Figure 3b-2.

![Figure 3b-2. Construction of a tubular neighborhood.](image)

**Figure 3b-2. Construction of a tubular neighborhood.** If $\alpha$ is a simple sublink in a scrap, then for sufficiently small $\epsilon$ the set of points $N$ whose distance from $\text{Im} \alpha$ in the norm $\| \cdot \|$ is less than $\epsilon$ is a tubular neighborhood of $\text{Im} \alpha$. One can set up a homeomorphism between $\text{Cl} N$ and $I \times I$ (the map $k$ with inverse $h$, as in Definition 3b.3) that takes triangles and quadrilaterals in $\text{Cl} N$ to triangles and quadrilaterals in $I \times I$, and has the properties listed in Table 3b-1.

**Threads and half-threads**

Figure 3-1 suggests that a simple link in a blanket should split it into two parts, and this claim we now prove. To be proper, one should not speak of a path separating a space, but rather of the image of the path doing so. Some new terminology is therefore helpful: a **thread** is the image of a simple link, and a **half-thread** is the image of a simple half-link. We could also consider "mid-threads", but they turn out not to be very useful. To summarize: threads separate scraps, but half-threads (and mid-threads) do not.

**Lemma 3b.5.** Removing a half-thread from a scrap leaves a scrap.

**Proof.** Let $\alpha$ be a simple half-link in a scrap $M$, let $A$ denote its image, and let $U$ be a tubular neighborhood of $A$ with homeomorphism $h: I \times I \to \text{Cl} U$. Because $A$ is a compact subset of the Hausdorff space $M$, it is closed, and hence $M - A$ is open.
in \( M \). Therefore \( M - A \) is an open subspace of a blanket. To show that \( M - A \) is simply connected, it suffices in view of Lemma 2a.7 to find a simply connected deformation retract of \( M - A \). In the notation of Table 3b-1, one can construct a deformation retraction \( F \) of \( I \times I - \overline{p_0 p_3} \) onto \( e_1 \cup e_3 \cup e_4 \). Then \( h \circ F \circ h^{-1} \) is a deformation retraction of \( \text{Cl} U \) onto \( \text{Fr} U \). Because it fixes \( \text{Fr} U \), this map extends to a deformation retraction of \( M - A \) onto \( M - U \). In a similar way one can construct a deformation retraction of \( M \) onto \( M - U \). Since \( M \) is simply connected, Lemma 2a.7 shows that \( M - U \) is simply connected. \( \Box \)

**Proposition 3b.6.** Removing a thread from a scrap leaves two scraps whose common frontier is the thread.

**Proof.** The construction is illustrated in Figure 3b-3 below. Let \( \alpha \) be a simple link in a scrap \( M \), let \( C \) denote its image, and let \( U \) be a tubular neighborhood of \( C \) with homeomorphism \( h: I \times I \to \text{Cl} U \). From Table 3b-1 we see that \( h \) carries \( I \times \frac{1}{3} \) and \( I \times (0, 1) \) onto \( C \) and \( U \), respectively. Hence \( U - C \) has two path components, call them \( A' \) and \( B' \), and their closures in \( U \) include \( C \). Lemma 2d.6 now implies that \( M - C \) has exactly two path components, each containing a path component of \( U - C \). Call them \( A \) and \( B \), and say \( A \supseteq A' \) and \( B \supseteq B' \). The set \( M - C \) is open in \( M \), because \( C \) is a compact subset of the Hausdorff space \( M \), and therefore closed in \( M \). Hence \( M - C \) is locally path-connected (because \( M \) is), and so its path components \( A \) and \( B \) are open. Therefore \( A \) and \( B \) are the components of \( M - C \); we have \( \text{Cl} A \subseteq M - B \) and \( \text{Cl} B \subseteq M - A \). But we also know

\[
\text{Cl} A \supseteq \text{Cl} A' \supseteq C \quad \text{and} \quad \text{Cl} B \supseteq \text{Cl} B' \supseteq C.
\]

Together these facts imply \( \text{Cl} A = A \cup C \) and \( \text{Cl} B = B \cup C \), whence \( \text{Fr} A = C = \text{Fr} B \) because \( A \) and \( B \) are open.

![Figure 3b-3. How a thread separates a scrap.](image)

The tubular neighborhood \( U \) of the thread \( \text{Im} \alpha \) shows us how \( \text{Im} \alpha \) is embedded in the blanket. Every topological relationship among the path \( p_0 \circ p_3 \) and the components of its complement in \( I \times I \) also obtains for \( \alpha \) and the components of its complement in \( \text{Cl} U \).
Section 3B

Separation Results

By the symmetry between $A$ and $B$, it suffices to prove that $A$ is a scrap. We know that $A$ is open, so we need only show that $A$ is simply connected. The set $I \times I - \bar{D}_{01}D_2$ has a deformation retraction onto $e_1 \cup e_4$. Pulling back via $h$, one obtains a deformation retraction of $(C U) - C$ onto $Fr U$. Since this map fixes $Fr U$, it extends to a deformation retraction of $M - C$ onto $M - U$. Restricting to $A$, we obtain a deformation retraction of $A$ onto $A - U$. For similar reasons, there is a deformation retraction of $A \cup C$ onto $A - U$. Hence by Lemma 2a.7, if we show that $A \cup C$ is simply connected, it will follow that $A - U$ and $A$ are simply connected. We exhibit $A \cup C$ as a retract of the simply connected space $M$. The map $\alpha$ is an embedding of $I$ as a closed subspace $C$ of $B \cup C$, and the latter space is normal. (The blanket containing $B \cup C$ is metrizable, hence $B \cup C$ is metrizable, and thus normal.) Since $I$ is an a absolute retract, there must be a retraction $r: B \cup C \to C$. This map $r$ may be extended over $M$ by making it the identity on $A$. Then $r$ is a retraction of $M$ onto $A \cup C$. Because $M$ is simply connected, so is the retract $A \cup C$ of $M$. □

Weaving threads into webs

Building on Proposition 3b.6, one can determine how groups of threads partition a blanket. Simple loops are especially important to analyze. Let $\lambda$ be a simple loop in a blanket $M$, and suppose that $Im \lambda \cap Bd M$ has $k > 0$ components, each containing more than one point. Then $\lambda$ is called a loop of $k$ links. In addition, the set $Cl(Im \lambda - Bd M)$ is the union of $k$ disjoint threads, and is called a web of $k$ threads. A straightforward induction shows that a web of $k$ threads splits a blanket into $k + 1$ parts.

**Lemma 3b.7.** Removing a web of $k$ threads from a blanket leaves exactly $k + 1$ scraps. One has the entire web as frontier, while the others border on one thread each.

**Proof.** Let $\lambda$ be a loop of $k$ links, and let $Im \beta_1, \ldots, Im \beta_k$ be the threads contained in $Im \lambda$. We apply Proposition 3b.6 to each of the threads $Im \beta_i$. First consider $Im \beta_1$: it separates the blanket into two scraps, only one of which contains the remaining links $\beta_2, \ldots, \beta_k$, because the loop $\lambda$ is simple. The thread $Im \beta_2$ separates this scrap into two scraps, one of which contains $\beta_3, \ldots, \beta_k$. Continue in this way, obtaining $k + 1$ scraps. At each stage, exactly one of the scraps contains the remaining threads, and borders on all the threads removed; each of the other scraps borders on one thread $Im \beta_i$. □

In Lemma 3b.7 the special scrap is called the inside of the loop, or of the web. Figure 3-1 suggests strongly that the inside of a web is compact and that it contains only parts of fringes. The following rather technical result bears out these conjectures.
**Proposition 3b.8.** No fringe lies inside a web of threads.

**Proof.** We begin by showing that the closure of the inside of a web is simply connected. Let $T_1, \ldots, T_k$ be the threads that make up a web $W$ of $k$ threads in a blanket $M$. Let $B$ denote the inside of $W$, and for $1 \leq i \leq k$, let $A_i$ be the component of $M - W$ that borders only the thread $T_i$. For each $i$, the absolute retract $I$ is embedded in the normal space $A_i \cup T_i$ as the closed set $T_i$, so there is a retraction $r_i$ of $A_i \cup T_i$ onto $T_i$. Define a retraction $r: M \to B \cup W$ by $r(x) = r_i(x)$ if $x \in A_i \cup T_i$, and $r(x) = x$ if $x \in B \cup W$. These definitions agree on their intersection, which is $W$, so $r$ is indeed continuous. Its image is the space $C = Cl B = B \cup W$. Thus $C$ is a retract of $M$, and because $M$ is simply connected, so is $C$.

In the remainder of the proof we prove that $Cl B$ is compact, whence it follows that $B$ includes no fringe of $M$. For if $X$ is a fringe of $M$, then by Lemma 3b.2, it is homeomorphic to $R^1$. Hence $X$ contains an infinite discrete subspace $Z$, and since $X$ is closed in $M$, this subset is discrete in $M$. The points of $Z$ cannot all lie in a compact subspace $C$ of $M$, and so neither does $X$.

Because $C$ is simply connected and contains $Im \lambda$, there is a path homotopy $F: I \times I \to C$ between the loop $\lambda$ and the constant loop at $\lambda(0)$. Suppose that $x \in C - Im F$. Certainly $x \notin Bd C$, because $Bd C \subset Im \lambda \subset Im F$. Let $y$ be an arbitrary point of $B - Bd B$. Because $Cl B$ is a connected manifold, there is by Lemma 2d.4 a homeomorphism $h: C \to C$ that fixes $Bd C$ and carries $x$ onto $y$. Then $h \circ F$ is a path homotopy between $\lambda$ and a constant loop, and $y \notin Im (h \circ F)$. Hence we may choose any point $x \in C - Bd C$ and assume $x \notin Im F$.

![Figure 3b-4. The inside of a loop of links.](image)

The path $\lambda$ is a loop of 4 links in the blanket $M$. It is essential in the space $M - x$, because $\lambda \simeq_p \mu \ast \nu$, and $\nu$ is essential in $M - x$ but $\mu$ is not. In fact, $\lambda$ is essential in $M - x$ for any point $x$ inside $\lambda$.

We choose a point $x \in C - Bd C$ very near $W$. Let $U$ be a neighborhood of $\lambda(0)$ that intersects only two line segments of $W$. We may assume that $\lambda(0)$ lies in $Bd M$. If $p: M \to R^2$ denotes the local embedding attached to $M$, we may also assume that $p$ embeds $U \cap C$ as a polygonal region in $R^2$. Choose $x \in U \cap B$. As shown in Figure 3b-4, there is a loop $\nu$ at $\lambda(0)$ and a loop $\nu$ links $\mu$ at $\lambda(0)$ such that

- $\lambda \simeq_p \mu \ast \nu$,
- the loop $p \circ \nu$ is a simple polygon that encloses $p(x)$, and

- $98$ -
Section 3C

Properties of Separations

- $x$ lies outside $\mu$.

Since the closure of the inside of $\mu$ is simply connected, $\mu$ is inessential in that subspace of $C - x$, and hence $\mu$ is inessential in $C - x$. On the other hand, $\nu$ is essential in $U \cap C - x$, because $p \circ \nu$ is essential in $p(U \cap C) - p(x)$, and $p|_{U \cap C}$ is an embedding.

Lemma 2d.7 applies to the neighborhood $U$ of $M$; it says that $\nu$ is essential in $M - x$. But $\mu$ is not, because $x$ lies outside $\mu$. Therefore $[\nu]_P = [\mu \ast \nu]_P = [\nu]_P \neq 0$ in the fundamental group of $M - x$. But $F$ is a path homotopy in $M - x$ from $\lambda$ to a constant map. This contradiction shows that our assumption $C \neq \text{Im } F$ was faulty. So $C = \text{Im } F$, which is compact because $I \times I$ is compact. \[ \square \]

3C. Properties of Separations

When a loop splits the plane or a blanket, there is a convenient distinction between the inside of the loop and its outside. And when a link separates a blanket, we can distinguish between the left-hand side of the link and the right-hand side. This section explores the implications of the distinctions between left and right, and between inside and outside. One important result is that link-homotopic simple links partition the fringes of a blanket in the same way. Chapter 4 uses this result to show that the necessity of a crossing is invariant under link homotopy. Another result of this section says that simple loops in a blanket behave a lot like polygons in the plane: the measures of their internal angles, at least, have the same sum as they would for a polygon of the same number of vertices.

The two sides of a link

Because links are paths and not their images, every link is oriented. Hence when a simple link cuts a blanket into two scraps, one of these lies to the left of the link, and one lies to the right. This may seem obvious, but it requires some justification. Since we know what left and right mean in a sheet, we use the covering map to give an orientation to the blanket.

**Definition 3c.1.** Let $\alpha$ be a simple link in the blanket $M$, and let $p: M \rightarrow S$ be the covering map. Let $\tau$ be a linear path that intersects $\text{Im } \alpha$ at the point $\tau(1) = \alpha(x)$ alone. We say $\tau$ contacts $\alpha$ **from the left** or **from the right** according to whether the path $p \circ \tau$ contacts $p \circ \alpha$ from the left or the right in $S$.

What one must prove is that $\tau$ contacts $\alpha$ from the left if and only if $\tau(0)$ lies in a particular scrap of $M - \text{Im } \alpha$. We call this scrap **left**($\alpha$), the **left side** or **left scrap** of $\alpha$, and we call the other scrap **right**($\alpha$), the **right side** or **right scrap** of $\alpha$.
Figure 3c.1. The two sides of a simple link in a blanket. The link $\alpha$ separates the blanket $M$ into two scraps, denoted left$(\alpha)$ and right$(\alpha)$. The shaded area represents a tubular neighborhood of $\alpha$. The straight paths $\sigma$ and $\tau$ both contact $\alpha$ from the left, and so $\sigma$ can be moved to coincide with $\tau$ by a series of dilations, translations, and rotations. At each stage the origin of the path stays in the same scrap of $M - \text{Im} \alpha$. This construction shows that the left and right scraps of $\alpha$ are uniquely determined.

To prove that the left and right sides of $\alpha$ are well defined, one shows that if both $\sigma$ and $\tau$ contact $\alpha$ from the left (or right), then $\sigma$ can be moved along $\alpha$ until it coincides with $\tau$. The construction, suggested by Figure 3c.1, uses a tubular neighborhood of $\alpha$, and is rather messy. A similar idea underlies the following important proposition.

**Proposition 3c.2.** Let $\alpha$ and $\beta$ be simple links in a blanket, and suppose for some $e \in \{0,1\}$ that the points $\alpha(e)$ and $\beta(e)$ share a fringe. If $\beta$ lies in left$(\alpha)$, then $\alpha$ lies in right$(\beta)$, and we have the relations

$$\text{left}(\beta) \subset \text{left}(\alpha) \quad \text{and} \quad \text{right}(\alpha) \subset \text{right}(\beta).$$

**Proof.** Let $M$ denote the blanket. Choose $x$ small enough that $\alpha_{e;x}$ and $\beta_{e;x}$ are straight, as shown in Figure 3c.2 below. Let $\kappa$ be a simple path in $\text{Bd} \ M$ from $\alpha(e)$ to $\beta(e)$; it intersects $\text{Im} \alpha \cup \text{Im} \beta$ at its endpoints alone. Choose $s$ and $t$ so that $\kappa_{s;0}$ and $\kappa_{t;1}$ are straight. Because $\kappa(1)$ lies in left$(\alpha)$, so does $\kappa(s)$, and hence the path $\kappa_{s;0}$ contacts $\alpha$ from the left. Now let $p: M \rightarrow S$ be the covering map, and let $F$ be the fringe of $S$ containing $p \circ \kappa$. Let $\alpha'$ denote the path $p \circ \alpha_{e;x}$ if $e = 0$, and $p \circ \alpha_{e;x}$ if $e = 1$. Similarly define $\beta'$. By Definition 3c.1, the path $p \circ \kappa_{s;0}$ contacts $\alpha'$ from the left. Hence $p \circ \kappa$ traverses $S$ in a counterclockwise direction if $e = 0$, or in a clockwise direction if $e = 1$. In either case, $p \circ \kappa_{t;1}$ contacts $\beta'$ from the right. Therefore $\kappa_{t;1}$ contacts $\beta$ from the right, which means that $\kappa(t)$ lies in right$(\beta)$. Therefore $\kappa(0)$, and in fact all of $\text{Im} \alpha$, falls in right$(\beta)$. 

- 100 -
The desired inclusions now follow. Because \( \text{Im } \alpha \) lies in \( \text{right}(\beta) \), it does not intersect \( \text{left}(\beta) \), and therefore the connected set \( \text{left}(\beta) \) lies entirely in one scrap of \( M - \text{Im } \alpha \). And since \( \text{left}(\beta) \) contains points arbitrarily close to \( \beta \), and \( \beta \) lies in \( \text{left}(\alpha) \), the intersection \( \text{left}(\beta) \cap \text{left}(\alpha) \) is nonempty. Thus we obtain \( \text{left}(\beta) \subseteq \text{left}(\alpha) \). And this inclusion implies the other, because

\[
\text{right}(\beta) = M - \text{Im } \beta - \text{left}(\beta) \supseteq M - \text{left}(\alpha) \supseteq \text{right}(\alpha). \]

By symmetry, the claim remains true if we exchange \text{left} and \text{right} throughout.

Separations by homotopic links

A simple link in a blanket partitions the fringes of the blanket into three categories: those in its left side, those in its right side, and those it intersects (its terminals). How does this classification change when a link homotopy is applied to the link? The answer is that it remains unchanged. This fact follows fairly easily from Proposition 3b.8 if the links are disjoint, for then their images form a web of two threads. To deal with the possibility that the links intersect, I introduce one more method for constructing links. I call it the “detour lemma”, because it constructs a simple link that detours around the right-hand sides of two given links.

**Lemma 3c.3.** (Detour Lemma) Let \( \alpha \) and \( \beta \) be link-homotopic simple links in a blanket \( M \). There is a simple link \( \gamma \) in \( \text{Im } \alpha \cup \text{Im } \beta \), link-homotopic to \( \alpha \) and \( \beta \), such that \( \text{right}(\gamma) \) includes \( \text{right}(\alpha) \) and \( \text{right}(\beta) \).

**Proof.** We construct \( \gamma \) by successive approximations. Begin with \( \gamma = \alpha \), and let \( L \) and \( R \) be the left-hand and right-hand scraps of \( \gamma \), respectively. Already \( \gamma \) satisfies all the conditions except \( \text{right}(\gamma) \supseteq \text{right}(\beta) \). Because \( \gamma \) and \( \beta \) are piecewise linear, the path \( \beta \) protrudes into \( L \) only \( n \) times for some finite \( n \). We proceed by induction on \( n \), preserving all the conditions on \( \gamma \) except \( \text{right}(\gamma) \supseteq \text{right}(\beta) \). In the basis
case \( n = 0 \), the links \( \alpha \) and \( \beta \) do not intersect. Since \( \alpha(0) \) and \( \beta(0) \) share a fringe, Proposition 3c.2 implies either \( \text{right}(\alpha) \subset \text{right}(\beta) \) (if \( \text{Im } \beta \subset \text{left}(\alpha) \)) or \( \text{right}(\beta) \subset \text{right}(\alpha) \) (if \( \text{Im } \beta \subset \text{right}(\alpha) \)). Choose \( \gamma = \beta \) or \( \gamma = \alpha \) accordingly.

Now suppose that \( n > 0 \). Let \((s, t)\) be one of that open intervals that compose \( \beta^{-1}(L) \). Splice the path \( \beta_{st} \) into \( \gamma \) to form a simple link \( \gamma' \). Let \( L' \) and \( R' \) be the left-hand and right-hand scraps of \( \gamma' \). Because \( \gamma' \) shares some line segments of \( \gamma \), the scraps \( R' \) and \( R \) intersect. Hence \( R' \supseteq R \), since \( R \) is connected and does not intersect \( \text{Im } \gamma' \). We also have \( L' \subseteq M - R' - \text{Im } \gamma \), whence \( L' \subseteq L \). The containment is proper because \( \beta_{st} \) lies in \( L \) but not \( L' \). Replacing \( \gamma \) by \( \gamma' \), we reduce \( n \) by at least 1. Furthermore, the conditions on \( \gamma \) are maintained: we have \( \text{Im } \gamma' \subseteq \text{Im } \alpha \cup \text{Im } \beta \); the terminals of \( \gamma' \) are those of \( \alpha \) and \( \beta \); and the right side \( R' \) of \( \gamma' \) includes \( R \), which includes \( \text{right}(\alpha) \) by assumption. The existence of the desired path \( \gamma \) follows by induction. □

Now we can prove the main result of this section.

**Proposition 3c.4.** Link-homotopic simple links in a blanket partition the fringes identically.

**Proof.** Let \( M \) be a blanket, with covering map \( p : M \to S \), and let \( \alpha \) and \( \beta \) be link-homotopic simple links in \( M \). We first show that \( \alpha \) and \( \beta \) may be assumed not to intersect, by finding a simple link \( \delta \) that is link-homotopic to both \( \alpha \) and \( \beta \), but intersects neither of them. Apply Lemma 3c.3 to \( \alpha \) and \( \beta \), obtaining a simple link \( \gamma \). The left-hand scrap of \( \gamma \) contains no points of \( \text{Im } \alpha \) or \( \text{Im } \beta \), else it would contain points in \( \text{right}(\alpha) \) or \( \text{right}(\beta) \), contradicting Lemma 3c.3. Let \( \delta \in [\gamma]^L \) be a simple link in \( \text{left}(\gamma) \). Then \( \delta \) intersects neither \( \alpha \) nor \( \beta \), and since \( \gamma \simeq_L \alpha \), we have \( \delta \simeq_L \alpha \) as well.

We may therefore assume that the link-homotopic links \( \alpha \) and \( \beta \) are disjoint. By Corollary 3a.5, \( \alpha \) and \( \beta \) have the same terminals. Hence the set \( \text{Im } \alpha \cup \text{Im } \beta \) is a web of 2 threads, because there is a loop of 2 links formed by \( \alpha \), a path in \( \text{Bd } M \) from \( \alpha(1) \) to \( \beta(1) \), the reverse of \( \beta \), and a path in \( \text{Bd } M \) from \( \beta(0) \) to \( \alpha(0) \). By Lemma 3b.7, the set \( M - (\text{Im } \alpha \cup \text{Im } \beta) \) has three components. One of these is a component of \( M - \text{Im } \alpha \), one is a component of \( M - \text{Im } \beta \), and the third (the inside of the web) contains no fringes, by Lemma 3b.8. Therefore \( \alpha \) and \( \beta \) separate the fringes into the same three categories. Moreover, the fringes in the left scrap of \( \alpha \) are also in the left scrap of \( \beta \), because of Lemma 3c.2. □

**The inside of a simple loop**

According to Proposition 3b.8, each loop of links has an inside that contains no fringes. The same goes for simple loops in general, although we cannot say as much about the remaining components. One can prove this fact by analyzing an arbitrary simple loop in terms of loops of links.

- 102 -
**Proposition 3c.5.** The image of a simple loop separates a blanket into two or more components, exactly one of which intersects no fringes. □

The distinguished component is, of course, the inside of the loop. Since we are removing the entire image of the loop, and not just the threads it contains, the inside component actually avoids all fringes. We cannot claim that the components are scraps: if the loop touches no fringes, then its outside component is not simply connected.

**Internal angles**

Any bent path in a blanket makes an angle, and this angle can be measured by projecting it to the sheet. There is some ambiguity in this measurement, however: is the measure of the angle \( \theta \) or \( 2\pi - \theta \)? If the bent path is part of a simple loop, then the ambiguity can be resolved by considering the interior of the angle to be the side facing the inside of the loop. The resulting angle is called internal angle of the loop at that vertex. If the projection of the loop is a polygon with \( n \) vertices, we know by Euclidean geometry that the measures of the internal angles sum to \( (n - 2)\pi \). The same is true for any simple loop.

**Lemma 3c.6.** If \( \lambda \) is a simple loop in a blanket, and \( \lambda \) has \( n \) vertices, then the measures of the internal angles of \( \lambda \) sum to \( (n - 2)\pi \).

**Proof.** The proof is an induction that works by triangulating the loop. Let \( \angle \lambda \) denote the sum of the measures of the internal angles of \( \lambda \). The basis case is \( n = 3 \), when the projection of \( \lambda \) is a triangle. For the induction step, let \( \lambda(a) \) and \( \lambda(b) \) be the vertices adjacent to \( \lambda(0) \), where \( 0 < a < b < 1 \), and denote by \( m \) the measure of the internal angle formed by these three points. We can create and delete vertices of \( \lambda \) with measure \( \pi \) at will, for these operations change \( \angle \lambda \) and \( (n - 2)\pi \) by the same amount. Hence we can assume \( m \neq \pi \).

We find a linear path \( \tau \) whose middle lies inside \( \lambda \), and which divides \( \lambda \) into two loops with fewer than \( n \) vertices. See Figure 3c-3. If \( m > \pi \), extend the linear path \( \lambda_{a,0} \) into \( \text{inside}(\lambda) \) until it reaches a point \( \lambda(t) \); let \( \tau \) be the resulting linear path \( \lambda(0) \triangleright \lambda(t) \). If \( m < \pi \), let \( T \subseteq I \) be set of values \( t \) for which there is a linear path from \( \lambda(ta) \) to \( \lambda(1 - t + tb) \) whose middle lies inside \( \lambda \). If \( 1 \in T \), then let \( \tau \) be the linear path \( \lambda(a) \triangleright \lambda(b) \). Otherwise for \( t = \sup T \) the middle of \( \lambda(ta) \triangleright \lambda(1 - t + tb) \) intersects a vertex \( \lambda(s) \) of \( \lambda \); let \( \tau \) be the linear path from \( \lambda(0) \) to \( \lambda(s) \).

In each case \( \tau \) divides \( \lambda \) into simple loops \( \mu \) and \( \nu \) with fewer vertices than \( \lambda \). If necessary, we create a vertex of \( \lambda \) at \( \tau(1) \), so that the both endpoints of \( \tau \) are vertices of \( \lambda \). Then if \( \lambda \) has \( n \) vertices, \( \mu \) has \( k + 2 \) and \( \nu \) has \( n - k \). You can check that \( 0 < k < n - 2 \) in each of the cases (a), (b), and (c). The insides of the loops \( \mu \) and \( \nu \) cannot intersect \( \text{outside}(\lambda) \), else they would contain an entire component of
Figure 3c.3. Triangulation of a simple loop. Any simple loop in a blanket can be triangulated using these three operations: (a) extending an edge into the loop at an internal angle of measure \( m > \pi \); (b) cutting off a triangle where the loop has an internal angle of measure \( m < \pi \); and (c) if the linear path in part (b) does not exist or leaves the loop, dividing the internal angle with a linear path to the "nearest" other vertex.

outside(\( \lambda \)) and hence intersect a fringe, in contradiction to Lemma 3c.5. It follows that every internal angle of \( \mu \) and \( \nu \) is part of an internal angle of \( \lambda \), and thus \( \angle \lambda = \angle \mu + \angle \nu \). The induction hypothesis now shows

\[
\angle \lambda = (k + 2 - 2)\pi + (n - k - 2)\pi = (n - 2)\pi,
\]

and the proof is complete. □

Corollary 3c.7. Every simple loop in a blanket has at least three internal angles of measure less than \( \pi \). □

3D. Elastic Chains in Sheets

Now we apply some of our results about blankets to paths in sheets. In Chapter 1 we saw the usefulness of rubber bands in sketches. The notion of a rubber band is even more natural in the sheet model, because the rubber band of a path need not leave the routing region. Recall that a chain for a path \( \alpha \) is any path in \([\alpha]_P\). An elastic path is a canonical path \( \alpha \) whose euclidean arc length is minimum among all paths in \([\alpha]_P\). The main result of this section is that every path has a unique elastic chain. It builds on two things: Lemma 3d.1 below, which says that every path can be made canonical without changing its path class, image, or arc length; and the results of the preceding section concerning loops in a blanket.

Parameterization of paths

The uniqueness result for elastic chains depends on the condition that an elastic chain be canonical. Without this restriction, all parameterizations of a minimum-length path would be elastic. The following lemma justifies our concentration on
canonical paths; it shows that every path of finite arc length can be reparameterized to make it canonical.

**Lemma 3d.1.** (Reparameterization Lemma) Let \( \alpha \) be a path in \( \mathbb{R}^n \) whose euclidean arc length \( |\alpha| \) is finite. Then the map \( f: s \mapsto |\alpha_{0:s}| / |\alpha| \) has a right inverse \( g: I \rightarrow I \), and the function \( \beta = \alpha \circ g \) is a canonical path with the same arc length as \( \alpha \). Furthermore,

(1) \( \beta \simeq P \alpha \) as paths in \( Im \alpha \);

(2) \( \beta \) is piecewise linear if \( \alpha \) is; and

(3) unless \( \alpha \) is constant, \( \beta \) is not constant on any open interval of \( I \).

The function \( g \) is not necessarily continuous, but \( \alpha \circ g \) is.

**Proof.** The function \( f: I \rightarrow I \) defined by \( f(s) = |\alpha_{0:s}| / |\alpha| \) is monotonic (non-decreasing) and continuous; it also satisfies \( f(0) = 0 \) and \( f(1) = 1 \). Hence \( f \) is surjective, so we can define a function \( g: I \rightarrow I \) by \( g(t) = \inf f^{-1}(t) \). Then \( g \) is monotonic because \( f \) is. Since \( f^{-1}(t) \) is closed, we have \( g(t) \in f^{-1}(t) \), which implies \( f \circ g = id_I \). In other words, \( g \) is a right inverse of \( f \). Put \( \beta = \alpha \circ g \). We prove \( \alpha = \beta \circ f \) by showing that for any \( s \in I \), the path \( \alpha \) maps \( s \) and \( g(f(s)) \) to the same point. Put \( s' = g(f(s)) \). Then \( f(s) = (f \circ g \circ f)(s') = f(s') \), which means that \( \alpha_{0:s} \) and \( \alpha_{0:s'} \) have the same length. Hence \( ||\alpha_{s:s'}|| = 0 \), which means that \( \alpha \) is constant on \([s',s] \).

To prove that \( \beta \) is continuous, let \( \delta \) and \( t \) be given; we set \( \epsilon = \delta / |\alpha| \) and show that \( |t'-t| < \epsilon \) implies \( |\beta(t') - \beta(t)| < \delta \). Put \( s = g(t) \) and \( s' = g(t') \). Then we have

\[
|\alpha_{0:s}| = t \cdot |\alpha| \quad \text{and} \quad |\alpha_{0:s'}| = t' \cdot |\alpha|.
\]

The difference between the left-hand sides of these equations is \( |\alpha_{s:s'}| \), which is no less than the distance from \( \alpha(s) \) to \( \alpha(s') \). But these points are just \( \beta(t) \) and \( \beta(t') \), respectively. Thus

\[
|\beta(t) - \beta(t')| \leq |\alpha_{s:s'}| = t' \cdot |\alpha| - t \cdot |\alpha| < \epsilon |\alpha| = \delta.
\]

Therefore \( \beta \) is a path.

Now we show that \( |\beta_{0:t}| = t \cdot |\alpha| \) for arbitrary \( t \in I \), thus proving that \( \beta \) is a canonical path with \( |\beta| = |\alpha| \). We have \( \beta(t) = \alpha(s) \) where \( |\alpha_{0:s}| = t \cdot |\alpha| \), so it suffices to show that \( \beta_{0:t} \) and \( \alpha_{0:s} \) have the same arc length. By the definition of arc length, it is enough to show that \( \beta_{0:t} \) and \( \alpha_{0:s} \) have the same polygonal approximations. Let \( \gamma \) be a polygonal approximation to \( \alpha_{0:s} \) with vertices \( \alpha(s_0), \alpha(s_1), \ldots, \alpha(s_n) \); we have \( s_0 = 0 \) and \( s_n = s \). The vertices of \( \gamma \) can also be written in the form \( \beta(f(s_0)), \beta(f(s_1)), \ldots, \beta(f(s_n)) \). Since \( f \) is a monotonic function satisfying

\[ -105 \]
$f(0) = 0$ and $f(s) = t$, the path $\gamma$ is also a polygonal approximation to $\beta$. Similarly, if $\gamma$ is a polygonal approximation to $\beta$ with vertices $\beta(t_0), \beta(t_1), \ldots, \beta(t_n)$, then this sequence can be written $\alpha(g(s_0)), \alpha(g(s_1)), \ldots, \alpha(g(s_n))$. Because $g$ is a monotonic function satisfying $g(0) = 0$ and $g(t) = s$, the path $\gamma$ is also a polygonal approximation to $\beta$.

Finally, we prove claims (1) through (3). The map $f$ is a path in $I$ from $0$ to $1$, and since $I$ is simply connected, there is a path homotopy $F: I \times I \rightarrow I$ between $f$ and the identity on $I$. Because $\beta = \alpha \circ f$, the map $\alpha \circ F$ is a path homotopy between $\beta$ and $\alpha$. Also $\text{Im}(\alpha \circ F) = \text{Im} \alpha$, so claim (1) is proved. For claim (2), suppose $\alpha$ is piecewise linear with vertices $\alpha(s_0), \alpha(s_1), \ldots, \alpha(s_n)$. Then the function $f$ is linear on each interval $[s_{i-1}, s_i]$ as is $\alpha$, and so the map $\beta = \alpha \circ g$ is also linear on each interval $[f(s_{i-1}), f(s_i)]$. Since these intervals cover $I$, the path $\beta$ is piecewise linear. For claim (3), suppose $\beta$ is constant on some open interval $(x, y)$. Then we have

$$0 = |\beta_{xy}| = |\beta_{0y}| - |\beta_{0x}| = (y - x) \cdot |\alpha|,$$

so $|\alpha| = 0$, which implies that $\alpha$ is constant. \qed

Existence and uniqueness of elastic chains

Our results concerning elastic chains are established in five steps. The first step is a very intuitive one. It says that for a path to be minimal in length, all its subpaths must also be minimal.

**Lemma 3d.2.** Every subpath of an elastic chain is elastic.

**Proof.** Let $\beta$ be an elastic chain, and let $\beta_{st}$ be a subpath of $\beta$. First of all, $\beta_{st}$ is canonical because for $x \in I$, we have

$$|(\beta_{st})_{a:x}| = |\beta_{a:t+s(t-a)}| = x \cdot |t - s| \cdot |\beta| = x \cdot |\beta_{st}|$$

since $\beta$ is canonical. And if $\gamma$ is path-homotopic to $\beta_{st}$, then the path $\beta'$ defined by

$$\beta'_{0:a} = \beta_{0:a}, \quad \beta'_{st} = \gamma, \quad \beta'_{t:1} = \beta_{t:1}$$

is path-homotopic to $\beta$, and its euclidean arc length differs from that of $\beta$ by $|\gamma| - |\beta_{st}|$. Because $\beta$ is elastic, we have $|\beta'| \geq |\beta|$, and hence $|\gamma| \geq |\beta_{st}|$. Hence $\beta_{st}$ has minimum length among all paths in its path-homotopy class. \qed

The second step provides an important special class of elastic paths.

**Lemma 3d.3.** A linear path is the unique elastic path in its path class.

**Proof.** Let $\alpha$ and $\sigma$ be path-homotopic elastic paths, and suppose $\sigma$ is linear. Put $l = |\alpha| = |\sigma|$. Then $l = |\sigma| = |\alpha(0) - \alpha(1)|$ because $\sigma$ is linear and has the same
endpoints as $\alpha$. For $t \in I$, we have

$$
|\alpha(t) - \alpha(0)| \leq |\alpha_{0,tl}| = tl,
$$

$$
|\alpha(t) - \alpha(1)| \leq |\alpha_{t,1}| = (1 - t)l.
$$

Thus $\alpha(t)$ lies within $tl$ units of $\alpha(0)$ and $(1 - t)l$ units of $\alpha(1)$. Only one point does so, namely $t\alpha(0) + (1 - t)\alpha(1)$, which is $\sigma(t)$. Therefore $\alpha(t) = \sigma(t)$, and this identity holds for all $t$. □

The third step is the construction of elastic chains by means of Proposition 2c.8 and Lemma 3d.1.

**Lemma 3d.4.** Every path in a sheet has an elastic chain.

**Proof.** Let $\alpha$ be a path in the sheet $S$. First we show that $[\alpha]_P$ contains a minimum-length path. Let $\Pi$ be the set of paths in $[\alpha]_P$, and let $l$ denote $\inf_{\rho \in \Pi} |\rho|$. If some path $\rho \in \Pi$ satisfies $|\rho| = l$, then we are done. Otherwise by Proposition 2c.8 there is a uniformly convergent sequence $(\rho_i)_{i=1}^\infty$ of links in $\Pi$ whose limit $\rho$ satisfies $|\rho| \leq l$. Because every link in $\Pi$ has the same endpoints as $\alpha$, we have $\rho(0) = \alpha(0)$ and $\rho(1) = \alpha(1)$.

We prove that $\rho$ and $\alpha$ are path-homotopic. Let $M$ be the blanket of $S$, and let $\tilde{\rho}$ be any lifting of $\rho$ to $M$. By Lemma 3a.7, there are liftings $\tilde{\rho}_k$ of the paths $\rho_k$ that converge uniformly to $\tilde{\rho}$. Because the inverse image of $\rho(0)$ under the covering map is discrete, and similarly for $\rho(1)$, the paths $\tilde{\rho}_k$ must have the same endpoints as $\tilde{\rho}$ for sufficiently large $k$. Hence $\tilde{\rho}_k \simeq_P \tilde{\rho}$, which implies $\rho_k \simeq_P \rho$, for sufficiently large $k$. Therefore $\alpha \simeq_P \rho$. Now by Lemma 3d.1 there is a canonical path $\beta \in [\rho]_P$ whose arc length is that of $\rho$. This path $\beta$ is an elastic chain for $\alpha$. □

The fourth step brings elastic chains into the universe of piecewise linear objects, where we can apply our previous results to them. Let $\alpha$ be a PL path in a sheet $S$, and let $x$ be a joint of $\alpha$. The sheet $S$ restrains $\alpha$ at $x$ if for all sufficiently small open intervals $(s,t)$ containing $x$, the path $\alpha(s) \rightarrow \alpha(t)$ leaves $S$. If $S$ restrains $\alpha$ at $x$, then $\alpha(x)$ is a vertex of a fringe of $S$, and $\alpha$ turns at $x$. We say $\alpha$ is **tight in $S$** if $S$ restrains $\alpha$ at each of its joints.

**Lemma 3d.5.** Elastic chains are piecewise linear and tight.

**Proof.** Let $\alpha$ be a path in a sheet $S$, and let $\rho$ be any elastic chain for $\alpha$. The lemma is trivial if $\rho$ is constant, so assume otherwise. We show that for every $x \in I$, either $\rho$ is straight at $x$ or $\rho$ is bent at $x$. In either case there is an interval $[s,t]$ containing a neighborhood of $x$ such that $\rho_{st}$ is bent. Since $I$ is compact, finitely many such intervals cover $I$, and it follows that $\rho$ is piecewise straight. The key fact we use is that every point $y$ in the sheet $S$ has a neighborhood that is starlike about $y$. 

---

107
Let \( x \) be a point of \([0, 1]\), and choose a neighborhood \( U \subseteq S \) of \( \rho(x) \) that is starlike about \( \rho(x) \). Because \( \rho \) is continuous, all points \( s \in I \) sufficiently close to \( x \) satisfy \( \rho(s) \in U \), implying that the linear path \( \sigma = \rho(x) \triangleright \rho(s) \) lies in \( U \). Because \( U \) is starlike, it is contractible and hence simply connected (Lemma 2a.8). Therefore \( \sigma \) and \( \rho_{x:s} \) are path-homotopic (Lemma 2a.5). By Lemma 3d.3, the path \( \sigma \) is the unique elastic path in its path class. Since \( \rho_{x:s} \) is elastic, by Lemma 3d.2, it follows that \( \rho_{x:s} = \sigma \). And since \( \rho \) is canonical, its subpath \( \rho_{x:s} \) is not constant, and so \( \rho_{x:s} \) is straight. We conclude that \( \rho \) is straight at \( x \) if \( x \in \{0, 1\} \), and a little further reasoning shows that \( \rho \) is bent at \( x \) if \( x \in (0, 1) \). Thus \( \rho \) is piecewise straight.

Now let \( x \) be a joint of \( \rho \); we show that \( S \) restrains \( \rho \) at \( x \). Let \((s, t)\) be an interval containing \( x \) such that \( \rho_{s:t} \) is bent. I show that for some interval \((s', t') \subseteq (s, t)\) the path \( \rho(s') \triangleright \rho(t') \) does not run in \( S \). Let \( C \) denote the convex hull of the points \( \rho(s), \rho(t), \rho(x) \). Because \( C \) is convex, it is simply connected. Hence if \( C \subseteq S \), then the path \( \sigma = \rho(s) \triangleright \rho(t) \) would be path-homotopic to \( \rho_{s:t} \) as paths in \( S \). Since \( \rho_{s:t} \) and \( \sigma \) are both elastic, they would have to be equal. But \( x \) is a joint of \( S \), and so \( \rho_{s:t} \) cannot equal the linear path \( \sigma \). Therefore \( C \not\subseteq S \), which implies that some linear path between \( \rho_{s:x} \) and \( \rho_{x:t} \) leaves \( S \). Since the interval \((s, t)\) was arbitrary, we conclude that \( S \) restrains \( \rho \) at \( x \). Thus \( \rho \) is tight in \( S \). \( \square \)

The fifth and final step establishes the uniqueness property. It also shows something more, namely that for canonical paths, tightness implies elasticity.

**Lemma 3d.6.** Let \( \kappa \) be a canonical, tight chain for a canonical path \( \sigma \). Then \( \|\kappa\| \leq \|\sigma\| \), with strict inequality if \( \|\cdot\| = |\cdot| \) and \( \kappa \not= \sigma \).

**Proof.** Let \( \overline{\kappa} \) and \( \overline{\sigma} \) be path-homotopic lifts of \( \kappa \) and \( \sigma \). By Lemma 3a.1, we can assume that \( \overline{\sigma} \) is simple, or else \( \|\sigma\| \) could be reduced without changing \( [\sigma]_{\rho} \).

The lifting \( \overline{\kappa} \) is also simple, because \( \kappa \) is tight. For if \( \overline{\kappa} \) were not simple, either two consecutive segments of \( \overline{\kappa} \) would overlap, or some subpath of \( \overline{\kappa} \) would form a simple loop, and \( \overline{\kappa} \) would have to turn toward the inside of this loop at least once (Corollary 3c.7). But since \( \kappa \) is tight, \( \overline{\kappa} \) only turns toward fringes, and there are no fringes inside a simple loop (Proposition 3c.5). Therefore both \( \overline{\sigma} \) and \( \overline{\kappa} \) are simple and canonical, and they have the same endpoints. It follows that if \( \overline{\kappa} \) and \( \overline{\sigma} \) have the same image, then the two paths are equal. In this case \( \kappa = \sigma \) and we are done. So we assume \( \text{Im} \overline{\kappa} \neq \text{Im} \overline{\sigma} \) and prove \( \|\kappa\| \leq \|\sigma\| \) with strictness if \( \|\cdot\| = |\cdot| \).

Let \((a, s)\) be the first crossing at which \( \overline{\sigma} \) leaves \( \text{Im} \overline{\kappa} \), and let \((b, t)\) be the next crossing at which they rejoin. Then the paths \( \overline{\kappa}_{a:b} \) and \( \overline{\sigma}_{t:s} \) intersect at their endpoints alone, and their concatenation is a simple loop \( \lambda \). We find a linear path in the blanket from \( \overline{\sigma}(s) \) to a point \( \overline{\sigma}(x) \); it will share a segment with \( \overline{\kappa} \). Because the blanket is simply connected, this path will be path-homotopic to \( \overline{\sigma}_{s:x} \). If we replace \( \overline{\sigma}_{s:x} \) by \( \overline{\sigma}(s) \triangleright \overline{\sigma}(x) \), its arc length in the norm \( \|\cdot\| \) will not increase; if \( \|\cdot\| \) is
the euclidean norm, then its arc length will actually decrease. Furthermore, \( \tilde{\sigma} \) will share one more segment of \( \tilde{\kappa} \). By repeated modifications of this kind, the path \( \tilde{\sigma} \) will converge to \( \tilde{\kappa} \).

**Figure 3d-1. Why elastic chains are shortest in every norm.** Wherever the paths \( \tilde{\kappa} \) and \( \tilde{\sigma} \) form a loop, as here with \( \tilde{\kappa}_{a:b} = \tilde{\sigma}_{s:t} \), we have \( \| \kappa_{a:b} \| \leq \| \sigma_{s:t} \| \) by repeated use of the polygon inequality.

Let \( N \) denote the inside component of the simple loop \( \lambda = \tilde{\kappa}_{a:b} \ast \tilde{\sigma}_{t:b} \). Because \( \kappa \) is tight, its lift \( \tilde{\kappa} \) cannot turn toward \( N \) at any point in \((a, b)\). If \( \kappa_{a:b} \) is straight, then we are done; put \( x = t \). Otherwise let \( c \) be the first point in \((a, b)\) at which \( \kappa \) turns, and extend the path \( \tilde{\kappa}_{a:c} \) linearly into \( N \). Eventually it must hit \( N \) again, either at \( \tilde{\kappa}(x) \) for \( x \in (c, b] \), or at \( \tilde{\sigma}(x) \) for \( x \in (s, t] \). In the latter case, we have the desired linear path \( \tilde{\sigma}(s) \circ \tilde{\sigma}(x) \). The former case is ruled out, for the resulting simple loop \( \tilde{\kappa}_{c:x} \ast \tilde{\kappa}(x) \circ \tilde{\kappa}(c) \) could turn toward its inside only at the two points \( \kappa(x) \) and \( \kappa(c) \), whereas Corollary 3c.7 requires three such turning points. \( \square \)

Two important results follow from Lemma 3d.6.

**Corollary 3d.7.** The elastic chain of a path \( \alpha \) is the unique canonical, tight chain in \([\alpha]_P \). \( \square \)

**Corollary 3d.8.** The elastic chain \( \kappa \) for a path \( \sigma \) satisfies \( \| \kappa \| \leq \| \sigma \| \) for any norm \( \| \cdot \| \). \( \square \)
Chapter 4

Flow Across Cuts and Half-Cuts

The results of the next four chapters concern a model of single-layer wiring based on the relation of link homotopy in sheets. This model represents a layer of an integrated circuit or printed circuit board by a structure called a design. The term 'design' should be taken in the sense of 'pattern' or 'drawing', not in the sense of 'specification'. Like a sketch, a design embodies only the geometry and topology of a circuit layer, and none of its functionality. Table 4-1 records the correspondence between the elements of the design model and those of the sketch model. Logically, the design model is prior to the sketch model in that all my results about sketches are justified by relating them to analogous results about designs.

The purpose of this chapter is to lay the groundwork for the constructions and theorems that characterize routability and optimal routings of designs. (We will not reach those theorems until the middle of Chapter 6.) It begins by defining the design model and the concepts we use in analyzing it, and it proceeds to develop a detailed theory of the design model. This theory is not an outgrowth of any existing body of mathematics. It deals primarily with the properties and relations of cuts and wires that it invents. Nothing you have seen before will make its results obvious, although a familiarity with topology helps. It does, however, share with the sketch model a concern for the congestions and capacities of cuts, and the main results of this chapter can be understood in those terms.

As its title suggests, this chapter centers around the concept of flow. Flow is an abstraction that is similar to, but more versatile than, the concept of congestion we used up through Chapter 1. After defining the design model in Section 4A, we spend a section exploring the various equivalent definitions of flow and the relationship of flow to congestion. The flow across a cut is strongly related to the necessary crossings of the cut by wires, which we also define in Section 4B. We prove in Proposition 4b.3 that link-homotopic cuts have equal flow, and in Proposition 4b.3 that the flow across a simple cut equals its congestion. Later, in Section 4D, we define the concept of a half-cut for route of a wire, and extend the definition of flow to encompass half-cuts. We then prove an important formula (Proposition 4d.2) relating the flow across a cut to to the flows of the half-cuts it includes. Finally,
Section 4F shows how to relate the flow and capacity of a cut to the flows and capacities in the links of a chain for that cut. We thereby obtain conditions under an unsafe simple cut or half-cut can be reduced to an unsafe straight cut or half-cut.

Comparing the two models

Sketches and designs differ in two major respects. First, we use the fringes of a sheet to represent the terminals and routing obstacles of a design, and hence these objects have positive size. Second, in a design we consider cuts that are not straight. Cuts and wires in the design model are links in a sheet, and they have the homotopy relation of links. Most terms, including capacity, congestion, empty, entanglement, proper, routable, route, safe, self-avoiding, terminal, and width, have approximately the same meaning in both models.

<table>
<thead>
<tr>
<th>Sketch model</th>
<th>Design model</th>
<th>Sketch Model</th>
<th>Design Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>feature</td>
<td>fringe</td>
<td>island</td>
<td>fringe</td>
</tr>
<tr>
<td>trace</td>
<td>wire</td>
<td>bridge</td>
<td>link</td>
</tr>
<tr>
<td>element</td>
<td>detail</td>
<td>bridge-homotopic</td>
<td>link-homotopic</td>
</tr>
<tr>
<td>realization</td>
<td>embedding</td>
<td>routing region</td>
<td>sheet</td>
</tr>
<tr>
<td>cut</td>
<td>straight cut</td>
<td>territory</td>
<td>extent</td>
</tr>
</tbody>
</table>

Table 4-1. The correspondence between the sketch and design models. Concepts that have the same name in both models are not shown.

The design model encompasses several ideas of what constitutes a proper design. To each there corresponds a routability theorem saying that a design is routable if and only if all of a certain class of straight cuts are safe. This class always excludes trivial cuts that are path-homotopic to paths in fringes. If one requires that the fringes of a proper design be self-avoiding, then the class includes all cuts with one terminal that are not trivial. If one allows the terminals of a wire in a proper design to be arbitrarily close, then the class excludes all cuts that are link-homotopic to wires. The most natural design model differs from the sketch model in these two ways. In order to support both models, we use the most permissive definition of a proper sketch, one in which fringes need not be self-avoiding and the terminals of a wire can be arbitrarily close.

4A. The Design Model

This section defines the design model and states the theorems that we set out to prove. These theorems, the design routability theorem and the design routability
theorem, are the deepest results of the design model and the precursors of the corresponding theorems about sketches.

A design is essentially a set of disjoint simple links in a sheet, each one representing a wire. For technical reasons, however, we place some restrictions on these links and their terminals. A fringe $F$ of a sheet $S$ is called inner if $\text{inside}(F) \subset R^2 - S$, and otherwise $F$ is outer. Every sheet has exactly one outer fringe and one or more inner fringes. A wire in a sheet $S$ is a simple link in $S$ with two convex inner fringes as terminals. A design on a sheet $S$, usually denoted $\Omega$, is a finite set of wires in $S$ whose images are disjoint and whose terminals are all distinct. The details of the design $\Omega$ are its wires and the fringes of $S$. An article of $\Omega$ is either a fringe of $S$ that is not a terminal of $\Omega$, or the union of the terminals and the image of some wire in $\Omega$. Equivalently, an article of $\Omega$ is a component of the space $\text{Bd} S \cup \bigcup_{\omega \in \Omega} \text{Im} \omega$.

**Figure 4a-1.** A design and one of its embeddings. Panel (a) represents a 4-wire design on a sheet with 10 fringes. The dark polygons represent the fringes; the space inside the inner fringes and outside the outer fringe does not belong to the sheet. The inner fringes need not have the same shape, although they do in this example. Part (b) shows an embedding of the design at left: the sheets of the two designs are identical, and their wires are in bijective correspondence, with corresponding wires being link-homotopic.

Parallel to the concept of realization for sketches is the concept of embedding for designs. And as bridge homotopy governs the routing of traces, link homotopy governs the routing of wires. A link that is link-homotopic to a wire $\omega$ is called a route of $\omega$. If this link is a wire, we call it an embedding of $\omega$. If $\Omega$ and $\Upsilon$ are designs on the same sheet, we say $\Upsilon$ is an embedding of $\Omega$ if there exists a bijection $f: \Omega \to \Upsilon$ such that $\omega \simeq_L f(\omega)$ for every wire $\omega \in \Omega$. The embedding relation is an equivalence relation among the designs on a sheet.

The main problem concerning designs is that of finding a proper embedding for a design: an embedding that represents a legal circuit layer. As with a sketch,
whether a design is proper depends upon the widths of its details. We assume that the design associates a positive width with each wire and fringe, with one important condition: no wire may be wider than either of its terminals. A route of a wire is always considered to have the same width as the original wire.

There are two ways a design can be improper. First, two of its articles may come too close. The extent of a detail $F$ of width $d$ is the set of points in $\mathbb{R}^2$ lying within $d/2$ units of $F$. Distances here are measured by a piecewise linear wiring norm, denoted $\| \cdot \|$, that is a parameter of the entire model. The extent of an article is the union of the extents of its details. Different articles should have disjoint extents. Second, one of the wires of the design can have an undesirable shape. A subset $X$ of $\mathbb{R}^2$ is said to divide a sheet $S$ if two fringes of $S$ fall in different components of $\mathbb{R}^2 - X$. An article of a design $\Omega$ on a sheet $S$ is called divisive if its extent divides $S$. Every wire should be self-avoiding, meaning that its article should not be divisive.

To summarize: A design is proper if (1) its articles have disjoint extents, and (2) its wires are self-avoiding. A design is routable if it admits a proper embedding, and the wires in the proper embedding are called feasible embeddings of the wires in the original design.

![Figure 4a.2. The extents of a design's details. This figure shows the thicknesses of the wires and fringes of the design in Figure 4a-1. As this drawing makes clear, the wires in that design were not routed arbitrarily. In fact, the embedding shown in Figure 4a-1 is optimal with respect to a certain octagonal norm, namely that in which each inner fringe is the set of points of distance 1 from its center of symmetry. By ‘optimal’ I mean that the embedding is proper and that no other proper embedding improves on the length of any wire.](image)

**Cuts and crossings**

We analyze the routability of a design in terms of the congestions and capacities of cuts. The definition of cut in the design model is very general: a cut of a sheet $S$
is a link in $S$ whose liftings to the sheet's blanket are simple. (Because the liftings of a link are related by covering transformations, either all the liftings of the link are simple, or none are.) Thus all simple links in the sheet, and all straight links in particular, are cuts. Let $\chi$ be a cut of $S$, and let $\Omega$ be a design on $S$. If the terminals of $\chi$ are $X$ and $Y$, then the capacity of $\chi$ in $\Omega$ is

$$cap(\chi, \Omega) = ||\chi|| - width(X)/2 - width(Y)/2,$$

where $||\chi||$ is the arc length of $\chi$ in the wiring norm. We often abbreviate the notation $cap(\chi, \Omega)$ to $cap(\chi)$, for we shall never compare two designs that assign different widths to fringes.

Before defining the congestion of a cut, we need a precise notion of crossing. Since cuts can have self-intersections, we must count crossings according to multiplicity. If $\alpha$ and $\beta$ are paths, a crossing of $\alpha$ by $\beta$ is a pair $(s, t) \in I \times I$ such that $\alpha(s) = \beta(t)$. The pair $(s, t)$ is ordered; $(t, s)$ would be a crossing of $\beta$ by $\alpha$. The number of crossings between $\alpha$ and $\beta$ is denoted $cross(\alpha, \beta)$. Of course, the set of crossings can be infinite or even uncountable, but in the cases of interest it will be finite. The entanglement of a cut $\chi$ by a wire $\omega$ is defined to be the minimum number of crossings of $\chi$ by a route of $\omega$. In symbols,

$$tangle(\chi, \omega) = \min\{cross(\chi, \omega') : \omega' \simeq_L \omega\}.$$  

Because cuts are piecewise linear, entanglement is always finite. The congestion of $\chi$ in the design $\Omega$, denoted $cong(\chi, \Omega)$, is the total entanglement of $\chi$ by wires in $\Omega$, where each crossing is weighted according to the width of its wire. Formally, we have

$$cong(\chi, \Omega) = \sum_{\omega \in \Omega} width(\omega) \cdot tangle(\chi, \omega).$$

A simple cut is called unsafe if its congestion exceeds its capacity, and safe otherwise. Safety for nonsimple cuts is defined in Section 4F.

The intuitive meaning of congestion is this: If $\chi$ is a simple cut in a design $\Omega$, then in any proper embedding of $\Omega$, the portion of $\chi$ within the extents of wires will have total arc length at least $cong(\chi, \Omega)$. If this quantity is positive, and exceeds the capacity of $\chi$, then no proper embedding of $\Omega$ can exist. Similarly, if the capacity of $\chi$ is negative, then the terminals of $\chi$ have overlapping extents. If these terminals lie in different articles, then $\Omega$ is again unroutable.

These and similar considerations motivate our definition of a major cut, one whose safety is necessary for the design to be routable. We say that a link $\chi$ is degenerate in $\Omega$ if $\chi$ is path-homotopic to a path in a single article of $\Omega$. A cut $\chi$ is empty in $\Omega$ if $cong(\chi, \Omega) = 0$ and $\chi$ has only one terminal. Degenerate and empty
cuts are called minor; others are major. The thin lines in Figure 4a.1 are major straight cuts whose flow and capacity are equal. If any of these cuts were shorter, that design would be unroutable.

Central results concerning designs

I prove two major theorems in the design model: one concerns routability, and the other concerns routing. Chapter 8 uses these two theorems to prove the sketch routability theorem and the sketch routing theorem of Section 1A. The definitions in this section are arranged so as to permit a very simple characterization of routable designs.

Theorem 6c.1. (Design Routability Theorem) A design $\Omega$ on the sheet $S$ is routable if and only if every major straight cut in $S$ is safe in $\Omega$.

If every major straight cut in $S$ is safe in $\Omega$, we say that $\Omega$ is safe. The design routability theorem has two parts: safe designs are routable (Theorem 5e.6), and unsafe designs are unroutable (Theorem 6a.5). The latter claim is the easier, and is proved in Section 6A.

The hard direction of the design routability theorem follows from a deeper result. It depends on the construction, presented in Section 5A, of an ideal embedding of every wire in a safe design. The ideal embedding of a wire is the shortest route for that wire that leaves enough space for other wires to be routed. Formally, it has minimum euclidean arc length among all routes for the wire whose nontrivial, straight half-cuts are safe.

Theorem 6c.2. (Design Routing Theorem) The ideal embeddings of the wires in a safe design form a proper design, and they have minimal euclidean arc length among all feasible embeddings of those wires.

In other words, when routing a safe design one can do no better than to use the ideal embedding of each wire. The proof of the design routing theorem occupies Chapter 5 and Section 6B.

4B. Flow: A Characterization of Congestion

Thanks to Chapters 2 and 3, we already have many tools for examining designs. We use them here to define formally the concept of a necessary crossing. As a consequence we are able to make sense of the congestion of nonsimple cuts. We characterize the congestion of a simple cut in terms of its necessary crossings by wires, and derive a statistic called the flow across a cut which agrees with congestion for simple cuts. The definition of flow turns out to be much more useful than the
original definition of congestion, in part because it makes sense for cuts that are not simple, and in part because the topological machinery of Chapter 3 applies powerfully to the liftings and crossings that define flow. This power shows up immediately in the proof of Proposition 4b.3, which says that link-homotopic cuts have equal flow.

**Necessary crossings in blankets**

Intuitively, a necessary crossing between two links is one that cannot be removed by a link homotopy. Given two links in a blanket, one can tell whether they necessarily cross by examining their fringes.

**Definition 4b.1.** A simple link $\alpha$ in a blanket $M$ cuts another link $\beta$ in $M$ if
1. the endpoints of $\alpha$ and $\beta$ lie on four distinct fringes of $M$, and
2. the endpoints of $\beta$ lie in different scraps of $M - \text{Im } \alpha$.

If $\alpha$ cuts $\beta$, then $\text{Im } \beta$ must intersect $\text{Im } \alpha$. For $\text{Im } \beta$ is a connected set; if it did not intersect $\text{Im } \alpha$ it would lie entirely in one component of $M - \text{Im } \alpha$. Furthermore, whether or not $\alpha$ cuts $\beta$ depends only on the terminals of $\beta$, and not on any other properties of $\beta$. Hence if $\beta$ is link-homotopic to another link $\beta'$, then

$$\alpha \text{ cuts } \beta \iff \alpha \text{ cuts } \beta',$$

since (by Corollary 3a.5) $\beta$ and $\beta'$ have the same terminals. Thus if $\alpha$ cuts $\beta$, they make a crossing that cannot be removed by applying a link homotopy to $\beta$.

On the other hand, if $\alpha$ does not cut $\beta$, the crossing (if any) between $\alpha$ and $\beta$ can be removed by applying a link homotopy to $\beta$. For if $\alpha$ does not cut $\beta$, then either (1) $\beta$ shares a terminal with $\alpha$, or else (2) the terminals of $\beta$ lie on the same side of $\alpha$. In either case, there is a link $\beta'$ with the same terminals as $\beta$ but whose endpoints lie in the same scrap of $M - \text{Im } \alpha$. By Proposition 3a.3 we can assume that $\beta'$ is a link in that scrap, so that $\alpha$ and $\beta'$ do not cross. Corollary 3a.5 implies that $\beta' \simeq_L \beta$. Thus the relation "$\alpha$ cuts $\beta$" captures the intuitive notion that "$\beta$ makes a necessary crossing with $\alpha$".

The cutting relation has several other nice properties. If $\alpha'$ is simple and link-homotopic to $\alpha$, then

$$\alpha \text{ cuts } \beta \iff \alpha' \text{ cuts } \beta,$$

because homotopic simple links separate the fringes identically (Proposition 3c.4). Moreover, if both $\alpha$ and $\beta$ are simple, then the relation "$\alpha$ cuts $\beta$" is symmetric. For if $\alpha$ does not cut $\beta$, then as shown above, some link $\beta' \in [\beta]_L$ lies in a single scrap of $\alpha$. Clearly $\beta'$ does not cut $\alpha$, because their images are disjoint. Hence $\beta$ does not cut $\alpha$. We conclude that when $\alpha$ and $\beta$ are simple,

$$\alpha \text{ does not cut } \beta \implies \beta \text{ does not cut } \alpha,$$
and the converse also holds by symmetry. Hence ‘\( \alpha \) cuts \( \beta \)’ is a symmetric relation if \( \alpha \) and \( \beta \) are simple.

**Necessary crossings in sheets**

The notion of necessary crossing for links in a blanket carries over to links in a sheet. To determine whether a crossing between two links in a sheet is necessary, we lift those links to the blanket in such a way that the lifts cross at the same point the original links cross, and check whether one lift cuts the other. The elegance and usefulness of this definition are two major motivations for using blankets to study wire routing.

**Definition 4b.2.** Let \( \omega \) be a link in a sheet \( S \), and let \( M \) be the blanket of \( S \) with covering map \( p: M \to S \). Let \( \chi \) be cut in \( S \), and let \( \tilde{\chi} \) be any lift of \( \chi \) to \( M \). Suppose that \( (s, t) \) is a crossing of \( \chi \) by \( \omega \). Because \( p(\tilde{\chi}(s)) = \omega(t) \), the link \( \omega \) has a unique lift \( \tilde{\omega} \) such that \( \tilde{\chi}(s) = \tilde{\omega}(t) \). We say that \( \tilde{\chi} \) and \( \tilde{\omega} \) reflect the crossing \( (s, t) \). The crossing \( (s, t) \) of \( \chi \) by \( \omega \) is necessary if \( \tilde{\chi} \) cuts \( \tilde{\omega} \). Two crossings of \( \chi \) by \( \omega \) are similar if the corresponding lifts of \( \omega \) are identical.

The initial choice of \( \tilde{\chi} \) is irrelevant; it amounts to a choice of base point for the blanket, and as shown in Section 2B this choice does not affect the topology. If one chooses two different lifts of \( \chi \), say \( \tilde{\chi} \) and \( \tilde{\chi}' \), then one obtains different lifts \( \tilde{\omega} \) and \( \tilde{\omega}' \) of \( \omega \), and Proposition 2b.7 gives us a covering transformation \( h: M \to M \) such that \( h \circ \tilde{\chi} = \tilde{\chi}' \) and \( h \circ \tilde{\omega} = \tilde{\omega}' \). Since the relation ‘\( \tilde{\chi} \) cuts \( \tilde{\omega} \)’ depends only on topological properties of \( M \), \( \tilde{\chi} \), and \( \tilde{\omega} \), which are preserved by the homeomorphism \( h \), the link \( \tilde{\chi} \) cuts \( \tilde{\omega} \) if and only if \( \tilde{\chi}' \) cuts \( \tilde{\omega}' \). Hence necessity for crossings is well defined, and by similar reasoning, similarity is also. The technique of lifting links to reflect certain crossings among them will appear in future definitions, and we shall normally take for granted the fact that the choice of the first lifting—though not the choice of later liftings—is immaterial.

A definition equivalent to Definition 4b.2 would hold \( \tilde{\omega} \) fixed and vary \( \tilde{\chi} \) according to the crossing.

By counting necessary crossings we obtain a measure of the entanglement of two links. Two immediate consequences of Definition 4b.2 are that similarity of crossings is an equivalence relation, and that two similar crossings are either both necessary or both unnecessary. We define the quantity \( \text{wind}(\chi, \omega) \), the winding of \( \chi \) and \( \omega \), to be the number of similarity classes of necessary crossings between \( \chi \) and \( \omega \). For any lift \( \tilde{\chi} \) of \( \chi \), it is the number of lifts of \( \omega \) that are cut by \( \tilde{\chi} \). (Each such lift makes crossings with \( \tilde{\chi} \), and these crossings form a similarity class of necessary crossings of \( \chi \) by \( \omega \); conversely, every similarity class corresponds to a particular lift of \( \omega \) that is cut by \( \tilde{\chi} \).) Equivalently, since cutting is symmetric, \( \text{wind}(\chi, \omega) \) is, for
Figure 4b-1. Necessity and similarity of crossings. In part (i), the wire $\omega$ makes six crossings with the cut $\chi$. The crossings fall into three similarity classes: crossings 1, 2, and 3 are similar and necessary; crossing 4 is necessary but not similar to the others; crossings 5 and 6 are similar and unnecessary. Parts (ii) through (iv) show the lifts of $\omega$ that correspond to these crossings, drawn in a fashion that emphasizes the covering map. Portions of these lifts are dotted to show that they run on a different level of the blanket from the lifting of $\chi$. Part (v) summarizes the liftings of $\omega$ in a way that emphasizes their topology. Because $\omega$ is simple, these liftings do not intersect.
any lift $\tilde{\omega}$ of $\omega$, the number of lifts of $\chi$ that cut, or are cut by, $\tilde{\omega}$. The winding of $\chi$ and $\omega$ in Figure 4b-1 is 2.

Summing the winding of $\chi$ over the wires in a design $\Omega$, and weighting each number according to the width of the wire, we obtain a measure of congestion. I call it the flow across $\chi$ in the design $\Omega$:

$$\text{flow}(\chi, \Omega) = \sum_{\omega \in \Omega} \text{width}(\omega) \text{wind}(\chi, \omega).$$

The flow statistic is invariant under link homotopy, both of wires and of cuts. Thus if $\Upsilon$ is an embedding of the design $\Omega$, then $\text{flow}(\chi, \Upsilon) = \text{flow}(\chi, \Omega)$. Similarly, if $\alpha$ and $\beta$ are link-homotopic cuts in the sheet of $\Omega$, then $\text{flow}(\alpha, \Omega) = \text{flow}(\beta, \Omega)$. To emphasize this fact, I give it a formal proof.

**Proposition 4b.3.** Link-homotopic cuts have equal flow.

**Proof.** Let $\alpha$ and $\beta$ be cuts of a sheet $S$, and let $\omega$ be a wire in $S$. Lift $\omega$ to the blanket of $S$, obtaining a link $\tilde{\omega}$. The flow of $\omega$ across $\alpha$ is the number of lifts $\tilde{\alpha}$ of $\alpha$ that cut $\tilde{\omega}$. Similarly, the flow of $\omega$ across $\beta$ is the number of lifts $\tilde{\beta}$ of $\beta$ that cut $\tilde{\omega}$. Assume now that $\alpha$ and $\beta$ are link-homotopic. By Proposition 3a.6, there is a bijective correspondence between the lifts of $\alpha$ and the lifts of $\beta$ such that corresponding lifts are link-homotopic. Hence if a lift of $\alpha$ cuts $\tilde{\omega}$, so does the corresponding lift of $\beta$, and vice versa. Therefore $\text{wind}(\alpha, \omega) = \text{wind}(\beta, \omega)$. Since this holds for all wires $\omega$, it follows that $\text{flow}(\alpha, \Omega) = \text{flow}(\beta, \Omega)$ for any design $\Omega$ on $S$. □

Proposition 4b.3 allows us to extend the concept of flow to all links. Since all cuts in a link-homotopy class have the same flow, and path-homotopic links are also link-homotopic, it suffices to define the flow of a link to be the flow of any path-homotopic cut. This works because every path class of links in a sheet contains a cut. For if $\alpha$ is a link with lifting $\tilde{\alpha}$, there is by Proposition 3a.3 a simple link $\tilde{\beta}$ between the endpoints of $\tilde{\alpha}$. By Lemma 2a.5, $\tilde{\beta}$ is path-homotopic to $\tilde{\alpha}$, and hence the projection $\beta$ of $\tilde{\beta}$ to the sheet is path-homotopic to $\alpha$. The link $\beta$ is a cut because $\tilde{\beta}$ is simple.

**Flow and congestion**

We now address the question of how congestion compares to flow. The answer is that flow is never greater than congestion, and for simple cuts they are equal. The following two lemmas clarify the relationship between flow and congestion.

**Lemma 4b.4.** Let $\chi$ be a cut of a sheet $S$. Every wire $\omega$ in $S$ satisfies $\text{tangle}(\chi, \omega) \geq \text{wind}(\chi, \omega)$.
Flow Across Cuts and Half-Cuts

Proof. Let \( M \) be a blanket of \( S \) with covering map \( p: M \to S \). Denote by \( n \) the winding of \( \omega \) and \( \chi \). Let \( \tilde{\omega} \) be any lift of \( \omega \) to \( M \). Every necessary crossing of \( \chi \) by \( \omega \) represents a lift of \( \chi \) that cuts \( \tilde{\omega} \); dissimilar crossings correspond to different lifts of \( \chi \). Let \( \tilde{\chi}_1, \ldots, \tilde{\chi}_n \) be the lifts of \( \chi \) that cut \( \tilde{\omega} \). Let \( v \) be any route of \( \omega \); we show that \( \text{cross}(\chi, v) \geq n \), thus proving that \( \text{tangle}(\chi, \omega) \geq n \). Using Proposition 3a.6, lift \( v \) to a link \( \tilde{v} \in [\tilde{\omega}]_L \). Then for \( 1 \leq i \leq n \), the link \( \tilde{\chi}_i \) cuts \( \tilde{v} \), so we have \( \tilde{\chi}_i(s_i) = \tilde{v}(t_i) \) for some \( s_i, t_i \in I \). Projecting to the sheet, we see that each pair \( (s_i, t_i) \) is a crossing of \( \chi \) by \( v \). All these crossings are distinct. If \( t_i = t_j \) for some \( i \) and \( j \), then \( \tilde{\chi}_i(s_i) = \tilde{\chi}_j(s_j) \), so by uniqueness of liftings, we cannot also have \( s_i = s_j \) unless \( i = j \). We conclude that \( \text{cross}(\chi, v) \geq n \) as claimed. \( \square \)

The other direction is somewhat harder, and it fails for cuts that are not simple, as shown in Figure 4b-2.

**Figure 4b-2.** A cut whose flow and congestion differ. The flow of \( \omega \) across \( \chi \) is zero, because \( \chi \) is homotopic to a cut (striped path) that does not intersect \( \omega \). The entanglement of \( \chi \) with \( \omega \) is nonzero, however; every link that is homotopic to \( \omega \) crosses \( \chi \) at least twice.

**Lemma 4b.5.** Let \( \Gamma \) be a set of disjoint simple cuts in a sheet \( S \), and let \( \omega \) be a link in \( S \). There is a link \( v \in [\omega]_L \) such that \( \text{cross}(\gamma, v) = \text{wind}(\gamma, v) \) for all \( \gamma \in \Gamma \).

Proof. Let \( M \) be a blanket of \( S \) with covering map \( p: M \to S \), and let \( \tilde{\omega} \) be any lift of \( \omega \) to \( M \). Now let \( \tilde{\gamma}_1, \ldots, \tilde{\gamma}_n \) be the links in \( M \) that lift elements of \( \Gamma \) and cut \( \tilde{\omega} \). (Here \( n = \sum_{\gamma \in \Gamma} \text{wind}(\gamma, \omega) \).) Two lifts \( \tilde{\gamma}_i \) and \( \tilde{\gamma}_j \) cannot intersect unless \( i = j \). For if \( \tilde{\gamma}_i(s) = \tilde{\gamma}_j(t) \), then \( \tilde{\gamma}_i \) and \( \tilde{\gamma}_j \) lift the same link \( \gamma \), since the elements of \( \Gamma \) are disjoint. Thus \( \gamma(s) = \gamma(t) \), whence \( s = t \) because \( \gamma \) is simple, and thence \( \tilde{\gamma}_i = \tilde{\gamma}_j \) by uniqueness of liftings. Figure 4b-3 illustrates the case where \( \Gamma \) contains a single cut \( \chi \).

We construct a path \( \tilde{v} \) that crosses each lifting \( \tilde{\gamma}_i \) exactly once, and does not intersect any other lifting of any cut in \( \Gamma \). In view of Lemma 4b.4, the conclusion will follow at once. Let \( A \) and \( B \) be the terminals of \( \tilde{\omega} \). Denote by \( L_i \) and \( R_i \) the scraps of \( M - \text{Im} \tilde{\gamma}_i \) that contain \( A \) and \( B \), respectively. When \( i \neq j \), the thread \( \text{Im} \tilde{\gamma}_j \) must lie entirely in \( L_i \) or in \( R_i \). By renumbering the lifts of \( \gamma \), we may assume that \( \text{Im} \tilde{\gamma}_j \subseteq R_i \) whenever \( j > i \). For each \( i \) such that \( 1 \leq i < n \), the set \( L_{i+1} \cap R_i \) is a scrap—one component of \( R_i - \text{Im} \gamma_{i+1} \).
Figure 4b-3. Why a simple cut’s congestion does not exceed its flow. The \( n \) lifts of the simple cut \( \chi \) that cut \( \bar{\omega} \) decompose the blanket into \( n + 1 \) parts. One can construct a link in \([\bar{\omega}]_L\) that crosses these lifts only once each: the concatenation of the paths \( \alpha_i \) and \( \lambda_i \) is such a link. If this link crosses any other lifts of \( \chi \) (thin lines), the crossings can be removed by inserting detours as shown.

First we establish the points at which \( \bar{\nu} \) crosses the lifts \( \bar{\gamma}_i \). For \( 1 \leq i \leq n \), choose a straight path that crosses \( p \circ \bar{\gamma}_i \) exactly once and crosses no other cuts in \( \Gamma \). Let \( \lambda_i \) be a lifting of either this path or its reverse, such that \( \lambda_i(0) \in L_i \) and \( \lambda_i(1) \in R_i \). Then \( \lambda_i \) makes exactly one crossing with a lifting of a cut in \( \Gamma \); that lifting is \( \bar{\gamma}_i \). Now we connect up the segments \( \lambda_i \) with paths that intersect no liftings of cuts in \( \Gamma \). By Proposition 3a.3, there is a simple half-link \( \alpha_0 \) in \( L_1 \) from \( \lambda_1(0) \), and a simple reverse half-link \( \alpha_n \) in \( R_n \) from \( \lambda_n(1) \) to \( B \). For \( i = 1, 2, \ldots, n \), use Proposition 3a.3 to find a simple path \( \alpha_i \) in \( L_{i+1} \cap R_i \) from \( \lambda_i(1) \) to \( \lambda_{i+1}(0) \). Define \( \alpha \) to be the path

\[
\alpha = \alpha_0 \ast \lambda_1 \ast \alpha_1 \ast \cdots \ast \lambda_n \ast \alpha_n.
\]

Then \( \alpha \) crosses each of the lifts \( \bar{\gamma}_i \) exactly once. It may intersect some other lift of a cut in \( \Gamma \), however.

Now we modify the subpaths \( \alpha \) so that it intersects no liftings of cuts in \( \Gamma \) except \( \bar{\gamma}_1, \ldots, \bar{\gamma}_n \). Using the fact that \( \text{Im} \alpha \) is compact, one can check that it intersects only finitely many lifts of cuts in \( \Gamma \). By induction, therefore, it suffices to show that a single unwanted crossing of \( \alpha \) can be removed. Suppose that \( \alpha \) crosses some lift \( \bar{\chi} \notin \{\bar{\gamma}_1, \ldots, \bar{\gamma}_n\} \) of a cut \( \chi \in \Gamma \). Since \( \bar{\chi} \) is a simple link, it splits \( M \) into two scraps. And because \( \bar{\chi} \) does not cut \( \bar{\omega} \), at least one of these scraps contains portions of both \( A \) and \( B \). Let \( N \) be such a scrap. Replace the portions of \( \alpha \) that leave \( N \) by paths that skirt \( \bar{\chi} \) closely enough not to intersect any lifting of a cut in \( \Gamma \). (Such a skirting path may be constructed using a tubular neighborhood of \( \text{Im} \chi \) that intersects no other cut in \( \Gamma \).) The resulting path is a still piecewise linear link from \( A \) to \( B \), and it makes fewer crossings with lifts of cuts in \( \Gamma \) than it used to.
Eventually we obtain a piecewise linear link $\tilde{\nu}: A \rightarrow B$ whose projection $\nu$ makes at most $n$ crossings with the cuts in $\Gamma$. Of course, the number of crossings it makes is actually $n$, by Lemma 4b.4. Corollary 3a.5 says that $\tilde{\nu}$ is link-homotopic to $\tilde{\omega}$, and hence its projection $\nu$ is link-homotopic to $\omega$. □

If in Lemma 4b.5 we take $\Gamma$ to be the set containing a single cut $\chi$, we deduce that $\text{tangle}(\chi, \omega) \leq \text{wind}(\chi, \omega)$ whenever $\chi$ is simple. Combining this result with Lemma 4b.4, and summing over all the wires in a design, gives us the desired answer.

**Proposition 4b.6.** If $\chi$ is a cut of a design $\Omega$, then $\text{cong}(\chi, \Omega) \geq \text{flow}(\chi, \Omega)$, with equality if $\chi$ is simple. □

Our interest in congestion comes from the design routability theorem (6c.1), which involves only straight cuts. Since congestion and flow agree for all simple cuts, we are free to discard the former in favor of the latter. And as Proposition 4b.3 suggests, flow is the more natural concept, and is far easier to work with. Henceforth we use the flow statistic exclusively, except in Chapter 8 when proving the sketch routability theorem.

### 4C. Relations Among Cuts and Wires

The main results of this chapter concern the flows across cuts. In order to relate the flows of different cuts in the same design, we first study relationships among simple links in a blanket. Of particular concern is the relation of one link cutting another, which forms the basis for the definition of flow. This section gives a condition under which one link must cut another, stated in Lemma 4c.1 below, and several conditions under which two links cannot cut one another, such as when they lift routes for wires in the same design.

We will use the following result many times.

**Lemma 4c.1.** Let $\alpha$ and $\beta$ be simple links in a blanket such that $\alpha$ cuts $\beta$, and let $\gamma$ be a simple link from a terminal of $\alpha$ to a terminal of $\beta$. Every link that cuts $\gamma$ also cuts either $\alpha$ or $\beta$.

**Proof.** Without loss of generality we may replace $\alpha$, $\beta$, and $\gamma$ by link-homotopic simple links. We may also reverse $\alpha$, $\beta$, and $\gamma$ as desired. Choose simple links $\alpha$ and $\beta$ that intersect in one point only, say $\alpha(s) = \beta(t)$, and let $\gamma$ be the simple link $\alpha \ast \beta_{1,1}$. By the symmetry between left and right we may assume that $\beta(1) \in \text{left}(\alpha)$, as in Figure 4c-1, and it follows that $\alpha(0) \in \text{left}(\beta)$.

I claim $\text{right}(\gamma) = \text{right}(\alpha) \cup \text{right}(\beta)$. The connected set $\text{right}(\alpha)$ does not intersect $\text{Im} \, \gamma$, but it borders on $\gamma$ from the right at $\gamma(0)$. Hence $\text{right}(\gamma) \supseteq \text{right}(\alpha)$. Similarly $\text{right}(\beta)$ does not intersect $\text{Im} \, \gamma$, and it borders on $\gamma$ from the right at $\gamma(1)$.
Hence \( \text{right}(\gamma) \supset \text{right}(\beta) \). If a point \( x \) lies neither in \( \text{right}(\alpha) \) nor \( \text{right}(\beta) \), it must lie on \( \text{Im} \alpha - \text{right}(\beta) \), or on \( \text{Im} \beta - \text{right}(\alpha) \), or in \( \text{left}(\alpha) \cap \text{left}(\beta) \). In the first two cases, \( x \) falls on \( \text{Im} \gamma \). In the last case, draw a piecewise linear path from \( x \) to any point of \( \text{Im} \alpha \) or \( \text{Im} \beta \). The first point at which it intersects \( \text{Im} \alpha \cup \text{Im} \beta \) must lie on \( \text{Im} \gamma \), because \( \text{Im} \alpha - \text{Im} \gamma \subseteq \text{right}(\beta) \) and \( \text{Im} \beta - \text{Im} \gamma \subseteq \text{right}(\alpha) \), whereas \( x \) lies in \( \text{left}(\alpha) \cap \text{left}(\beta) \). Of course, the path intersects \( \alpha \) or \( \beta \) from the left. It follows that it intersects \( \gamma \) from the left, and hence \( x \) lies in \( \text{left}(\gamma) \). We conclude that \( \text{right}(\gamma) = \text{right}(\alpha) \cup \text{right}(\beta) \), and \( \text{left}(\gamma) = \text{left}(\alpha) \cap \text{left}(\beta) \).

Figure 4c-1. A link formed when two others cross. The link \( \gamma \), shown in grey, comprises parts of \( \alpha \) and \( \beta \). The left side of \( \gamma \) (dark shading) is the intersection of the left sides of \( \alpha \) and \( \beta \); the right side of \( \gamma \) (light shading) is the union of the right sides of \( \alpha \) and \( \beta \).

Now let \( \eta \) be a simple link that cuts \( \gamma \). Then one terminal of \( \eta \) lies entirely in \( \text{left}(\alpha) \cap \text{left}(\beta) \), and the other lies entirely in \( \text{right}(\alpha) \cup \text{right}(\beta) \). Call the second terminal \( X \). If \( X \) intersects \( \text{right}(\alpha) \), then either it lies entirely in \( \text{right}(\alpha) \), in which case \( \eta \) cuts \( \alpha \), or else it is a terminal of \( \alpha \). It cannot be the fringe containing \( \alpha(0) \), because this is a terminal of \( \gamma \). Hence \( X \) must be the fringe containing \( \alpha(1) \), which lies wholly in \( \text{right}(\beta) \). Then \( \eta \) cuts \( \beta \). Similarly, if \( X \) intersects \( \text{right}(\beta) \), then either \( X \subseteq \text{right}(\beta) \) or else \( X \) is the fringe containing \( \beta(0) \), which is a subset of \( \text{right}(\alpha) \). In either case \( \eta \) cuts \( \alpha \) or \( \beta \).

Liftings of wires and their routes

Many of the links we consider will be liftings of wires, or routes of wires, taken from the same design. Such links, if simple, are called coherent.

Definition 4c.2. Let \( \Upsilon \) be a set of links obtained by replacing each wire in a design by a route of that wire. If the links in \( \Upsilon \) have simple liftings, then any set of these liftings is called coherent. If \( \alpha \) and \( \beta \) are simple liftings of links in \( \Upsilon \), then we say \( \alpha \) coheres with \( \beta \).

Coherent links do not cut one another. If they did, their projections to the sheet would have nonzero winding; and since winding is invariant under link homotopy, there would be two wires in a design with nonzero winding. But I claim that if \( \omega \) and \( \upsilon \) are wires in a design, then \( \text{wind}(\omega, \upsilon) = 0 \). For if \( \omega \neq \upsilon \), then \( \omega \) and \( \upsilon \) do not intersect, and hence their lifts cannot intersect. Or if \( \omega = \upsilon \), then since this link is
simple, its lifts are all disjoint. In neither case can a lift of $\omega$ cut a lift of $v$, and thus $\text{wind}(\omega, v) = 0$.

Next we show that coherent links cannot have both terminals in common without being equal. This is a corollary of a result about link homotopy in blankets that is useful in its own right.

**Lemma 4c.3.** If $\eta$ is a link with two terminals, then no two distinct lifts of $\eta$ are link-homotopic.

**Proof.** Let $\alpha$ and $\beta$ be link-homotopic lifts of $\eta$. We prove $\alpha = \beta$. By Corollary 3a.5, the points $\alpha(0)$ and $\beta(0)$ lie on the same fringe $X$, while $\alpha(1)$ and $\beta(1)$ lie on a different fringe $Y$. These fringes are different because $\eta$ has two terminals. Let $\sigma$ be a simple path in $X$ from $\beta(0)$ to $\alpha(0)$, and let $\tau$ be a simple path in $Y$ from $\alpha(1)$ to $\beta(1)$. The loop $\lambda = \alpha \ast \tau \ast \beta \ast \sigma$ is inessential because the blanket $M$ is simply connected.

![The loop $\lambda$.](image)

![The loop $\mu$.](image)

**Figure 4c-2.** Link-homotopic lifts of a wire are identical. For the fact that $\alpha$ and $\beta$ are link-homotopic implies that the loop $\lambda$ shown here is inessential, and consequently its projection $\mu$ is inessential. It would not be inessential if it wrapped around either terminal of $\omega$, and so $\sigma$ and $\tau$ are actually constant paths.

Now we project $\lambda$ to the sheet $S$. Because $\lambda$ is inessential in $M$, the resulting loop $\mu$ is inessential in $S$. Therefore $\mu$ is also inessential in the larger sheet $S' = S \cup \text{inside}(p(Y))$. Now

$$\mu = \eta \ast (p \circ \tau) \ast \tilde{\eta} \ast (p \circ \sigma).$$

In $S'$ the path $p \circ \tau$ is inessential, and hence $\mu$ is path-homotopic to $p \circ \sigma$ in $S'$. So $p \circ \sigma$ is inessential in $S'$. But $p \circ \sigma$ lies in $p(X)$, which by Lemma 3b.1 is a retract of $S'$. So $p \circ \sigma$ is inessential in $p(X)$, and hence in $S$. Therefore the endpoints of its lift $\sigma$ are equal, which means $\alpha(0) = \beta(0)$. Hence $\alpha = \beta$ by uniqueness of liftings (Theorem 2b.2).

**Corollary 4c.4.** If $\alpha$ and $\beta$ are unequal coherent links, then $\beta$ has a terminal that is not a terminal of $\alpha$. 
Proof. Let \( p \) denote the covering map. If \( p \circ \alpha \neq p \circ \beta \), then these links are link-homotopic to distinct wires in a design. In this case all four terminals belonging to \( p \circ \alpha \) and \( p \circ \beta \) are different, and the same goes for \( \alpha \) and \( \beta \). On the other hand, if \( p \circ \alpha = p \circ \beta \), then this link is homotopic to a wire, and hence has two terminals. Now Lemma 4c.3 shows that \( \alpha \) and \( \beta \) are not link-homotopic. We cannot have \( \alpha \simeq_L \beta \) either. Thus \( \alpha \) and \( \beta \) have at least three terminals among them. \( \square \)

The following lemma derives a further fact about coherent links. It will be needed in Chapter 5.

**Lemma 4c.5.** If \( \alpha \) and \( \beta \) are unequal coherent links, then the endpoints of \( \beta \) lie on the same side of \( \alpha \).

**Proof.** Let \( p \) denote the covering map. If \( p \circ \alpha \neq p \circ \beta \), then the terminals of \( p \circ \alpha \) differ from those of \( p \circ \beta \), and since coherent links do not cut one another, the endpoints of \( \beta \) must lie on the same side of \( \alpha \). We may therefore assume that \( \alpha \) and \( \beta \) are liftings of the same path \( \eta \), which is link-homotopic to a wire \( \omega \). If \( \beta \) does not share any terminals with \( \alpha \), we are done, because \( \beta \) does not cut \( \alpha \). So assume that \( \alpha(0) \) and \( \beta(0) \) lie on the fringe \( X \). Then \( \alpha(1) \) and \( \beta(1) \) lie on different fringes, by Corollary 4c.4.

Because \( \eta \) is link-homotopic to a wire \( \omega \), Proposition 3a.6 provides lifts \( \alpha' \) and \( \beta' \) of \( \omega \) that are link-homotopic to \( \alpha \) and \( \beta \), respectively. Since \( \alpha' \) and \( \beta' \) are distinct lifts of a simple path, they do not intersect, and hence the endpoints of \( \beta' \) lie on the same side, say the left, of \( \alpha' \). Let \( F: \alpha \simeq_L \alpha' \) and \( G: \beta \simeq_L \beta' \) be lifts of a link homotopy between \( \eta \) and \( \omega \), as in Lemma 3a.6. For every \( t \in I \), the point \( F(0, t) \) separates the fringe \( X \) into two components, a left component and a right component. We show that for every \( t \), the point \( G(0, t) \) lies to the left of \( F(0, t) \). This is true at \( t = 1 \), because \( G(0, 1) = \beta'(1) \) and \( F(0, 1) = \alpha'(1) \). For no \( t \) are \( F(0, t) \) and \( G(0, t) \) equal, else \( F = G \) by uniqueness of liftings and thus \( \alpha = \beta \). Hence by continuity of \( F \) and \( G \), the point \( \beta(0) = G(0, 0) \) lies to the left of \( \alpha(0) = F(0, 0) \). Also the fringe containing \( \beta(1) \) lies in \( \text{left}(\alpha) \) because it lies in \( \text{left}(\alpha') \). (Here we are using Lemma 3c.4.) Hence both endpoints of \( \beta \) lie left of \( \alpha \). \( \square \)

A generalization of simplicity

Though the design routability theorem refers only to straight cuts, the results that lead to it use many cuts that are not straight, nor even simple. Often to make a proof go through one need not assume that a cut is simple, but rather that the cut respects the design in question. This relation is discussed in detail in Section 4E. For now we define a weaker relation, called weak respect, that is useful in obtaining upper bounds on flow.
**Definition 4c.6.** Let $\Omega$ be a design on a sheet $S$. A cut $\chi$ of $S$ weakly respects the design $\Omega$ if whenever

(a) $\omega$ is a wire in $\Omega$,
(b) $\tilde{\omega}$ and $\tilde{\omega}'$ are two lifts of $\omega$ that share a terminal, and
(c) $\tilde{\chi}$ is a lift of $\chi$ that cuts $\tilde{\omega}$,

the terminals of $\tilde{\omega}'$ lie within the same side of $\tilde{\chi}$.

For a cut to respect a design weakly means, in essence, that each fringe of the blanket contributes at most one necessary crossing to the flow across the cut. As one can check, this relation is invariant under link homotopy. In other words, if $\chi$ weakly respects a design $\Omega$, then every cut $\chi'$ in $[\chi]_L$ weakly respects every embedding of $\Omega$. Figure 4c-1 illustrates weak respect; Figure 4c-3 illustrates its absence.

![Diagram](image)

**Figure 4c-3. Lack of weak respect.** The cut $\chi$ (at left) does not weakly respect the wire $\omega$ (or rather, the one-element design $\{\omega\}$), because two lifts of $\omega$ that share a terminal cut the same lift of $\chi$ (at right).

If two lifts of a wire share a fringe, then some path from one to another wraps around the fringe one or more times. Figure 4c-3 shows a cut that does not weakly respect its design; it wraps around the terminal of a wire. This figure suggests that simple cuts have weak respect for all designs, a fact which we now prove.

**Lemma 4c.7.** Simple cuts respect all designs weakly.

**Proof.** Let $M$ be a blanket on a sheet $S$, with covering map $p: M \to S$. Let $\chi$ be a simple cut of $S$, and let $\alpha$ be any lift of $\chi$. Suppose that $\tilde{\omega}$ and $\tilde{\omega}'$ share a terminal and lift the same wire in $S$. We assume that $\alpha$ cuts $\tilde{\omega}$, and show that the terminals of $\tilde{\omega}'$ lie on the same side of $\alpha$. For some $i, j \in \{0, 1\}$, the endpoints $\tilde{\omega}(i)$ and $\tilde{\omega}'(j)$ lie on the same fringe $F$. Because $p \circ \tilde{\omega} = p \circ \tilde{\omega}'$ is a wire, its terminals are distinct, and hence $i = j$; we may assume $i = j = 0$. Let $\alpha_1, \ldots, \alpha_n$ be the lifts of $\chi$ that cut $\tilde{\omega}$. (We have $n > 0$ because $\alpha$ cuts $\tilde{\omega}$.) Being distinct liftings of the simple path $\chi$, the links $\{\alpha_i\}$ cannot intersect. Assume that $\alpha_1$ is chosen so that $\alpha_2, \ldots, \alpha_n$ lie in the scrap of $M - \text{Im} \alpha_1$ that does not contain $F$. Let $A$ denote the scrap of $M - \text{Im} \alpha_1$ that contains $F$. 

- 126 --
Figure 4c-4. Why a simple cut respects all designs weakly. The paths $\alpha_1, \ldots, \alpha_n$ are the lifts of a simple cut $\chi$ that cut the lift $\tilde{\omega}$ of the wire $\omega$. Here $\tilde{\omega}'$ is another lift of $\omega$ that shares terminals with $\tilde{\omega}$ and $\alpha_1$. The link $\tilde{\chi}$ is a lift of $\chi$ that is to $\tilde{\omega}'$ as $\alpha_1$ is to $\omega$. Where can it go? It must either cut $\tilde{\omega}$ or cross $\alpha_1$, but it can do neither.

Because $\tilde{\omega}$ and $\tilde{\omega}'$ lift the same link $\omega$, there is a covering transformation $h: M \to M$ such that $h \circ \tilde{\omega} = \tilde{\omega}'$. If $h$ had a fixed point $x$, then $h$ and $id_M$ would be lifts of $p$ agreeing at $x$, and Theorem 2b.2 would imply $h = id_M$. But $h$ is not the identity transformation, because $\tilde{\omega} \neq \tilde{\omega}'$, so it has no fixed points, and hence $\alpha_1 \neq h \circ \alpha_1$. Because $T$ is a homeomorphism, the lifts of $\chi$ that cut $\tilde{\omega}'$ are precisely $h \circ \alpha_1, \ldots, h \circ \alpha_n$. Also for this reason, and because $p \circ h = p$, the link $h \circ \alpha_1$ lies to the left of $h \circ \alpha_j$ whenever $i < j$. Note also that $h(F)$ is $F$.

We show that both terminals of $\tilde{\omega}'$ lie in $A$. Suppose not. Then either $\alpha_1$ cuts $\tilde{\omega}'$, or else the two links share a terminal. In either case, $h \circ \alpha_1$ lies in $A$. (See Figure 4c-4.) For if $\alpha_1$ cuts $\tilde{\omega}'$, then $\alpha_1 = h \circ \alpha_k$ for some $k > 1$, and $h \circ \alpha_1$ lies on the side of $h \circ \alpha_k$ that contains $F$, namely $A$. If instead $\alpha_1$ shares a terminal with $\tilde{\omega}'$, that terminal cannot be $F$, and again $h \circ \alpha_1$ lies in $A$. Hence in either case, $h \circ \alpha_1 \neq \alpha_m$ for any $m$, so $h \circ \alpha_1$ does not cut $\tilde{\omega}$. Let $\eta \in [\tilde{\omega}]_L$ be simple and not intersect $h \circ \alpha_1$. The terminal $F$ lies in $A$, so by assumption, the other terminal of $\tilde{\omega}'$ must have points outside $A$. By modifying the portion of $\eta$ lying in $M - A$, we can obtain a simple link $\eta' \in [\tilde{\omega}]_L$ that does not intersect $h \circ \alpha_1$ either. Then $h \circ \alpha_1$ does not cut $\eta'$, and hence does not cut $\tilde{\omega}'$. But we know that $h \circ \alpha_1$ does cut $\tilde{\omega}'$. This contradiction shows that the terminals of $\tilde{\omega}'$ must lie in $A$.

The rest is easy. Since $\alpha = \alpha_k$ for some $k$, neither terminal of $\alpha$ lies in $A$, while both terminals of $\tilde{\omega}'$ lie in $A$. Hence $\alpha$ cannot cut $\tilde{\omega}'$, and the two links cannot even share a terminal. Therefore the terminals of $\tilde{\omega}'$ lie on the same side of $\alpha$. □
4D. Properties of Flow

The flow across a straight cut measures the total width of the wiring that must pass between two fringes. Another quantity that needs analysis is the amount of wiring that must pass between a fringe and a wire. To define it we introduce the concept of a half-cut, a half-link that begins on a fringe and ends on a wire.

This section defines half-cuts and explores their properties. Fortunately, we can study the attributes of half-cuts without introducing a lot of new concepts. Instead we define properties of half-cuts in terms of the properties of their associated cuts. In particular, the flow, degeneracy, triviality, and weak respect of a half-cut are defined in terms of the cut properties of the same names. (Much of the complexity of this theory, but also much of the interest, arises because the associated cuts of a half-cut are not, in general, simple, or even link-homotopic to anything simple.)

In particular, we can relate the flows across half-cuts by analyzing flows across cuts. In this section we start to examine the methods for relating three or more cuts simultaneously. One important result is Proposition 4d.2. Whenever a wire (or a route of a wire) makes a necessary crossing with a cut, the half-cuts of the cut ending at that crossing have flows whose sum, when added to the width of the wire, equals or exceeds the flow across the cut.

Definition of a half-cut

Half-cuts arise as follows. Suppose \( \omega \) routes a wire in a design \( \Omega \). Formally, a half-cut for \( \omega \) at \( t \) is a half-link \( \sigma \) whose liftings are simple and which satisfies \( \sigma(1) = \omega(t) \). Thus \( \sigma(0) \) lies on a fringe and \( \sigma(1) = \omega(t) \). For example, if \( (s, t) \) is a crossing of a cut \( \chi \) by a link \( \omega \), then the half-links \( \chi_{0,s} \) and \( \chi_{1,s} \) are both half-cuts for \( \omega \) at \( t \). When the crossing \( (s, t) \) is clear from context we often omit mention of \( \omega \) and \( t \), and refer to \( \chi_{0,s} \) and \( \chi_{1,s} \) simply as half-cuts.

Attributes of half-cuts

Like cuts, half-cuts have flow and capacity in the context of a design. Let \( \sigma \) be a half-cut for \( \omega \) at \( t \), and suppose \( \sigma(0) \) lies on the fringe \( F \). The capacity of \( \sigma \) is defined to be

\[
\text{cap}(\sigma, \omega) = \|\sigma\| - \text{width}(F)/2 - \text{width}(\omega)/2.
\]

Since \( \omega \) is a route of a wire in a particular design, the widths of \( \omega \) and \( F \) are taken from this design. We sometimes abbreviate \( \text{cap}(\sigma, \omega) \) to \( \text{cap}(\sigma) \), even though \( \sigma \) alone does not specify which link \( \omega \) is involved.

The flow across \( \sigma \) depends even more strongly on \( \omega \) and on the crossing \( (1, t) \) of \( \sigma \) by \( \omega \). If \( \Omega \) is a design, we define \( \text{flow}(\sigma, \Omega) \) to be \( \text{flow}(\sigma \ast \omega_{t,1}, \Omega) \), which by
Section 4D

Properties of Flow

Figure 4d-1. The flow across a half-cut. The half-link $\sigma$ is a half-cut for the wire $\omega$ at $t$. If the wires $\omega$ this design have width 1, then the flow across $\sigma$, defined as the flow across the link $\sigma \ast \omega_{t:1}$, is 2.

definition is $\text{flow}(\gamma, \Omega)$ where $\gamma$ is any cut in $[\sigma \ast \omega_{t:1}]_\rho$. Thus the flow across a half-cut is defined in terms of the flow across a cut.

This definition of flow makes intuitive sense, for if $\omega$ is a route of a wire in $\Omega$, no wire in $\Omega$ can make a necessary crossing with $\omega$. Hence the necessary crossings of $\sigma \ast \omega_{t:1}$ must somehow reflect necessary crossings of $\sigma$. From a technical point of view the definition makes less sense, for two reasons. First, it can happen that no link in $[\sigma \ast \omega_{t:1}]_L$ is simple, and thus we are forced to consider the flow across nonsimple cuts. Second, the choice of $\sigma \ast \omega_{t:1}$ rather than $\sigma \ast \omega_{t:0}$ is arbitrary, and yet significant: these two links can have different flows, even in a design consisting of $\omega$ alone.

Both technical difficulties can be overcome by extending the notion of weak respect (Definition 4c.6) to half-cuts. We do so by referring again to cuts. If $\sigma$ is a half-cut for $\omega$ at $t$, then the cuts in the sets $[\sigma \ast \omega_{t:0}]_L$ and $[\sigma \ast \omega_{t:1}]_L$ are called associated to $\sigma$. The half-cut $\sigma$ weakly respects a design $\Omega$ if every cut associated to $\sigma$ weakly respects $\Omega$. Since weak respect is invariant under link homotopy, this condition is not as restrictive as it sounds. Lemma 4d.3 below shows that if $\sigma$ respects $\Omega$ weakly, then all associated cuts of $\sigma$ have the same flow in $\Omega$.

Associated cuts help us define other properties of half-cuts as well. For instance, we call a half-cut $\sigma$ degenerate if it has a degenerate associated cut. Similarly, a half-cut is trivial if it has a trivial associated cut. Triviality can be cast in terms of liftings. Let $\sigma$ be a half-cut for $\omega$ at $t$, and let $\tilde{\sigma}$ and $\tilde{\omega}$ be lifts of $\sigma$ and $\omega$ that reflect the crossing $(1,t)$. In other words, $\tilde{\sigma}(1) = \tilde{\omega}(t)$. Then $\sigma$ is trivial if and only if the terminal of $\tilde{\sigma}$ is a terminal of $\tilde{\omega}$.

Equivalence of half-cuts

If $\sigma$ and $\tau$ are homotopic as half-links, meaning that $\sigma(1) = \tau(1)$ and $\sigma \ast \hat{\tau}$ is trivial, then $\sigma$ is a half-cut for $\omega$ at $t$ if and only if $\tau$ is. We also have $\sigma \ast \omega_{t:1} \simeq_L \tau \ast \omega_{t:1}$, and hence (by Proposition 4b.3) homotopic half-cuts have equal flow. There is, however, a much coarser equivalence relation on half-cuts that preserves flow.

**Definition 4d.1.** Let $\sigma$ and $\tau$ be half-cuts for $\omega$ at $s$ and $v$ at $t$, respectively, where $\omega \simeq_L v$. Suppose $\tilde{\omega}$ and $\tilde{\sigma}$ are lifts of $\omega$ and $\sigma$ such that $\tilde{\sigma}(1) = \tilde{\omega}(s)$. Also
suppose \( \tilde{v} \) and \( \tilde{r} \) are lifts of \( v \) and \( r \) such that \( \tilde{r}(1) = \tilde{v}(t) \). We say that \( \sigma \) and \( \tau \) are akin if the lifts may be chosen so that \( \tilde{\omega} \simeq_L \tilde{v} \) and \( \tilde{\sigma} \) and \( \tilde{\tau} \) have the same terminal.

If \( \sigma \) and \( \tau \) are akin, then Corollary 3a.5 implies that \( \tilde{\sigma} \star \tilde{\omega}_{1;1} \) and \( \tilde{\tau} \star \tilde{\omega}_{1;1} \) are link-homotopic, and hence their projections to the sheet are link-homotopic also. It follows from Proposition 4b.3 that \( \sigma \) and \( \tau \) have the same flow in any design.

In fact, all the properties we define for half-cuts, except geometric quantities like capacity, are invariant under kinship—the relation of being akin. The reason is that half-cuts that are akin have link-homotopic associated cuts. Hence half-cuts that are akin are equally degenerate, and they have weak respect for the same designs.

**Mid-cuts**

Just as there are cuts from fringes to fringes and half-cuts from fringes to wires, there are mid-cuts from wires to wires. We shall occasionally have use for them.

Suppose \( v \) and \( \omega \) are routes of wires in a design \( \Omega \) on the sheet \( S \). For \( s, t \in (0, 1) \), a mid-cut between \( v \) at \( s \) and \( \omega \) at \( t \) is a mid-link \( r \) in \( S \) whose liftings are simple and which satisfies \( r(0) = v(s) \) and \( r(1) = \omega(T) \). We define the properties of mid-cuts by analogy with half-cuts. The capacity of the mid-cut \( r \) is

\[
cap(r) = \|r\| - \text{width}(v)/2 - \text{width}(\omega)/2,
\]

and its associated cuts are the cuts in the sets \([v_{i;1} \star r \star \omega_{t;1}]_L\), for \( i, j \in \{0, 1\} \). A mid-cut respects a design weakly if all its associated cuts do, and it is degenerate if its associated cuts are. We define the flow across the mid-cut \( r \) to be the flow across the link \( v_{1;1} \star r \star \omega_{t;1} \). I leave it to the reader to adapt the definition of kinship (4d.1) to mid-cuts.

Together with cuts, half-cuts and mid-cuts are collectively known as **subcuts**. All liftings of subcuts are simple sublinks.

**Combining two half-cuts**

The main impact of the study of blankets is that it allows us to relate the flows of different cuts. These relationships are the theme of the rest of the chapter. Our
first result relates the flow across a cut $\chi$ to the sum of the flows across half-cuts $\chi_{0,s}$ and $\chi_{1,t}$ for a link $\omega$. It says that if the half-cuts lie on "opposite sides" of $\omega$, then the flows across these half-cuts, and $\omega$ itself, all contribute to the flow across $\chi$. This result was first claimed (in a different model) by Cole and Siegel, who used it in [6] without proof. Our knowledge of the topology of blankets allows us to give a rigorous proof of a more general claim.

**Proposition 4d.2.** Let $\chi$ be a cut of a sheet $S$, and let $\Omega$ be a design on $S$. Suppose that $(s, t)$ is a necessary crossing of $\chi$ by a route $\rho$ of a wire $\omega$ in $\Omega$. Then

$$\text{flow}(\chi, \Omega) \geq \text{flow}(\chi_{0,s}, \Omega) + \text{flow}(\chi_{1,t}, \Omega) + \text{width}(\omega),$$

with equality if $\chi$ respects $\omega$ weakly.

**Proof.** Let $M$ be a blanket for $S$, with covering map $p: M \to S$. Because the crossing $(s, t)$ is necessary, there are lifts $\tilde{\chi}$ of $\chi$ and $\tilde{\rho}$ of $\rho$ such that $\tilde{\chi}(s) = \tilde{\rho}(t)$, and $\tilde{\chi}$ cuts $\tilde{\rho}$. Let $X$ and $Y$ be the fringes of $M$ containing $\tilde{\chi}(0)$ and $\tilde{\chi}(1)$, respectively, and let $Z$ be the fringe of $M$ containing $\tilde{\rho}(1)$. Assume without loss of generality that $Z \subset \text{right}(\tilde{\chi})$. Let $\tilde{\alpha}$ be a simple link in $\text{right}(\tilde{\chi})$ from $X$ to $Z$. Then by Proposition 3c.2, we have $Im \chi \in \text{left}(\tilde{\alpha})$. Let $\tilde{\beta}$ be a simple link in $\text{right}(\tilde{\chi}) \cap \text{left}(\tilde{\alpha})$ from $Y$ to $Z$. By Corollary 3a.5, we have the relations

$$\tilde{\alpha} \preceq_L \tilde{\chi}_{0,s} \ast \tilde{\rho}_{t:1} \quad \text{and} \quad \tilde{\beta} \preceq_L \tilde{\chi}_{1,s} \ast \tilde{\rho}_{t:1}.$$

Write $\alpha = p \circ \tilde{\alpha}$ and $\beta = p \circ \tilde{\beta}$. Projecting to the sheet, we have $\alpha \preceq_L \chi_{0,s} \ast \rho_{t:1}$ and $\beta \preceq_L \chi_{1,s} \ast \rho_{t:1}$. Hence by Proposition 4b.3 and the definition of the flow across a half-cut, we have $\text{flow}(\chi_{0,s}, \Omega) = \text{flow}(\alpha, \Omega)$ and $\text{flow}(\chi_{1,s}, \Omega) = \text{flow}(\beta, \Omega)$. Hence it suffices to prove

$$\text{flow}(\chi, \Omega) \geq \text{flow}(\alpha, \Omega) + \text{flow}(\beta, \Omega) + \text{width}(\omega), \quad (4-1)$$

with the reverse inequality

$$\text{flow}(\chi, \Omega) \leq \text{flow}(\alpha, \Omega) + \text{flow}(\beta, \Omega) + \text{width}(\omega) \quad (4-2)$$

holding also if $\chi$ respects $\Omega$ weakly.

Bounds on the flow across $\chi$ come from comparing the links that cross $\tilde{\chi}$ to those that cross $\tilde{\alpha}$ and $\tilde{\beta}$. Every lift $\tilde{\nu}$ of $\nu$ that cuts $\tilde{\chi}$ contributes exactly $\text{width}(\nu)$ to $\text{flow}(\chi, \Omega)$, and similar statements hold for $\text{flow}(\alpha, \Omega)$ and $\text{flow}(\beta, \Omega)$. So suppose that $\tilde{\nu}$ lifts a wire $\nu \in \Omega$. We show that if $\tilde{\nu}$ cuts $\tilde{\alpha}$ or $\tilde{\beta}$, then $\tilde{\nu}$ cuts $\tilde{\chi}$. By Proposition 3a.6, there is a lift $\tilde{\omega}$ of $\omega$ in $[\tilde{\rho}]_L$; its terminals are those of $\tilde{\rho}$, and hence $\tilde{\chi}$ cuts $\tilde{\omega}$. Since $\tilde{\alpha}$ runs from a terminal of $\tilde{\chi}$ to one of $\tilde{\omega}$, Lemma 4c.1 shows that if $\tilde{\nu}$ cuts $\tilde{\alpha}$, it must also cut either $\tilde{\chi}$ or $\tilde{\omega}$. Similarly, if $\tilde{\nu}$ cuts $\tilde{\beta}$, it must also cut either $\tilde{\chi}$ or $\tilde{\omega}$. But the links $\tilde{\nu}$ and $\tilde{\omega}$ cohere, and therefore they do not cut one another.
Figure 4d-3. Combining half-cuts to form a cut. At left, the straight half-cuts \( \chi_{0t} \) and \( \chi_{1t} \) for \( \omega \) at \( t \) connect to form the straight cut \( \chi \). Because the crossing \((s,t)\) of \( \chi \) by \( \omega \) is necessary, the lift \( \tilde{\chi} \) cuts the lift \( \tilde{\omega} \), at right. Every wire lifting that cuts \( \tilde{\alpha} \) or \( \tilde{\beta} \) also cuts \( \tilde{\chi} \) because it cannot cut \( \tilde{\omega} \). Conversely, every wire lifting that cuts \( \tilde{\chi} \) also cuts \( \tilde{\alpha} \) or \( \tilde{\beta} \) unless it shares the terminal \( Z \) with \( \tilde{\omega} \).

Hence every lift \( \tilde{v} \) that contributes to the flow across \( \alpha \) or \( \beta \) contributes the same amount to the flow across \( \chi \). This observation alone implies that \( \text{flow}(\chi, \Omega) \) is no less than \( \text{flow}(\alpha, \Omega) + \text{flow}(\beta, \Omega) \). The term \( \text{width}(\omega) \) in inequality (4-1) is accounted for by the lift \( \tilde{\omega} \) of \( \omega \). It cuts \( \tilde{\chi} \), but does not cut \( \tilde{\alpha} \) or \( \tilde{\beta} \), since it shares the terminal \( Z \) with the latter links. So \( \tilde{\omega} \) contributes an extra amount \( \text{width}(\omega) \) to \( \text{flow}(\chi, \Omega) \). Thus inequality (4-1) is established.

Now we suppose that \( \chi \) weakly respects \( \omega \), and prove inequality (4-2). The threads \( \text{Im} \tilde{\alpha} \), \( \text{Im} \tilde{\beta} \), and \( \text{Im} \tilde{\chi} \) form a web of 3 threads. By Lemma 3b.7 and Proposition 3b.8, they separate \( M \) into 4 scraps. Three of these, call them \( A \), \( B \), and \( C \), border on the threads \( \text{Im} \tilde{\alpha} \), \( \text{Im} \tilde{\beta} \), and \( \text{Im} \tilde{\chi} \), respectively; the fourth scrap, call it \( N \), borders on all three threads, and it contains no fringes. Let \( \tilde{v} \) lift any wire \( v \in \Omega \). If \( \tilde{v} \) cuts \( \tilde{\chi} \), it must have one terminal in \( C \), and its other terminal, if not \( Z \), must lie in \( A \), \( B \), or \( N \). It cannot lie in \( N \), because \( N \) contains no fringes. And if it lies in \( A \) or \( B \), then \( \tilde{v} \) cuts \( \tilde{\alpha} \) or \( \tilde{\beta} \), respectively. Hence the only way that \( \tilde{v} \) can contribute to the flow across \( \chi \) without contributing to the flow across \( \alpha \) or \( \beta \) is if \( \tilde{v} \) has \( Z \) as a terminal and cuts \( \tilde{\chi} \). And since different wires in a design have different terminals, this implies \( v = \omega \). Because \( \chi \) weakly respects \( \omega \), Definition 4c.6 says that no lift of \( \omega \) other than \( \tilde{\omega} \) can cut \( \tilde{\chi} \) and have \( Z \) for a terminal. Inequality (4-2) follows.

Respect and half-cuts

As previously mentioned, the definition of flow for half-cuts is somewhat arbitrary; the associated cuts of a half-cut can have quite different properties. In particular, some can respect a design without the others doing so, and they can
have different flows, even considering only the half-cut's wire. These facts are illustrated by Figure 4d-4. But if a half-cut respects its wire weakly, this problem goes away.

**Figure 4d-4. Lack of weak respect in a half-cut.** Even a simple half-cut can easily lack weak respect for its wire, as this example shows. At left, $\sigma$ is a half-cut for $\omega$ at $t$. The associated cut $\sigma \ast \omega_{t:0}$ is simple and therefore respects $\omega$. But $\sigma \ast \omega_{t:1}$ does not respect $\omega$. The picture at right shows two lifts of $\omega$ that share the terminal $X$. One cuts a lift of $\sigma \ast \omega_{t:1}$, and the other shares a terminal with it.

**Lemma 4d.3.** Let $\eta$ route a wire $\omega$ in the design $\Omega$, and let $\sigma$ be a half-cut for $\eta$. If $\sigma$ respects $\omega$ weakly, then all cuts associated to $\sigma$ have the same flow in $\Omega$.

**Proof.** Let $\sigma$ be a half-cut for $\eta$ at $t$, and let $\alpha$ and $\beta$ be cuts associated to $\sigma$. If $\alpha$ and $\beta$ are link-homotopic, then they automatically have the same flow, so we may assume that $\alpha \simeq_L \sigma \ast \omega_{t:0}$ and $\beta \simeq_L \sigma \ast \omega_{t:1}$. We must prove $\text{flow}(\alpha, \Omega) = \text{flow}(\beta, \Omega)$. It suffices to prove $\text{wind}(\alpha, \omega) = \text{wind}(\beta, \omega)$ for all wires $\omega \in \Omega$.

**Figure 4d-5. Flow across the associated cuts of a half-cut.** The simple links $\tilde{\alpha}$ and $\tilde{\beta}$ lift non-homotopic cuts associated to a half-cut $\sigma$ for the link $\eta$, whose lift $\tilde{\eta}$ is link-homotopic to a wire lifting $\tilde{\omega}$. No wire lifting cuts $\tilde{\omega}$, so every wire lifting that cuts $\tilde{\alpha}$ or $\tilde{\beta}$ also cuts the other, unless it shares a terminal with $\tilde{\omega}$. The latter option is ruled out if $\sigma$ weakly respects the design.

We imitate the proof of Proposition 4d.2. Let $S$ be the sheet of $\Omega$, let $M$ be its blanket, and let $p: M \to S$ be the covering map. Lift $\sigma$ and $\eta$ to $\tilde{\sigma}$ and $\tilde{\eta}$ such
that $\tilde{\sigma}(1) = \tilde{\eta}(t)$. Find simple links $\tilde{\alpha}$ and $\tilde{\beta}$ in $M$ from the fringe containing $\tilde{\sigma}(0)$ to the fringes containing $\tilde{\omega}(0)$ and $\tilde{\omega}(1)$, respectively, such that $\text{Im} \tilde{\alpha} \cup \text{Im} \tilde{\beta} \cup \text{Im} \tilde{\eta}$ is a web of 3 threads. Then $p \circ \tilde{\alpha} \simeq_L \alpha$ and $p \circ \tilde{\beta} \simeq_L \beta$. Let $v \neq \omega$ be a wire of $\Omega$. There are exactly $\text{wind}(\alpha, v)$ lifts of $v$ that are cut by $\tilde{\alpha}$; because $v \neq \omega$, none of these can share a terminal with $\tilde{\eta}$. Also because they cohere with $\tilde{\eta}$, none are cut by $\tilde{\eta}$. We conclude that all such lifts are cut by $\tilde{\beta}$. Hence $\text{wind}(\alpha, v) \leq \text{wind}(\beta, v)$. By symmetry, the opposite inequality holds as well.

To establish a similar formula with $\omega$ in place of $v$, it suffices to show that no lift $\tilde{\omega}'$ of $\omega$ that shares a terminal with $\tilde{\eta}$ is cut by $\tilde{\alpha}$. Let $\tilde{\omega} \in [\tilde{\eta}]_L$ lift $\omega$; then $\tilde{\omega}$ and $\tilde{\omega}'$ share a terminal. Because $\sigma$ respects $\omega$ weakly, its associated cut $p \circ \tilde{\alpha}$ also respects $\omega$ weakly. Hence $\tilde{\alpha}$ cutting $\tilde{\omega}'$ would imply that $\tilde{\omega}$ could not share a terminal with $\tilde{\alpha}$. But $\tilde{\eta}$ does, and $\tilde{\eta} \simeq_L \tilde{\omega}$. We conclude that $\tilde{\alpha}$ cannot cut $\tilde{\omega}$, and hence $\tilde{\alpha}$ cannot cut $\tilde{\eta}$. Thus $\text{wind}(\alpha, v) \leq \text{wind}(\beta, v)$ even when $v = \omega$. By symmetry, this inequality is actually an equality. □

Lemma 4d.3 gives evidence that weak respect of a design is a good condition to require of a half-cut.

More bounds on flow

The technique used to prove Proposition 4d.2 and Lemma 4d.3 is a very powerful one. It compares flows by building a loop of links in a blanket and drawing correspondences among the wire liftings that cut those links. The following lemma encapsulates the technique for future use.

**Lemma 4d.4.** Let $\gamma$ be a simple link in a blanket $M$, and suppose $\alpha \simeq_P \gamma$. Every simple link in $M$ that cuts $\gamma$ either cuts some link within $\alpha$, or contacts a fringe of $M$ that intersects $\alpha$ but not $\gamma$.

**Proof.** We inductively apply the technique of Proposition 4d.2, constructing webs in the blanket $M$. Let $\alpha_1$, $\ldots$, $\alpha_n$ be the links contained in $\alpha$. For $1 \leq i \leq n$, let $F_{i-1}$ be the fringe of $M$ containing $\alpha_i(0)$, and let $F_i$ be the fringe containing $\alpha_i(1)$. Then the terminals of $\tilde{\chi}$ are $F_0$ and $F_n$. The lemma says that every simple link in $M$ that cuts $\gamma$ either cuts some link $\alpha_i$ or else has some fringe $F_i$ as a terminal where $1 \leq i < n$. To prove this claim, we construct a sequence of simple links $\kappa_0$, $\ldots$, $\kappa_{n-1}$ where $\kappa_m$ has terminals $F_m$ and $F_n$. At the same time, we prove by induction on $m$ that every link that cuts $\gamma$ either cuts $\kappa_m$, cuts one of the links $\alpha_1$, $\ldots$, $\alpha_m$, or has one of the fringes $F_1$, $\ldots$, $F_m$ as a terminal. Because $\kappa_{n-1}$ is link-homotopic to $\alpha_n$, the case $m = n - 1$ will establish the claim. The basis case is easy: for $m = 0$ we let $\kappa_0$ be the simple link $\gamma$.

Now supposing the induction hypothesis is true for $m - 1$, we prove it for $m$. It is enough to show that a simple link cutting $\kappa_{m-1}$ either cuts $\alpha_m$ or $\kappa_m$, or else has
Section 4E

The Branches of a Blanket

Figure 4d-6. Flow across a chain of links. The paths \( \eta_i \) are the links of a chain for \( \tilde{\chi} \). By induction on \( m \), any simple link that cuts \( \gamma \) either cuts \( \kappa_m \), or cuts one of the links \( \alpha_1, \ldots, \alpha_m \), or has one of the fringes \( F_1, \ldots, F_m \) as a terminal.

\( F_m \) as a terminal. If the fringe \( F_m \) is a terminal of \( \kappa_{m-1} \), then \( \alpha_m \) is either trivial or link-homotopic to \( \kappa_{m-1} \), so we can simply set \( \kappa_m = \kappa_{m-1} \). Otherwise let \( A \) be the scrap of \( M - \text{Im} \, \kappa_{m-1} \) that contains \( F_m \), let \( \mu_m \in [\alpha_m]_L \) be a simple link in \( A \), and let \( \kappa_m \) be a simple link between \( F_m \) and \( F_n \) in the appropriate scrap of \( A - \text{Im} \, \mu_m \). Then the set

\[
\text{Im} \, \kappa_{m-1} \cup \text{Im} \, \mu_m \cup \text{Im} \, \kappa_m
\]

is a web of 3 threads. Because its inside contains no fringes (Proposition 3b.8), every link that cuts \( \kappa_{m-1} \) must either cut \( \mu_m \) or \( \kappa_m \), or else it must have \( F_m \) as a terminal. To cut \( \mu_m \) is to cut \( \alpha_m \). This step completes the induction, and thereby the proof. \( \square \)

4E. The Branches of a Blanket

To make further progress, we need more information on degenerate links and on the lifts of a wire that contribute to the flow across a cut. We obtain the latter by studying the relation of respect between cuts and designs, presented in Definition 4e.1 below, which is similar to but more powerful than the relation of weak respect defined in Section 4C (Definition 4c.6). We show that every simple cut respects every design, and show that when a simple cut makes a necessary crossing with a wire in a design, the resulting semisimple half-cuts respect that design. Most of the cuts and half-cuts at issue in a particular design turn out to be simple or semisimple, and hence respect the design. As a by-product of this study, we discover two important correlates of nondegeneracy. First, semisimple half-cuts are nondegenerate. Second, although simple cuts can be degenerate, the ones that are have zero flow.

Degeneracy and respect are closely related: both can be best understood in terms of a division of the fringes of a blanket into branches. A design partitions
the fringes of a blanket into branches just as it partitions the fringes of a sheet into articles. If $\Omega$ is a design on a sheet $S$, the articles of $\Omega$ are the components of the set $X = Bd S \cup \bigcup_{\omega \in \Omega} Im \omega$. Let $M$ be the blanket of $S$, with covering map $p: M \to S$. The branches of the design $\Omega$ are the components of the set $p^{-1}(X)$ in $M$. Two different fringes in $M$, say $A$ and $B$, are in the same branch if and only if for some wire $\omega$ in $\Omega$, there is a sequence of fringes $A = F_0$, $F_1$, $\ldots$, $F_n = B$ such that for $1 \leq i \leq n$, some lift of $\omega$ has terminals $F_{i-1}$ and $F_i$. We use the branches of a design to classify the lifts of a wire, and to identify degenerate cuts.

**Degeneracy**

A degenerate cut in a design $\Omega$ is one with a lifting whose endpoints fall in the same branch of $\Omega$. For if a cut $\sigma$ is degenerate in the design $\Omega$, then $\sigma$ has a chain $\tau$ in the set $X$. Hence by Proposition 2b.4, any lifting $\tilde{\sigma}$ of $\sigma$ has a path-homotopic lifting $\tilde{\tau}$ of $\tau$ which lies in $p^{-1}(X)$. Conversely, if $\tilde{\sigma}$ is any lift of $\sigma$ whose endpoints lie in the same branch, then there is path $\tilde{\tau}$ in $p^{-1}(X)$ between the endpoints of $\tilde{\sigma}$. (For some wire $\omega \in \Omega$, it is the concatenation of subpaths of lifts of $\omega$ with paths along fringes.) By Lemma 2a.5, there is a path homotopy $F$ between $\tilde{\sigma}$ and $\tilde{\tau}$, and the projection of $F$ is a path homotopy between $\sigma$ and a path $\tau: I \to X$.

This characterization of degeneracy clarifies several facts about degenerate cuts. First, a cut that is degenerate in one design is also degenerate in any embedding of that design. Second, if one associated cut of a subcut is degenerate, then all are. Third, the concatenation of degenerate subcuts is degenerate.

**Strong respect**

For a cut to respect a design strongly means, in essence, that each branch of the design contributes at most one necessary crossing to the flow across the cut.

**Definition 4e.1.** Let $\Omega$ be a design on a sheet $S$. A cut $\chi$ of $S$ respects $\Omega$ (strongly) if whenever

(a) $\omega$ is a wire in $\Omega$,
(b) $\tilde{\omega}$ and $\tilde{\omega}'$ are two lifts of $\omega$ in the same branch of $\Omega$, and
(c) $\tilde{\chi}$ is a lift of $\chi$ that cuts $\tilde{\omega}$,

the terminals of $\tilde{\omega}'$ lie within the same side of $\tilde{\chi}$.

The definition of strong respect (4e.1) differs from that of weak respect (4c.6) only in part (b), where the requirement that $\omega$ and $\omega'$ lie in the same branch of $\Omega$ has replaced the weaker condition that $\omega$ and $\omega'$ share a terminal. Like weak respect, strong respect is invariant under link homotopy.

The significance of Definition 4e.1 will become increasingly clear in later sections. From a technical point of view, it is central to the study of the design model. In the remainder of this section, we give sufficient conditions for a cut to respect a design.
Figure 4e-1. Weak respect but not strong respect. The pretzel-shaped cut at left respects the wire shown in grey weakly but not strongly. At right are the relevant liftings. The three wire liftings (grey) lie in the same branch of the design. Two of them cut the cut lifting (thin black path), but they do not share a terminal.

Figure 4e-1 shows a nearly minimal example of a cut that weakly respects a design without strongly respecting it. As the next result shows, in any such example the cut must have self-intersections.

Proposition 4e.2. Simple cuts respect all designs.

Proof. Let $\chi$ be a simple cut of a sheet $S$, and let $\omega$ be a wire in $S$. Suppose $\tilde{\chi}$ and $\tilde{\omega}$ are lifts of $\chi$ and $\omega$ that cut one another. For $n \geq 1$, we say $\tilde{\chi}$ $n$-respects $\omega$ if whenever $\alpha_0n$ are distinct lifts of $\omega$ starting with $\tilde{\omega}$ such that for $1 \leq i \leq n$, the links $\alpha_{i-1}$ and $\alpha_i$ share a terminal, the terminals of $\alpha_n$ lie on the same side of $\tilde{\chi}$. Then $1$-respect is the same as weak respect, and strong respect is $n$-respect for all $n$. We prove by induction on $n$ that $\chi$ $n$-respects $\omega$. The basis case is established by Lemma 4c.7; it says that $\chi$ respects $\Omega$ weakly because $\chi$ is simple.

Figure 4e-2. Why simple cuts respect all designs strongly. In the situation depicted, a simple cut $\chi$ with lifting $\tilde{\chi}$ does not $n$-respect a wire $\omega$. (Here $n = 3$.) A branch of $\omega$ includes liftings $\alpha_0$, ..., $\alpha_n$, of which $\alpha_0$ cuts $\tilde{\chi}$, and $\alpha_n$ cuts or shares a terminal with $\tilde{\chi}$. The link $\tilde{\chi}'$ is to $\alpha_1$ as $\tilde{\chi}$ is to $\alpha_0$. Where can its terminals lie? Wherever $\tilde{\chi}'$ goes it contradicts the induction hypothesis that $\chi$ $(n - 1)$-respects $\omega$.

For the induction step, where $n \geq 2$, we suppose that $\alpha_n$ either cuts or shares a terminal with $\tilde{\chi}$, and derive a contradiction. By the induction hypothesis, none of the links $\alpha_i$ for $i < n$ intersects $\tilde{\chi}$. Furthermore, the lifts of $\omega$ are all disjoint since
\( \omega \) is simple. Hence \( \tilde{\chi} \) and \( \alpha_1, \ldots, \alpha_n \) form a web of \( n - 1 \) threads (if \( \alpha_n \) cuts \( \tilde{\chi} \)) or \( n \) threads (if it does not), as shown in Figure 4e-2. Let \( \gamma \) be a simple link whose image is the thread of this web that intersects \( \text{Im} \tilde{\chi} \). Let \( h \) be a covering transformation that takes \( \alpha_0 \) to \( \alpha_1 \), and put \( \tilde{\chi}' = h \circ \tilde{\chi} \). Then \( \tilde{\chi}' \) cuts \( \alpha_1 \), because \( \tilde{\chi} \) cuts \( \alpha_0 \).

Consider how \( \tilde{\chi} \) leaves the inside of the web. One terminal of \( \tilde{\chi}' \) lies on the other side of \( \alpha_1 \) from \( \tilde{\chi} \). Regarding the position of the other terminal, there are three possibilities: it may be a terminal of \( \alpha_i \) for some \( i \neq 1 \), or \( \tilde{\chi}' \) may cut \( \alpha_i \) for some \( i \neq 1 \), or \( \tilde{\chi}' \) may cut \( \gamma \). Since \( \tilde{\chi}' \) does not cut \( \tilde{\chi} \), as these are lifts of a simple link, one or two applications of Lemma 4e.1 show that if \( \tilde{\chi}' \) cuts \( \gamma \), it also cuts \( \alpha_i \) where \( i \) is either \( 0 \) or \( n \). In each case, apply the induction hypothesis to the lift \( \tilde{\chi}' \) of \( \chi \), and the lifts \( \alpha_1 \) and \( \alpha_i \) of \( \omega \). Because \( \tilde{\chi} \) cuts \( \alpha_1 \), and \( \chi \) \( k \)-respects \( \omega \) where \( k = |i - 1| \), the link \( \alpha_i \) cannot cut \( \tilde{\chi}' \) or share a terminal with it. This contradiction means that \( \chi \) \( n \)-respects \( \omega \), completing the induction. \( \Box \)

One interesting consequence of Proposition 4e.2 is the following.

**Corollary 4e.3.** Degenerate simple cuts have zero flow.

**Proof.** More precisely, if \( \chi \) is a simple cut that is degenerate in a design \( \Omega \), then \( \text{flow}(\chi, \Omega) = 0 \). For if \( \chi \) is degenerate in \( \Omega \), then any lift \( \tilde{\chi} \) of \( \chi \) is path-homotopic to a path \( \beta \) in a single branch \( B \) of \( \Omega \). If the article \( C \) that is the projection of \( B \) is a single fringe, then so is \( B \), which makes \( \chi \) trivial. In this case no simple link cuts \( \tilde{\chi} \), and so \( \text{flow}(\chi, \Omega) = 0 \). So suppose \( C \) contains a wire \( \omega \in \Omega \). By Lemma 4d.4, no lift of a wire in \( \Omega \) can cut \( \tilde{\chi} \) unless it intersects a fringe of \( \beta \) or cuts a link in \( \beta \). In either case it intersects \( B \), and is therefore part of \( B \), and hence is a lift of \( \omega \). But the terminals of \( \tilde{\chi} \) lie in \( B \), and by Proposition 4e.2, \( \chi \) respects \( \omega \). Therefore no lift of \( \omega \) in \( B \) cuts \( \tilde{\chi} \). We conclude that no wire in \( \Omega \) has a lifting that cuts \( \tilde{\chi} \), and so \( \text{flow}(\chi, \Omega) = 0 \). \( \Box \)

**Half-cuts and mid-cuts**

As one might expect, we say that a half-cut or mid-cut respects a design if and only if its associated cuts do. Because respect is invariant under link homotopy, half-cuts and mid-cuts that are akin respect the same designs. Our next result gives us a tool for constructing respectful subcuts from respectful cuts. With Lemma 4e.4 and Proposition 4e.2 together, we have enough leverage to prove that almost any useful subcut respects its design.

**Lemma 4e.4.** Let \( \Omega \) be a design on the sheet \( S \), let \( \omega \) be a routing of a wire in \( \Omega \), and let \( \chi \) be a cut of \( S \) that respects \( \Omega \). If \((s, t)\) is a necessary crossing of \( \chi \) by \( \omega \), then for \( e \in \{0, 1\} \), every cut in \([\chi_{0,e} \ast \omega_{t,e}]]_L \) respects \( \Omega \).

**Proof.** Let \( M \) be the blanket of \( S \), with covering map \( p: M \to S \). Because respect is invariant under link homotopy, no generality is lost by assuming that \( \omega \in \Omega \). So
let $\tilde{\chi}$ and $\tilde{\omega}$ be lifts of $\chi$ and $\omega$ such that $\tilde{\chi}(s) = \tilde{\omega}(t)$, and let $\beta$ be any simple link in $[\tilde{\chi}_{0.s} \ast \tilde{\omega}_{t.e}]_L$. Also let $v$ be any wire in $\Omega$. It suffices to show that for any lift of $v$ that cuts $\beta$, every other lift of $v$ in the same branch has its terminals in the same scrap of $\beta$.

![Image of lift relation]

**Figure 4e.3. Construction of respectful subcuts.** The lift $\tilde{\chi}$ of $\chi$ cuts the lift $\tilde{\omega}$ of the link $\omega$, and $\chi$ respects the design. No wire lifting cuts $\tilde{\omega}$, so any wire lifting that cuts $\beta$ also cuts $\tilde{\chi}$. Of any two wire liftings in the same branch, at most one can cut $\beta$. We show that the other cannot even share a terminal with $\beta$.

I claim that if a lift $\tilde{v}$ of $v$ cuts $\beta$, then it also cuts $\tilde{\chi}$. The link $\tilde{\chi}$ cuts $\tilde{\omega}$ because the crossing $(s,t)$ is necessary. The link $\tilde{v}$ does not cut $\tilde{\omega}$, because their projections to $S$ form a subdesign of $\Omega$. In other words, $\tilde{v}$ coheres with $\tilde{\omega}$. Now apply Lemma 4c.1: because $\tilde{v}$ cuts $\beta$ but not $\tilde{\omega}$, it must cut $\tilde{\chi}$.

Let $\tilde{v}$ and $\tilde{v}'$ be two lifts of $v$ in the same branch of $\Omega$, and suppose $\tilde{v}$ cuts $\beta$. Then $\tilde{v}$ cuts $\tilde{\chi}$, and because $\chi$ respects $\Omega$, the other lift $\tilde{v}'$ neither cuts $\tilde{\chi}$ nor shares a terminal with it. In particular, $\tilde{v}'$ does not cut $\beta$, and it does not share a terminal with $\beta(0)$, because $\beta(0)$ lies on a terminal of $\tilde{\chi}$. One other possibility remains: that $\tilde{v}'$ might share a terminal with $\beta(1)$. But $\beta(1)$ lies on a terminal of $\tilde{\omega}$, so if $\tilde{v}'$ shares this terminal, then $\tilde{\omega}$ and $\tilde{v}$ lie in the same branch of $\Omega$. Both these links cut $\tilde{\chi}$, however, and since $\chi$ respects $\Omega$, they cannot lie in the same branch. We conclude that if $\tilde{v}$ cuts $\beta$, then the terminals of $\tilde{v}'$ lie on the same side of $\beta$. Therefore $\beta$ respects $\Omega$. □

To put Lemma 4e.4 into practice, we define an important class of half-cuts to which it applies.

**Definition 4e.5.** Let $\omega$ route a wire in the design $\Omega$. A half-cut $\sigma$ for a link $\omega$ is **semisimple** in $\Omega$ if there is a simple cut $\chi$ and a necessary crossing $(c,t)$ of $\chi$ by $\omega$ such that $\chi_{0.c}$, as a half-cut for $\omega$ at $t$, is akin to $\sigma$.

One could extend Definition 4e.5, and the following proposition as well, to midcuts. A mid-cut $\tau$ between links $\nu$ and $\omega$ would be semisimple if there were a simple cut $\chi$ and necessary crossings $(a,s)$ and $(b,t)$ of $\chi$ by $\nu$ and $\omega$, respectively, such
that $\chi_{st}$, as a mid-cut between $v$ at $s$ and $\omega$ at $t$, was akin to $\tau$. As natural as the generalization is, I have no use for it.

**Proposition 4e.6.** Semisimple half-cuts are nondegenerate and respectful.

**Proof.** Let $\sigma$ be a semisimple half-cut in the design $\Omega$. The claim is that $\sigma$ respects $\Omega$ and is nondegenerate in $\Omega$. Let $\chi$ be the simple cut whose half-cut $\chi_{0:c}$ is akin to $\sigma$. Then $\chi$ respects $\Omega$ by Proposition 4e.2, and by Lemma 4e.4, every cut associated to $\chi_{0:c}$ respects $\Omega$. Thus $\chi_{0:c}$ respects $\Omega$. Since half-cuts that are akin respect the same designs, $\sigma$ also respects $\Omega$.

To see that $\sigma$, or equivalently $\chi_{0:c}$, is nondegenerate, let $\tilde{\chi}$ and $\tilde{\omega}$ be lifts of $\chi$ satisfying $\tilde{\chi}(c) = \tilde{\omega}(t)$. If $\chi_{0:c}$ were degenerate, then the endpoints of $\tilde{\chi}_{0:c} \ast \tilde{\omega}_{1:1}$ would lie in the same branch of $\Omega$. And since $\omega$ routes a wire $v$ in $\Omega$, this branch would include a lifting $\tilde{v} \in [\tilde{\omega}]_L$ cutting $\tilde{\chi}$, as well as the fringe containing $\tilde{\chi}(0)$. But $\chi$ respects $\Omega$, so this cannot happen. $\square$

### 4F. Safety of Cuts and Half-Cuts

So far we have only considered the topology of designs, using concepts such as necessary crossings and the flows across cuts. Now we mix in some geometry: the arc lengths and capacities of cuts. The result is a powerful set of lemmas concerning the safety of cuts. For example, Corollary 4f.5 shows that an unsafe, major, simple cut gives rise to an unsafe, major, straight cut, and therefore every major simple cut in a safe design is safe.

The technique we use generalizes that discovered by Cole and Siegel [6] and independently by Leiserson and the author. It involves shrinking the cut to its elastic chain, and relating the flow and capacity of the cut to the flows and capacities of the links in the elastic chain. If the cut respects its design, the flow across the chain can be smaller than the flow across the cut only by the widths of the wires whose terminals touch the middle of the chain. But in going from the cut to the chain, the total capacity is reduced by the width of every fringe that touches the middle of the chain. Since wires are no wider than their terminals, the total capacity decreases by at least as much as the total flow. Hence if the cut was unsafe, one of the links in the chain is unsafe as well. Of course, I have glossed over some technical issues, such as the need to ignore minor cuts.

**Redefinition of safety**

In order to accommodate half-cuts as well as cuts, we redefine safety in terms of flow. Let $\theta$ be any cut or half-cut. The **margin** of $\theta$ in a design $\Omega$ is the capacity
of $\theta$ minus the flow across $\theta$:

$$\text{margin}(\theta, \Omega) = \text{cap}(\theta, \Omega) - \text{flow}(\theta, \Omega).$$

The terminology is meant to suggest ‘margin of safety’. We say $\theta$ is unsafe in the design $\Omega$ if and only if $\text{margin}(\theta, \Omega)$ is negative. A cut whose margin is 0 is called marginal (or “marginally safe”).

Chains of links

Proposition 4f.1, which forms the basis for the results of this section, relates the flow across a cut or half-cut to the flow across a chain of links and half-links. Recall from Chapter 3 that a chain for a path in a manifold is any homotopic path, and from Section 3D that every path in a sheet has a unique elastic chain.

A chain for a cut or half-cut consists of major links and gaps, which are defined as follows. Let $\alpha$ be a chain for a cut or half-cut $\sigma$ in a design $\Omega$. If $\sigma$ is a half-cut, then $\alpha$ ends with a half-link $\tau$. Aside from this, $\alpha$ consists of major links $\alpha_1, \ldots, \alpha_n$ interspersed with minor links and paths along fringes. The major links of the chain $\alpha$ are the paths $\alpha_i$, plus $\tau$ if $\tau$ is nondegenerate. (If $\tau$ is degenerate, we say the chain is degenerate.) The portions of $\alpha$ between its major links are called the gaps in $\alpha$. Each gap $\gamma$ can intersect at most one article $C$ of $\Omega$. For a gap consists of empty and degenerate links, and none of these connect different articles of a design. The width of the gap $\gamma$ is defined to be the width of the wire in the article $C$, or zero if $C$ includes no wire.

The flow across a chain is just the sum of the flows across its major links, but the capacity of a chain is slightly more complex. We denote by $\text{gaps}(\alpha)$ the sum of the widths of the gaps in a chain $\alpha$. If $\alpha$ is a chain for $\chi$, the quantity $\text{gaps}(\alpha)$ represents the amount of wiring that might contribute to the flow across $\chi$ but escape detection by any link of $\chi$. The capacity of a chain is the sum of the capacities of its major links, plus the sum of the widths of its gaps.

The flow across a chain

We use three tools to derive sufficient conditions for the existence of an unsafe or marginal link in a chain. One is Proposition 4f.1, coming up, which bounds from below the flow across a chain. Another is Lemma 4f.2, which gives conditions for the existence of a link in the chain. The third is Lemma 4f.3, which bounds from above the capacity of a chain.

**Proposition 4f.1.** Let $\sigma$ be a cut or half-cut that respects the design $\Omega$. If $\alpha$ is a chain of straight links for $\sigma$, then

$$\text{flow}(\alpha, \Omega) \geq \text{flow}(\sigma, \Omega) - \text{gaps}(\alpha). \quad (4-3)$$
Proof. The first step is to reduce the case of a half-cut to that of a cut. If \( \sigma \) is a cut, put \( \eta = \alpha \) and \( \chi = \sigma \). Now suppose \( \sigma \) is a half-cut for a wire \( \omega \) at \( t \). Let \( \chi \) be a cut in \( [\sigma \star \omega_{t:1}]_P \), and let \( \eta \) be the path \( \alpha \star \omega_{t:1} \). Then \( \eta \) is a chain for \( \chi \). The cut \( \chi \) respects \( \Omega \) because \( \sigma \) respects \( \Omega \), and \( \chi \) is associated to \( \sigma \). By Proposition 4b.3 and the definition of the flow across a half-cut, we have \( \text{flow}(\chi, \Omega) = \text{flow}(\sigma, \Omega) \). We also have \( \text{flow}(\eta, \Omega) = \text{flow}(\alpha, \Omega) \), because their nondegenerate links correspond. (Empty links are irrelevant because they have zero flow.) In other words, the only way \( \eta \) and \( \alpha \) differ is that where \( \alpha \) has a half-link, \( \eta \) has an associated cut for that half-link. One is degenerate if and only if the other is degenerate. Hence it suffices to prove

\[
\text{flow}(\eta, \Omega) \geq \text{flow}(\chi, \Omega) - \text{gaps}(\alpha).
\]

(4-4)

![Figure 4f-1. Bounding the flow across a chain.](image)

This figure depicts liftings of certain paths and chains involved in Proposition 4f.1. Here \( \sigma \) is a half-cut for \( \omega \). Its chain \( \alpha \) reaches only to \( \sigma(1) \), so we replace \( \sigma \) by its associated cut \( \chi \), and \( \alpha \) by a chain \( \eta \) for \( \chi \). Though the links of \( \alpha \) are straight, the final link of \( \eta \) need not be.

We apply Lemma 4d.4 to lifts of \( \chi \) and \( \eta \) for each wire \( v \) in \( \Omega \). Using Proposition 2b.4, let \( \tilde{\chi} \) and \( \tilde{\eta} \) be path-homotopic lifts of \( \chi \) and \( \eta \), respectively. Let \( \eta_1, \ldots, \eta_m \) be the links of \( \eta \), and let \( \tilde{\eta}_1, \ldots, \tilde{\eta}_m \) be the corresponding subpaths of \( \tilde{\eta} \). By Lemma 4d.4, any lift of \( v \) that cuts \( \tilde{\chi} \) either cuts \( \tilde{\eta}_i \) for some \( i \), or else has as terminal a fringe that intersects \( \tilde{\eta} \) but not \( \tilde{\chi} \). Let \( B \) denote the branch of \( v \) that contains this fringe. Because \( \chi \) respects \( \Omega \), the branch \( B \) contains at most one lift of \( \tilde{v} \) that cuts \( \tilde{\chi} \), and \( B \) does not include either terminal of \( \tilde{\chi} \).

The contribution of \( B \) to \( \text{flow}(\chi, \Omega) \) can be charged to a gap in \( \alpha \) of width \( \text{width}(v) \) in such a way that no gap is charged twice. Let \( M \) be the blanket, and let \( \tilde{\alpha} \) be the lift of \( \alpha \) to \( M \) such that \( \tilde{\alpha}(0) = \tilde{\eta}(0) = \tilde{\chi}(0) \). Pick \( x \in (0,1) \) such that \( \tilde{\alpha}(x) \in B \cap Bd M \), and let \( [s,t] \) be the maximal interval containing \( x \) such that \( \gamma = \alpha_{st} \) consists of minor links interspersed with paths along fringes. If \( \alpha \) ends with a degenerate half-cut, this may be included in \( \gamma \). The only fringes that \( \gamma \) touches are those in the article of \( \Omega \) that includes \( Im v \). Let \( \tilde{\gamma} \) be the subpath of \( \tilde{\eta} \) corresponding to \( \gamma \).
To show that $\gamma$ is a gap of $\alpha$, it is enough to prove $s \neq 0$ and $t \neq 1$. The links (and possible terminating half-link) of $\gamma$ are straight and minor, and since the terminals of $\nu$ are convex, each link is nonempty and therefore degenerate instead. All the fringes that intersect $\tilde{\gamma}$ are part of the same branch $B$. So if $s = 0$, then $B$ contains $\tilde{\gamma}(0) = \tilde{\chi}(0)$. Or if $t = 1$, then the final link or half-link of $\alpha$ is degenerate, and $B$ contains $\tilde{\chi}(1)$. But $B$ includes neither terminal of $\tilde{\chi}$, so $s > 0$ and $t < 1$. Thus $\gamma$ is a gap of $\alpha$, and its width is $\text{width}(\nu)$. We charge the contribution of $B$ to $\gamma$. Because all the fringes that $\tilde{\gamma}$ touches are part of $B$, and $B$ contains at most one lift of $\tilde{\nu}$ that cuts $\tilde{\chi}$, the gap $\gamma$ is charged only once.

The upshot of this analysis is that the difference between the flow across $\chi$ and the flows across the links $\eta_i$ is accounted for by the widths of the gaps in $\alpha$. In symbols,

$$\text{flow}(\chi, \Omega) - \sum_{i=1}^{m} \text{flow}(\eta_i, \Omega) \leq \text{gaps}(\alpha).$$

To prove inequality (4–5), it remains to identify $\text{flow}(\eta, \Omega)$, the sum of the flows across the major links of $\eta$, with $\sum_{i=1}^{m} \text{flow}(\eta_i, \Omega)$. In other words, we must show that the minor links of $\eta$ contribute nothing to the flow across $\chi$. Certainly the empty ones contribute nothing, because their flow is zero in $\Omega$. The straight degenerate links, also, have zero flow in $\Omega$ by Lemma 4e.3. The remaining case is the final link $\eta_m$ of $\eta$, which can be nonsimple if $\sigma$ is a half-cut. If $\eta_m$ is degenerate, then the endpoints of $\tilde{\eta}_m$ lie in the same branch $T$ of $\Omega$. Thus $\tilde{\eta}_m$ is path-homotopic to a path in a single branch $T$ of $\Omega$, and hence it cannot cut links in any other branch. The branch $T$ contains a terminal of $\tilde{\chi}$, because $\tilde{\eta}_m(1)$ and $\tilde{\chi}(1)$ lie on the same fringe. Because $\chi$ respects $\Omega$, no link in $T$ that lifts a wire in $\Omega$ can cut $\tilde{\chi}$. Therefore no lift of a wire in $\Omega$ can cut both $\tilde{\eta}_m$ and $\tilde{\chi}$. In other words, $\eta_m$ contributes nothing to the flow across $\chi$. 

Elastic chains

As mentioned before, our strategy is to reduce a major cut or nondegenerate half-cut to the major links in its elastic chain. The next lemma ensures that some major link always remains. It uses the fact that all the links and half-links in an elastic chain are straight (Lemma 3d.5).

**Lemma 4f.2.** The elastic chain for a major cut or a nondegenerate half-cut includes at least one major link.

**Proof.** First consider the case of a cut. Let $\chi$ be a major cut in the design $\Omega$, and let $\alpha$ be the elastic chain for $\chi$. If the endpoints of $\chi$ lie in different articles of $\Omega$, then $\alpha$ must contain a link that passes between two different articles, and this link is major. So we assume that all the links in $\alpha$ have their endpoints in the same
article. Suppose first that this article contains no wire of $\Omega$. Then $\text{gaps}(\alpha) = 0$, but since $\chi$ is nonempty, its flow is positive. Hence by Proposition 4f.1, the flows across the major links of $\alpha$ sum to a positive quantity, and therefore $\alpha$ must have a major link. Now suppose the article contains a wire $\omega$ of $\Omega$. Using Proposition 2b.4, lift $\chi$ and $\alpha$ to path-homotopic paths $\tilde{\chi}$ and $\tilde{\alpha}$. The endpoints of $\tilde{\chi}$ lie in different branches of $\omega$ because $\chi$ is nondegenerate. Hence $\tilde{\alpha}$ contains a link that passes between two branches of $\omega$, and consequently $\alpha$ has a nondegenerate link $\beta$. Since $\beta$ is straight and terminals are convex, $\beta$ has two terminals. Thus $\beta$ is a nonempty, nondegenerate link of $\alpha$.

Now consider the case of a half-cut. Let $\sigma$ be a nondegenerate half-cut for a link $\omega$ at $t$, and let $\eta$ be the elastic chain for $\sigma$. Let $\chi$ be a cut in $[\sigma * \omega_{t_1}]$, and let $\alpha$ be the chain $\eta * \omega_{t_1}$ for $\chi$. The cut $\chi$ is nondegenerate in $\Omega$ because $\sigma$ is nondegenerate in $\Omega$ and $\chi$ is associated to $\sigma$. As before, we may assume that links of $\alpha$ all lie in the same article of $\Omega$. This article contains a wire in $\Omega$, of which $\omega$ is a route. Let $\tilde{\chi}$ and $\tilde{\alpha}$ be path-homotopic lifts of $\chi$ and $\alpha$. Then the endpoints of $\tilde{\chi}$ lie in different branches of $\omega$. Hence $\tilde{\alpha}$ contains a link that passes between two branches of $\omega$, and consequently $\alpha$ has a nondegenerate link $\beta$. If $\beta$ is part of $\eta$, then $\beta$ is straight, and hence nonempty as shown above. Otherwise $\beta$ is an associated cut of the final half-cut in $\eta$. Since $\beta$ is nondegenerate, this half-cut is nondegenerate, and again $\eta$ includes a major link. □

The most important fact about elastic chains is that they have minimal euclidean arc length, and minimal arc length in all other norms as well (Corollary 3d.8). Combined with the following lemma, this fact implies that the capacity of a cut or half-cut is no smaller than the capacity of its elastic chain.

**Lemma 4f.3.** The elastic chain $\theta$ of a major cut or nondegenerate half-cut $\sigma$ satisfies $\text{cap}(\theta) - ||\theta|| \leq \text{cap}(\sigma) - ||\sigma||$, and the inequality is strict if $\theta$ is degenerate.

**Proof.** We assume, of course, that the widths of wires and fringes are specified by some design $\Omega$. Let $\theta_1, \ldots, \theta_n$ be the major links of $\theta$, where $n \geq 1$ by Lemma 4f.2. For $1 \leq i < n$ the points $\theta_{i}(1)$ and $\theta_{i+1}(0)$ lie in the same article $C_i$ of $\Omega$. Let $C_0$ denote the article containing $\theta_1(0)$, and let $C_n$ be the article containing $\theta_n(1)$. For $1 \leq i \leq n$, let $a_i$ be the width of the detail containing $\theta_i(0)$, and let $b_i$ be the width of the detail containing $\theta_i(1)$. Also put $w_i$ equal the width of the wire in the article $C_i$, if any, and otherwise put $w_i = 0$. Then $\sum_{i=1}^{n-1} w_i = \text{gaps}(\theta)$, and hence by the definition of the capacity of $\theta$ we have

$$\text{cap}(\theta) = \sum_{i=1}^{n} \text{cap}(\theta_i) + \sum_{i=1}^{n-1} w_i. \quad (4-5)$$

No wire is wider than either of its terminals, and therefore $w_i \leq a_i$ and $w_i \leq b_i$ for
Figure 4f.2. A simple cut and its elastic chain. Here $\sigma$ is a cut in a design whose wires are shown in grey. The elastic chain $\theta$ for $\sigma$ has three minor links (thin black lines) and four major links $\theta_1, \ldots, \theta_4$ (thick black lines).

Each $i$.

Now we evaluate inequality (4-3) using the definition of capacity for cuts:

$$
cap(\theta) = \sum_{i=1}^{n} (||\theta_i|| - a_i/2 - b_i/2) + \sum_{i=1}^{n-1} w_i
$$

$$
\leq \left( \sum_{i=1}^{n} ||\theta_i|| \right) - a_1/2 - b_n/2
$$

$$
\leq ||\theta|| - a_1/2 - b_n/2. \tag{4-6}
$$

The final inequality holds because the paths $\theta_i$ are disjoint subpaths of $\theta$, and hence is strict if $\theta$ contains anything but major links. In particular, it is strict if $\theta$ is degenerate. Comparing inequality (4-6) to $cap(\sigma)$, which is $||\sigma|| - a_1/2 - b_n/2$, we get the inequality

$$
cap(\theta) - ||\theta|| \leq -a_1/2 - b/2 \leq cap(\sigma) - ||\sigma||,
$$

with strictness if $\theta$ is degenerate. □

Applications

There are three main applications of Proposition 4f.1 and Lemmas 4f.2 and 4f.3. One concerns unsafe cuts, one concerns unsafe half-cuts, and the third concerns half-cuts of margin zero. The first two are given here; the third is discussed in Section 5B. All the applications start with a cut or half-cut, compare the flow and capacity of an elastic chain, and show that one of the links in that chain must have low margin.

Lemma 4f.4. Let $\chi$ be a major simple cut in a design $\Omega$. There is a major straight cut $\beta$ satisfying $\text{margin}(\beta, \Omega) \leq \text{margin}(\chi, \Omega)/n$ for some $n > 0$.

Proof. Let $\alpha$ be the elastic chain for $\chi$. By Lemma 4f.3 and Corollary 3d.8, the chain $\alpha$ satisfies $\text{cap}(\alpha) \leq \text{cap}(\chi)$. Let $\alpha_1, \ldots, \alpha_n$ be the major links of $\alpha$. Because
\( \chi \) is simple, it respects \( \Omega \), by Proposition 4e.2. Hence Proposition 4f.1 applies to \( \chi \), \( \alpha \), and \( \Omega \). The result is

\[
\sum_{i=1}^{n} \text{flow}(\alpha_i, \Omega) \geq \text{flow}(\chi, \Omega) - \text{gaps}(\alpha).
\]

The definition of \( \text{cap}(\alpha) \) and the fact that \( \text{cap}(\alpha) \leq \text{cap}(\chi) \) imply

\[
\sum_{i=1}^{n} \text{cap}(\alpha_i, \Omega) \leq \text{cap}(\chi) - \text{gaps}(\alpha).
\]

Subtracting the previous inequality from this one shows that \( \sum_{i=1}^{n} \text{margin}(\alpha_i, \Omega) \leq \text{margin}(\chi, \Omega) \). Because \( n > 1 \), by Lemma 4f.2, there must be some link \( \beta \) among the \( \alpha_i \) such that \( \text{margin}(\beta, \Omega) \leq \text{margin}(\chi, \Omega)/n \). This link \( \beta \) is a major straight cut. \( \Box \)

If one applies Lemma 4f.4 to a major simple cut with negative margin, the major straight cut it produces has negative margin, and is therefore unsafe. In a safe design, this cannot happen.

**Corollary 4f.5.** All major simple cuts in a safe design are safe. \( \Box \)

The second application concerns half-cuts. Its proof is very similar to that of Lemma 4f.4, except for some additional concern about degeneracy.

**Lemma 4f.6.** Let \( \omega \) route a wire in a safe design \( \Omega \). If \( \omega \) has an unsafe, nondegenerate, simple half-cut that respects \( \Omega \), then \( \omega \) has an unsafe, nondegenerate, straight half-cut.

**Proof.** Let \( \sigma \) be the unsafe, nondegenerate, simple half-cut for \( \omega \), and let \( \alpha \) be the unique elastic chain for \( \sigma \). Let \( \alpha_1, \ldots, \alpha_n \) be the major links of \( \alpha \). Lemma 4f.2 and Corollary 3d.8 imply \( \text{cap}(\alpha) \leq \text{cap}(\sigma) \), and Lemma 3d.5 implies that the links of \( \alpha \) are straight. Because \( \sigma \) respects \( \Omega \), Proposition 4f.1 shows that

\[
\sum_{i=1}^{n} \text{flow}(\alpha_i, \Omega) \geq \text{flow}(\sigma, \Omega) - \text{gaps}(\alpha).
\]

By the definition of \( \text{cap}(\alpha) \) and the fact that \( \text{cap}(\alpha) \leq \text{cap}(\sigma) \), we also have

\[
\sum_{i=1}^{n} \text{cap}(\alpha_i, \Omega) \leq \text{cap}(\sigma) - \text{gaps}(\alpha).
\]
Subtracting the previous inequality from this one shows that

\[ \sum_{i=1}^{n} \text{margin}(\alpha_i, \Omega) \leq \text{margin}(\sigma, \Omega). \]  \hspace{1cm} (4-7)

The right-hand side of (4-7) is negative by assumption. Since \( \Omega \) is safe, every major straight cut in \( \Omega \) has nonnegative margin. Hence not every \( \alpha_i \) can be a link; \( \alpha_n \) must be a nondegenerate half-cut \( \tau \) for \( \omega \). Then inequality (4-7) implies \( \text{margin}(\tau, \Omega) \leq \text{margin}(\sigma, \Omega) - \sum_{i=1}^{n-1} \text{margin}(\alpha_i, \Omega) \), which means \( \text{margin}(\tau, \Omega) < 0 \). Therefore \( \tau \) is an unsafe, nondegenerate, straight half-cut for \( \omega \).  \( \square \)
Chapter 5

Routing a Safe Design

The most natural way to prove that a safe design has a proper embedding is to construct one. And, in fact, all the methods I have considered for proving the design routability theorem are essentially constructive. But the construction can tend toward either of two extremes. It might be a deterministic algorithm that builds the embedding step by step, maintaining some invariant that ensures that the final embedding is proper. Or it might be a mathematical description that distinguishes a certain design; the description could involve limits and other infinitary "constructors". One would then need to prove the existence of the limits, and deduce from the description that the resulting design is proper.

Many methods for proving the design and sketch routability theorems succeed without being particularly enlightening. One algorithmic approach, for example, is to begin with rubber bands and slowly move them apart, keeping them taut and bending them as necessary, until they reach their natural width. This process gives rise to a constructive proof of the sketch routability theorem and a routing algorithm that runs in time \( \Theta(n^7) \) or so [33]. The mathematical approach, also, can probably be made to work. One can prove theorems similar to the sketch routability theorem in the grid-based wiring model, as Cole and Siegel claimed in [6]. Letting the grid size approach zero and taking the limit of embeddings with minimal wire length, one can probably obtain proper embeddings for safe sketches in other wiring models as well. (There are some difficult technical issues concerning self-avoidance.) But again, this approach gives little guidance for developing an efficient routing algorithm and proving it correct.

My construction, presented in this chapter, is a compromise that lies closer to the mathematical extreme. Given a safe design, we first construct an evasive route for each wire. We then prove that each wire in the design has a minimum-length evasive route. (Here a limiting process comes in, via Proposition 2c.8.) Finally, we characterize these routes in sufficient detail to show that they form a proper design. Thus we prove the hard direction of the design routability theorem, and with some extra work in Chapter 6, we get the design routing theorem as well. But the real advantage of this approach shows up in Chapter 7: the information it provides
about ideal routes allows us to develop efficient algorithms for constructing them.

Inspiration

The idea of using minimum-length evasive routes for single-layer routing first appeared in a paper by Tompa [52], who considers river routing across a channel. I have adapted his construction to the case of a multiply connected routing region, but the outline of the proof is the same. Unfortunately, the technical difficulties of working in a blanket, rather than in the plane, make my proof about fifteen times longer than his.

A simplified problem

What follows is a brief overview of Tompa’s construction. I have simplified it by using a piecewise linear wiring norm rather than the euclidean norm used in [52]. Suppose one wishes to connect terminals $a_1, \ldots, a_n$ on the bottom of a rectangular channel to terminals $b_1, \ldots, b_n$ on the top, using “wires” of unit width. Assume these terminals are numbered from left to right. We argue that the conditions

$$\begin{align*}
\|a_i - a_j\| &\geq |i - j| \\
\|a_i - b_j\| &\geq |i - j| \\
\|b_i - b_j\| &\geq |i - j|
\end{align*}$$

(*)

are necessary and sufficient for the channel to be routable by wires that remain at least one unit apart. We can interpret these inequalities as saying that certain nondegenerate cuts are safe. If $c_i \in \{a_i, b_i\}$ and $c_j \in \{a_j, b_j\}$, then the “capacity” of the “cut” from $c_i$ to $c_j$ is $\|c_i - c_j\| - 1$; the “flow” across this cut is $\max\{0, |i - j| - 1\}$; and if $i = j$, then the cut from $c_i$ to $c_j$ is “degenerate”. See Figure 5-1(i).

Suppose that one of the inequalities in (*) fails to hold. Then for some $c_i$ and $c_j$, with $i \neq j$, the line segment $\chi$ from $c_i$ to $c_j$ has length less than $|i - j|$. No matter how the wires are routed, each of the wires whose index lies between $i$ and $j$ must cross $\chi$. Counting also the wires $i$ and $j$, which intersect the endpoints of $\chi$, there are $|i - j| + 1$ different wires that must intersect $\chi$. They cannot do so and still remain one unit apart. Therefore condition (*) is necessary.

To show that condition (*) is sufficient, we route the wires assuming that it holds. From the perspective of wire $i$, each terminal $c_j \in \{a_j, b_j\}$ for $j \neq i$ presents a barrier for wire $i$. The barrier surrounding $c_j$ is the set $\{ x : \|x - a_j\| < |i - j| \}$. We call this a left barrier if $j < i$, and a right barrier if $j > i$. Part (ii) of Figure 5-1 suggests why left and right barriers do not intersect. If the barrier around $c_j$ with $j < i$ intersected the barrier around $c_k$ with $k > i$, there would be a point $x$ such that $\|x - c_j\| < |i - j|$ and $\|x - c_k\| < |i - k|$. Then the triangle inequality would
Figure 5-1. A simplified routing problem. The chief ideas behind the routing of safe designs are amply illustrated by the much simpler problem of river routing in a simply connected region. This problem supports analogues of the design routability and routing theorems. Under natural definitions of safety and degeneracy for cuts, shown in part (i), the terminals of a channel can be connected by wires of unit width if and only if every nondegenerate cut is safe. Furthermore, if the channel can be routed, one can route it using minimum-length evasive wires, as shown in part (iii). An evasive wire is one that avoids the barriers presented by the terminals of other wires (part (iv)). The remaining parts illustrate the argument that the minimum-length evasive wires exist and are sufficiently separated.
Chapter 5

Routing a Safe Design

imply $||c_j - c_k|| < |j - k|$, contradicting (*). Hence there is a path from $a_i$ to $b_i$ that avoids both its left and right barriers. Such a path is called an evasive route for wire $i$. We route wire $i$ by choosing the minimum-length evasive path $\omega_i$ from $a_i$ to $b_i$, as shown in part (iii) of Figure 5.1. Part (iv) of that figure illustrates the barriers for a particular wire.

Now for the interesting part: we prove, in three steps, that the minimum-length evasive routes $\omega_i$ stay one unit apart. The first step appeals to the reader’s geometric intuition, although a rigorous proof could be provided. Because the wiring norm is piecewise linear, the barriers are polygonal. The first claim is that the routes are piecewise straight. We also claim that wherever $\omega_i$ has a joint $t$, the point $\omega_i(t)$ lies on the frontier of a barrier around some terminal $c_k$, and $\omega_i$ turns toward $c_k$ at $t$.

The second step shows that none of the routes cross over. If $\omega_i$ and $\omega_j$ cross over, then they form a polygon as shown in Figure 5.1(v). The portion of $\omega_i$ in this polygon lies on the opposite side of $\omega_j$ from $a_i$ and $b_i$. Similarly, the portion of $\omega_j$ in this polygon lies on the opposite side of $\omega_i$ from $a_j$ and $b_j$. Like all polygons, this one has three or more internal angles of measure less than $\pi$. Two of these can lie at the points where $\omega_i$ and $\omega_j$ intersect, but at the third angle $x$ one of the routes, say $\omega_i$, turns toward the inside of the polygon. Let $c_k$ be the terminal whose barrier $\omega_i$ bends around at $x$. Then the line segment from $x$ to $c_k$ lies within this barrier, and its length is $|i - k|$. This line segment intersects $\omega_j$. Moreover, the terminals of $\omega_j$ lie on the opposite side of $\omega_i$ from $c_k$. Therefore $|j - k|$ is greater than $|i - k|$, and so $\omega_j$ comes closer than $|j - k|$ units to $c_k$. But this means $\omega_j$ enters the barrier presented to it by $c_k$.

The third step shows that the routes $\omega_i$ stay at least one unit apart. Suppose $\omega_i$ comes within $d < 1$ units of $\omega_j$, as shown in Figure 5.1(vi). Then either a terminal of one comes within $d$ units of the other, contradicting evasiveness, or else there is a point $x$ at which one route, say $\omega_i$, lies within $d$ units of the other, but turns away from it. Let $c_k$ be the terminal whose barrier $\omega_i$ bends around at $x$. Then $\omega_i$ turns toward $c_k$ at $x$, and since $\omega_j$ does not cross over $\omega_i$, it follows that $c_k$ lies on the opposite side of $\omega_i$ from the terminals of $\omega_j$. Hence the barrier for $\omega_j$ around $c_k$ has radius at least $|i - k| + 1$. We have $||x - c_k|| = |i - k|$, and hence by the triangle inequality for norms, $\omega_j$ comes within $|i - k| + d$ units of $c_k$. Again $\omega_j$ enters the barrier presented to it by $c_k$. We conclude that the routes $\omega_j$ actually do stay one unit apart.

Outline of the construction

Many ideas from river routing in a channel carry over to the design routing problem. The first insight that applies is this: a feasible wire must stay far enough away from the fringes, other than its terminals, to allow the other wires to be routed.
We express this constraint in our model by saying that a route of a wire $\omega$ is evasive in a design $\Omega \ni \omega$ if every nontrivial straight half-cut for that route is safe in $\Omega$. Evasiveness alone does not guarantee feasibility, however. An evasive link can be divisive, and it can have self-intersections. Moreover, it might not leave enough space for wires to pass between two different portions of itself. But if one chooses a canonical, evasive route of minimum length, these problems go away; no two parts of the route are close together except where it is necessary. A route $\alpha$ of a wire $\omega$ in a safe design $\Omega$ is called ideal for $\Omega$ if $\alpha$ is canonical, evasive in $\Omega$, and has minimum length among all routes of $\omega$ that are evasive in $\Omega$.

To prove that the ideal routes form a design, we analyze the points at which they turn. Intuitively, wherever an ideal link has a joint, it is being pushed away from a fringe by the evasiveness condition. As we show in Section 5B, there is a marginally safe half-cut from that fringe to the joint, and it lies interior to the angle made by the link at that joint. (Ideal links are piecewise straight.) From this result we can deduce many properties of the ideal routes. For example, if the ideal routes of two wires were to cross over, they would form a loop, and one of the two links would turn toward the inside of the loop. But then the marginally safe half-cut would contain an unsafe nontrivial half-cut for the other link, contradicting the evasiveness of that link. Therefore the ideal routes do not cross. A similar argument shows that they have no self-intersections, and are actually wires.

Furthermore, ideal wires are sufficiently separated. For suppose that two ideal wires were to come closer than the mean of their widths, causing their extents to overlap. Then either their terminals would be too close, or else one wire would turn away from the other at a point where they were close. Concatenating the marginally safe half-cut to that turning point with a short straight path to the other wire yields an unsafe, nondegenerate, bent half-cut for the second ideal wire, and thus by Lemma 4f.6, an unsafe, nondegenerate (and hence nontrivial), straight half-cut. Again, this would contradict the evasiveness of the second wire. If one carefully applies this argument to different parts of the same wire, one can also show that ideal wires are self-avoiding. These are the main steps in the proof that the ideal routes form a proper design.

5A. Construction of Ideal Routes

The result of this section is quite simple: every wire in a safe sketch has an ideal route. We prove this result in two steps. Given a wire in a safe sketch, we first attempt to construct an evasive route for it. If this construction were to fail, we show, the sketch could not be safe. Then we prove that the family of evasive routes for the wire has a minimum-length, canonical element: an ideal route.
In my initial attempts to prove a routability theorem, I tried to construct evasive routes of wires by pushing the wires away from the fringes from which they had unsafe straight half-cuts. I hoped to show that if the process failed to converge, then the wire had straight, unsafe, nontrivial half-cuts pushing it from both sides, and these half-cuts combined to form an unsafe, bent, major cut. By a result like Corollary 4f.5, the existence of this cut would contradict the safety of the design. These attempts failed, partly due to the difficulty of defining the "sides" of a wire in the plane. But in a blanket, where the sides of a simple link are well defined, the idea works. For every wire in a safe sketch we identify left and right forbidden zones in the blanket, and show that they do not intersect. Further maneuvering shows that the wire has an evasive route, which in turn yields an ideal route.

Forbidden zones

The blanket provides us with a spatial characterization of evasiveness. Given a wire \( \omega \) and a lifting \( \tilde{\omega} \) of that wire, we identify two forbidden zones, one to the left of \( \tilde{\omega} \) and one to the right. A route of \( \omega \) need only have a lifting in \( [\tilde{\omega}]_L \) that avoids these zones in order to be evasive. It follows that \( \omega \) has an evasive route if the forbidden zones for \( \tilde{\omega} \) do not separate the terminals of \( \tilde{\omega} \).

**Definition 5a.1.** Let \( \Omega \) be a design on a sheet \( S \), and let \( p: M \to S \) be the covering map. Let \( \tilde{\omega} \) be a lift of a wire \( \omega \in \Omega \). A straight half-link \( \tilde{\sigma} \) in \( M \) is forbidden to \( \tilde{\omega} \) if for some link \( \tilde{v} \in [\tilde{\omega}]_L \) the path \( p \circ \tilde{\sigma} \) is an unsafe, nontrivial half-cut for \( p \circ \tilde{v} \). The left-hand (right-hand) forbidden zone for \( \tilde{\omega} \) is the set of all endpoints \( \tilde{\sigma}(1) \) of the forbidden half-links \( \tilde{\sigma} \) for \( \tilde{\omega} \) with \( \tilde{\sigma}(0) \in \text{left}(\tilde{\omega}) \) (or \( \tilde{\sigma}(0) \in \text{right}(\tilde{\omega}) \)).

The forbidden zones for \( \tilde{\omega} \) depend only on its link-homotopy class, or equivalently, its terminals. The choice of the link \( \tilde{\omega} \) is also irrelevant, provided that it passes through \( \tilde{\sigma}(1) \). This point will be clarified in Lemma 5a.2.

Requiring that \( p \circ \tilde{\sigma} \) be nontrivial is equivalent to requiring that \( \tilde{\sigma}(0) \) not lie on a terminal of \( \tilde{\omega} \). Therefore every forbidden half-link \( \tilde{\sigma} \) for \( \tilde{\omega} \) satisfies either \( \tilde{\sigma}(0) \in \text{left}(\tilde{\omega}) \) or \( \tilde{\sigma}(0) \in \text{right}(\tilde{\omega}) \), and hence contributes to one of the forbidden zones for \( \tilde{\omega} \). Conversely, every sufficiently short straight half-link \( \tilde{\sigma} \) is forbidden unless it shares a terminal with \( \tilde{\omega} \). For if \( \|\tilde{\sigma}\| < \text{width}(\omega)/2 \), then then \( \text{cap}(p \circ \tilde{\sigma}) < 0 \), and consequently \( p \circ \tilde{\sigma} \) is unsafe. Therefore every point sufficiently close to a fringe of \( M \) that is not a terminal of \( \tilde{\omega} \) belongs to a forbidden zone of \( \tilde{\omega} \).

Connection with evasiveness

The first consequence of Definition 5a.1 is that one can find an evasive route of \( \omega \) by finding a link that is homotopic to \( \tilde{\omega} \) and avoids its forbidden zones.
**Lemma 5a.2.** Let \( \omega \) be a wire with lift \( \tilde{\omega} \). The projection of a link \( \tilde{\rho} \in [\tilde{\omega}]_L \) is evasive if and only if \( \tilde{\rho} \) avoids the forbidden zones of \( \tilde{\omega} \).

**Proof.** Let \( p \) denote the covering map. First we tackle the "if" direction. Suppose that \( p \circ \tilde{\rho} \) is not evasive; let \( \sigma \) be an unsafe, straight, nontrivial half-cut for \( \tilde{\rho} \) at \( t \). Let \( \tilde{\sigma} \) be a lift of \( \sigma \) with \( \tilde{\sigma}(1) = \tilde{\rho}(t) \). Then \( \tilde{\sigma} \) is a forbidden half-link for \( \tilde{\omega} \) (take \( \tilde{v} = \tilde{\rho} \)), and hence \( \tilde{\sigma}(1) = \tilde{\rho}(t) \) lies in a forbidden zone of \( \tilde{\omega} \).

Now we prove the "only if" direction. If \( \tilde{\rho} \) enters a forbidden zone of \( \tilde{\omega} \), then we have \( \tilde{\sigma}(1) = \tilde{\rho}(r) \) for some forbidden half-link \( \tilde{\sigma} \) and some point \( r \in (0, 1) \). Hence for some link \( \tilde{v} \in [\tilde{\omega}]_L \) and some point \( t \in I \), the path \( p \circ \tilde{\sigma} \) is an unsafe, straight, nontrivial half-cut for \( p \circ \tilde{v} \) at \( t \). But the half-cuts \( p \circ \tilde{\sigma} \) for \( p \circ \tilde{\rho} \) at \( r \), and \( p \circ \tilde{\sigma} \) for \( p \circ \tilde{v} \) at \( t \), are akin (Definition 4d.1); they have the same flow, and neither is trivial. And obviously they have the same capacity. Therefore \( p \circ \tilde{\sigma} \) is an unsafe, straight, nontrivial half-cut for \( p \circ \tilde{\rho} \) at \( r \), and \( p \circ \tilde{\rho} \) is not evasive. \( \square \)

Next we show that the forbidden zones for a link do not intersect. The construction is illustrated in Figure 5a-1.

![Figure 5a-1](image)

**Figure 5a-1.** In a safe design, forbidden zones are disjoint. Here \( \tilde{\omega} \) lifts a wire \( \omega \). If its forbidden zones intersect, some point \( z \) (small circle) is touched by forbidden half-links \( \tilde{\sigma} \) and \( \tilde{\tau} \) from both sides. The concatenation \( \tilde{\chi} \) of these half-links cuts \( \tilde{\omega} \). Hence its projection is a bent cut \( \chi \) that makes a necessary crossing of \( \omega \), and the half-links corresponding to this crossing are unsafe. We infer that the bent cut itself is unsafe and major.

**Lemma 5a.3.** Let \( \omega \) be a wire in a safe design, and let \( \tilde{\omega} \) lift \( \omega \). The forbidden zones for \( \tilde{\omega} \) are disjoint.

**Proof.** Call the design \( \Omega \). Let \( S \) be the sheet of \( \Omega \), let \( M \) be its blanket, and let \( p: M \to S \) be the covering map. Suppose \( \tilde{\omega} \) splits \( M \) into scraps \( L \) and \( R \), and let \( z \) be a point in both forbidden zones of \( \tilde{\omega} \). Let \( \tilde{v} \) be any link in \([\omega]_L \) that passes through \( z \); say \( \tilde{v}(t) = z \). Applying Definition 5a.1 to \( z \), we find straight half-links \( \tilde{\sigma} \) and \( \tilde{\tau} \) ending at \( z \) such that \( \tilde{\sigma}(0) \in L \), \( \tilde{\tau}(0) \in R \), and both \( \sigma = p \circ \tilde{\sigma} \) and \( \tau = p \circ \tilde{\tau} \) are unsafe, straight, nontrivial half-cuts for \( p \circ \tilde{v} \) at \( t \). Thus if \( A \) and \( B \) are the
terminals of $\sigma$ and $\tau$ respectively, we have

$$\text{flow}(\sigma, \Omega) > \text{cap}(\sigma, \Omega) = \|\sigma\| - \text{width}(A)/2 - \text{width}(\omega)/2;$$

$$\text{flow}(\tau, \Omega) > \text{cap}(\tau, \Omega) = \|\tau\| - \text{width}(B)/2 - \text{width}(\omega)/2.$$  

We find an unsafe, major, simple cut in $\Omega$, which by Corollary 4f.5 implies that $\Omega$ is not safe. Let $\tilde{\chi}$ equal $\tilde{\sigma} \ast \tilde{\tau}_{1,0}$, and put $\chi = p \circ \tilde{\chi}$. Then $\chi$ is a bent cut. Because $\sigma$ and $\tau$ are nontrivial, their terminals lie wholly in opposite scraps of $\tilde{\omega}$, and hence the link $\tilde{\chi}$ cuts $\tilde{\omega}$. Therefore the crossing $(\frac{1}{2}, t)$ of $\chi$ by $p \circ \tilde{v}$ is necessary. By Proposition 4d.2, we have

$$\text{flow}(\chi, \Omega) \geq \text{flow}(\chi_{0:1/2}, \Omega) + \text{flow}(\chi_{1:1/2}, \Omega) + \text{width}(\omega)$$

$$= \text{flow}(\sigma, \Omega) + \text{flow}(\tau, \Omega) + \text{width}(\omega)$$

$$> \|\sigma\| + \|\tau\| - \text{width}(A)/2 - \text{width}(B)/2.$$  

Because $\|\chi\| = \|\sigma\| + \|\tau\|$, the final quantity is just the capacity of $\chi$. We conclude that $\text{flow}(\chi, \Omega)$ exceeds $\text{cap}(\chi, \Omega)$, making $\chi$ unsafe. Because its flow is nonzero, $\chi$ is nonempty in $\Omega$. And since $\chi$ is simple, Lemma 4e.3 implies that $\chi$ is nondegenerate. Hence $\chi$ is major in $\Omega$. 

**Forbidden zones are made of barriers**

The fact the forbidden zones for a link $\tilde{\omega}$ do not intersect does not immediately imply that the zones can be avoided by a piecewise linear link between the terminals of $\tilde{\omega}$. To construct this link, we analyze the forbidden zones themselves.

First we chop the fringes of the blanket into small pieces. Let $\Omega$ be a design on the sheet $S$, and let $M$ be a blanket of $S$ with covering map $p: M \to S$. Let $\omega$ be a wire in $\Omega$, and let $\tilde{\omega}$ be a lift of $\omega$ to $M$. Choose $\epsilon$ smaller than the minimum dimension of the fringes of $S$, and cover $Bd S$ with connected open sets of size $\epsilon$ or less. Because $Bd S$ is compact, finitely many sets suffice. Each set in the resulting open cover is contractible and locally path-connected, and hence can be lifted to $M$ by Proposition 2b.8.

Each lift of these fringe pieces gives rise to some part of a forbidden zone. For each lift $U$ of a set $V$ in the open cover, we define the barrier for $\tilde{\omega}$ growing from $U$ to be the set of endpoints $\sigma(1)$ of forbidden half-links $\tilde{\sigma}$ for $\tilde{\omega}$ with $\tilde{\sigma}(0) \in U$. The base of this barrier is the fringe containing $U$. A barrier for $\tilde{\omega}$ is a left-hand barrier or right-hand barrier according to whether its base lies in $\text{left}(\tilde{\omega})$ or $\text{right}(\tilde{\omega})$.

Both forbidden zones for $\tilde{\omega}$ are unions of barriers for $\tilde{\omega}$. The following lemma relates the barriers and zones more directly.

**Lemma 5a.4.** Let $\tilde{\omega}$ lift a wire $\omega$ in the design $\Omega$; let $Z$ be the right-hand forbidden zone of $\tilde{\omega}$ and put $R = \text{right}(\tilde{\omega})$. Then
(1) every barrier for $\tilde{\omega}$ is a lift of the inside of a polygon;
(2) only finitely many right-hand barriers for $\tilde{\omega}$ intersect $\tilde{\omega}$; and
(3) the union $X$ of those barriers satisfies $Z - R \subseteq X \subseteq Z$.

Proof. Say $\omega$ is a wire in the design $\Omega$ on the sheet $S$. Call a point $x$ in a flat manifold visible from a set $U$ in that manifold if there is a straight path from $x$ to a point of $U$. Combining Definition 5.1.1 with the definition of the flow across a half-cut, we characterize barriers as follows:

Claim: A point $x \in M$ is in the barrier for $\tilde{\omega}$ growing from $U$ if and only if there is a straight half-link $\tilde{\sigma}$ from $U$ to $x$, and a half-link $\tilde{\alpha}$ from the fringe containing $\tilde{\omega}(1)$ to $x$, such that

$$flow(p \circ (\tilde{\sigma} \star \tilde{\alpha}_{1:0}), \Omega) > cap(p \circ \tilde{\sigma})$$
$$= \|p \circ \tilde{\sigma}\| - width(F)/2 - width(\omega)/2,$$

where $F$ is the fringe of $S$ containing $p(U)$.

The quantity $f = flow(p \circ (\tilde{\sigma} \star \tilde{\alpha}_{1:0}), \Omega)$ depends only on the fringe containing $\tilde{\sigma}(0)$, because if this and $\tilde{\omega}$ are held fixed, all the links $p \circ (\tilde{\sigma} \star \tilde{\alpha}_{1:0})$ are link-homotopic. Hence the barrier for $\tilde{\omega}$ growing from $U$ is the set of points in $M - Bd M$ visible from $U$ whose distance from $U$ (in the wiring norm) is less than $f + width(F)/2 + width(\omega)/2$, proving the claim.

Figure 5a.2. The shape of a barrier. We break up the forbidden zones into barriers, which are lifts of polygonal regions like the shaded set $Q$ shown here. This region is the set of points interior to the sheet that are visible from the striped set $V$, and lie within a certain distance of it.

Now put $V = p(U)$, and let $Q$ denote the set of points in $S - Bd S$ visible from $V$ whose distance from $p(U)$ is less than $f + width(F)/2 + width(\omega)/2$. Because the wiring norm is piecewise linear, and the fringes are polygons, $Q$ is bounded by line segments. Because $V$ is connected and open in $Bd S$, the set $Q$ is connected and open; because $V$ is smaller than any fringe of $S$, the closure of $Q$ has no "holes". Hence $Q$ is the inside of a polygon in $S$, and $Cl Q$ is simply connected. By Lemma 2b.8, $p^{-1}(Cl Q)$ consists of disjoint copies of $Cl Q$. One of these, call it $P'$, contains $U$. Put $P = P' \cap p^{-1}(Q)$. One easily checks that $P$ is the barrier for $\tilde{\omega}$ growing from $U$, proving (1).
We now prove part (2) by showing that \( \tilde{\omega} \) intersects only finitely many of its barriers. Given any \( x \) in \( M \), choose a polygonal neighborhood \( O \) of \( p(x) \) in \( S \), and define the neighborhood \( N_x \) of \( x \) to be the component of \( p^{-1}(O) \) that contains \( x \). For each set \( Q \) described above, the intersection \( O \cap Q \) has finitely many components, and hence \( N_x \cap p^{-1}(Q) \) has finitely many components. Therefore \( N_x \) intersects only finitely many components of \( p^{-1}(Q) \). Every barrier for \( \tilde{\omega} \) is a component of \( p^{-1}(Q) \) for some set \( Q \) described above, and the number of such sets \( Q \) is finite. Hence \( N_x \) intersects only finitely many barriers of \( \tilde{\omega} \). Because \( \text{Im } \tilde{\omega} \) is compact, it can be covered by finitely many neighborhoods \( N_x \), and hence intersects only finitely many barriers, which proves (2). Let \( \{P_i\} \) be the set of right-hand barriers of \( \tilde{\omega} \) that intersect \( \tilde{\omega} \).

To show (3) we prove the inclusions \( Z - R \subseteq \bigcup P_i \subseteq Z \). Certainly every barrier \( P_i \) is a subset of \( Z \). On the other hand, if \( x \) is a point in \( Z - R \), then \( x \) is in some right-hand barrier \( P \) for \( \tilde{\omega} \); we have \( x = \tilde{\alpha}(1) \) where \( \tilde{\alpha}(0) \in R \) and \( \tilde{\alpha} \) is a forbidden half-link for \( \tilde{\omega} \). This half-link must intersect \( \tilde{\omega} \); say \( \tilde{\alpha}(s) \in \text{Im } \tilde{\omega} \) where \( s > 0 \). Then \( \tilde{\alpha}_s \) is a forbidden half-link for \( \tilde{\omega} \), so \( \alpha(s) \in P \). Therefore \( P = P_i \) for some \( i \) because \( P \) intersects \( \tilde{\omega} \). This proves part (3).

So every barrier in \( M \) is a connected component of the inverse image (under the covering map) of an open set in \( S \). Hence barriers are open. It follows that forbidden zones, which are unions of barriers, are themselves open.

**Detours**

To construct a link that avoids forbidden zones, we start with any link and repeatedly insert detours around barriers. Lemma 5a.4 shows that the number of barriers we need to consider is finite. This proof technique is not the most elegant, but it works. The following definition aids in describing the process of inserting detours.

**Definition 5a.5.** Let \( \alpha \) be a simple link in a blanket \( M \), and let \( P \) be an open subset of \( M \). A *detour of \( \alpha \) around \( P \)* is a simple link \( \alpha' \in [\alpha]_L \) such that

\[
\text{Im } \alpha' \subseteq (\text{Im } \alpha \cup \text{Fr } P) - P.
\]

The detour \( \alpha' \) is **leftward** if \( P \subseteq \text{right}(\alpha') \) and \( \alpha' \) does not intersect \( \text{right}(\alpha) \).

The next lemma takes care of the induction step by showing that two detours can be combined into one. Its use of the Detour Lemma gives that result its name.

**Lemma 5a.6.** Let \( \beta \) and \( \beta' \) be leftward detours of a simple link \( \alpha \) around the open sets \( P \) and \( P' \). Then there is a leftward detour of \( \alpha \) around \( P \cup P' \).

**Proof.** We apply the Detour Lemma (3c.3) to the links \( \beta \) and \( \beta' \), obtaining a simple link \( \gamma \) that is link-homotopic to both \( \beta \) and \( \beta' \), and hence to \( \alpha \). Part of that lemma
states that the right-hand scrap of $\gamma$ contains those of $\beta$ and $\beta'$, so both $P$ and $P'$ lie to the right of $\gamma$, as does the right-hand scrap of $\alpha$. Hence if $\gamma$ is a detour of $\alpha$ around $P \cup P'$, it is leftward. The other claim of Lemma 3c.3 is that $\text{Im} \gamma$ lies within $\text{Im} \beta \cup \text{Im} \beta'$. Therefore

$$\text{Im} \gamma \subseteq \text{Im} \beta \cup \text{Im} \beta'$$

$$\subseteq (\text{Im} \alpha \cup \text{Fr} P) \cup (\text{Im} \alpha \cup \text{Fr} P')$$

$$= \text{Im} \alpha \cup (\text{Fr} P \cup \text{Fr} P').$$

An elementary topological calculation shows that $\text{Fr}(P \cup P') = (\text{Fr} P \cup \text{Fr} P') - (P \cup P')$. Since $\text{Im} \gamma$ does not intersect $P$ or $P'$, it follows that

$$\text{Im} \gamma \subseteq (\text{Im} \alpha \cup \text{Fr}(P \cup P')) - (P \cup P').$$

Therefore $\gamma$ is a detour of $\alpha$ around $P \cup P'$. □

Now we consider the basis case: making a single detour around a barrier. Because barriers are polygons, this is manageable, but somewhat tedious.

**Lemma 5a.7.** Let $\widetilde{\omega}$ be a lift of a wire in a safe design $\Omega$. Then for every right-hand barrier $P$ of $\tilde{\omega}$, there is a leftward detour of $\tilde{\omega}$ around $P$.

**Proof.** Let $L$ and $R$ denote $\text{left}(\tilde{\omega})$ and $\text{right}(\tilde{\omega})$; let $Z$ be the right-hand forbidden zone of $\tilde{\omega}$. The set $P$ is homeomorphic to the inside of a simple polygon by Lemma 5a.4, and we may assume it intersects $\text{Im} \tilde{\omega}$. To construct the detour, we replace the parts of $\tilde{\omega}$ that pass through $P$ by a path along $\text{Fr} P$. Put $s = \inf \tilde{\omega}^{-1}(\text{Cl} P)$ and $t = \sup \tilde{\omega}^{-1}(P)$. Now let $\tau$ be a simple sublink within $P$ from $\tilde{\omega}(s)$ to $\tilde{\omega}(t)$, and consider the link

$$\alpha = \tilde{\omega}_{0:s} \ast \tau \ast \tilde{\omega}_{t:1}.$$

It separates the polygon $\text{Fr} P$ into two components; let $\tau'$ be the path in $\text{Fr} P$ from $\tilde{\omega}(s)$ to $\tilde{\omega}(t)$ that lies to the left of $\alpha$, and hence keeps $P$ to the right. Define a path $\beta$ by

$$\beta = \tilde{\omega}_{0:s} \ast \tau' \ast \tilde{\omega}_{t:1}.$$
The path $\beta$ is piecewise linear, and lies in $\text{Im} \, \bar{\omega} \cup \text{Fr} \, P$. We show that $\beta$ intersects no fringe $F$ other than the terminals of $\bar{\omega}$. If $F$ lies completely in $L$, then all points close enough to $F$ are in the left-hand forbidden zone of $\bar{\omega}$, which $P$ cannot touch by Lemma 5a.3. Furthermore, $\beta$ cannot intersect any fringe that lies completely in $R$, because $\beta$ lies to the left of $\alpha$, while the fringes of $R$ lie to the right of $\alpha$. (The link $\alpha$ is link-homotopic to $\bar{\omega}$, and hence by Proposition 3c.4, separates the fringes of $M$ as $\bar{\omega}$ does.) Therefore $\beta$ runs between the terminals of $\bar{\omega}$. Its middle may intersect these terminals, however.

We now convert $\beta$ into a simple link. Say that $\beta$ runs from fringe $X$ to fringe $Y$. Set $s = \sup \beta^{-1}(X)$ and $t = \inf((s, 1) \cap \beta^{-1}(Y))$. Then $\gamma = \beta_{st}$ is a simple link between $X$ and $Y$, and is piecewise linear; its image also lies in $\text{Im} \, \bar{\omega} \cup \text{Fr} \, P$. Then $\gamma$ is a detour of $\bar{\omega}$ around $P$: it is link-homotopic to $\bar{\omega}$ because it runs between the same terminals, and its image lies in $(\text{Im} \, \bar{\omega} \cup \text{Fr} \, P) - P$. By the construction of $\beta$, there are some points of $P$ that lie in $\text{right}(\gamma)$. Hence $P$, being connected, must lie entirely in $\text{right}(\gamma)$.

Still, $\gamma$ may not be a leftward detour, because it may enter the right-hand scrap of $\bar{\omega}$. But the Detour Lemma (3c.3) applied to $\bar{\omega}$ and $\gamma$ solves this problem. □

All that remains is to put the pieces together.

**Proposition 5a.8.** Every wire in a safe design has an evasive route.

**Proof.** Let $\Omega$ be a safe design on a sheet $S$, and let $\omega$ be a wire in $\Omega$. Let $M$ be a blanket of $S$ with covering map $p: M \to S$. Lift $\omega$ to a simple link $\hat{\omega}$ in $M$. We construct an evasive route $\delta$ by finding detours of $\bar{\omega}$ around its forbidden zones.

Let $L$ and $R$ be the left and right scraps of $\bar{\omega}$, and let $Z$ be the right-hand forbidden zone of $\bar{\omega}$. Apply Lemma 5a.4 to find barriers $P_i$ such that $Z - R \subseteq \bigcup P_i$. For each $i$, apply Lemma 5a.7 to obtain a leftward detour of $\bar{\omega}$ around $P_i$. Repeated application of Lemma 5a.6 then gives us a leftward detour of $\bar{\omega}$ around $\bigcup P_i$. Call it $\delta$. Because $\delta$ does not enter $R$, and avoids $Z - R$, it must avoid $Z$ entirely. Now let $Z'$ be the right-hand forbidden zone of $\hat{\delta}$, which is the left-hand forbidden zone of $\bar{\omega}$. Decompose $Z - L$ into barriers $P'_i$, and apply the same technique to find a leftward detour $\eta$ of $\hat{\delta}$ around $Z'$. By construction, $\eta$ avoids $Z'$, and $\text{Im} \, \eta \subseteq \text{Im} \, \hat{\delta} \cup \text{Fr} \, Z'$.

I claim that $\eta$ avoids $Z$ as well as $Z'$. Let $x$ be a point of $\text{Im} \, \eta$. If $x \in \text{Im} \, \hat{\delta}$, then $x \in \text{Im} \, \delta$, and hence $x \notin Z$. So assume that $x \in \text{Fr} \, Z'$. If $x$ were in $Z$, then because $Z$ is open (Lemma 5a.4), $Z$ and $Z'$ would intersect, contradicting Lemma 5a.3. Therefore $\eta$ is a simple link in $[\bar{\omega}]_L$ that avoids the forbidden zones of $\bar{\omega}$. By Lemma 5a.2, the route $p \circ \eta$ of $\omega$ is evasive in $\Omega$. □

**Ideal routes**

Building on Proposition 5a.8, we now complete the construction of ideal routes for wires in a safe design. All that remains is to show that among the evasive routes
of a wire there is one of minimum length. The Reparameterization Lemma (3d.1) allows us to make this route canonical.

**Proposition 5a.9.** Every wire in a safe design has an ideal route.

**Proof.** Let $\Omega$ be a safe design on a sheet $S$; let $\omega$ be a wire in $\Omega$ with lifting $\tilde{\omega}$. Let $M$ be the blanket for $S$, and denote by $Z$ the union of the forbidden zones of $\tilde{\omega}$. Let $X$ and $Y$ denote the fringes of $M$ containing $\tilde{\omega}(0)$ and $\tilde{\omega}(1)$, respectively. We consider the family $\Lambda$ of canonical, evasive routes of $\omega$. Since reparameterizing a path does not affect its evasiveness or its arc length (see Lemma 3d.1), Proposition 5a.8 shows that $\Lambda$ is nonempty, and $l = \inf\{ |\lambda| : \lambda \in \Lambda \}$. By Proposition 2c.8, the collection $\Lambda$ contains a uniformly convergent sequence $(\alpha_k)_{k=1}^{\infty}$ whose limit $\alpha$ has euclidean arc length at most $l$.

I claim that $\alpha$ has a lifting $\tilde{\alpha} : X \leadsto Y$ that avoids $Z$. Let $\beta$ be any lift of $\alpha$ to $M$. By Lemma 3a.7, the paths $\alpha_k$ have lifts $\beta_k$ that converge uniformly to $\beta$. In particular, there is a constant $K$ such that the paths $\{ \beta_k \}_{k>K}$ run between the same fringes of $M$. Let $h$ be a covering transformation that carries those fringes onto the fringes of $\tilde{\omega}$. (One must exist, since the paths $\alpha_k$ have liftings in $[\tilde{\omega}]_L$.) Then $h \circ \beta_k$ is a path in $M - Z$ for each $k > K$, and since forbidden zones are open, by Lemma 5a.4, the limit $h \circ \beta$ of the sequence $(h \circ \beta_k)$ also avoids $Z$. Write $h \circ \beta$ as $\tilde{\alpha}$. The fringes of $M$ are closed, so the endpoints of $\beta$ lie on the terminals of $\beta_k$, and therefore $\tilde{\alpha}$ has the same terminals as $\tilde{\omega}$.

Now we convert $\alpha$ into a canonical route of $\omega$. Put $s = \sup \tilde{\alpha}^{-1}(X)$ and $t = \inf((s,1] \cap \tilde{\alpha}^{-1}(Y))$. Then $\tilde{\alpha}_{st}$ intersects $X$ and $Y$ at its endpoints alone. It cannot intersect any other fringe of $M$, for it would have to cross $Z$ to do so. Therefore $\tilde{\alpha}_{st}$ is a link, and $\alpha_{st}$ is an evasive route of $\omega$. Using Lemma 3d.1, let $\tilde{\gamma}$ be a canonical version of $\tilde{\alpha}_{st}$; it has the same image, arc length, and path class. One can check that $\tilde{\gamma}^{-1}(X) = \{0\}$ and $\tilde{\gamma}^{-1}(Y) = \{1\}$, which makes $\tilde{\gamma}$ a link. Hence its projection $\gamma$ is a canonical, evasive route of $\omega$; in symbols, $\gamma \in \Lambda$. We have $|\gamma| = |\tilde{\alpha}_{st}| \leq |\alpha| \leq l$, and hence $|\gamma| = l$. Therefore $\gamma$ has minimum length among all evasive routes of $\omega$. In other words, $\gamma$ is an ideal route of $\omega$. \qed

## 5B. Ideal Routes Are Taut

Now we begin the process of characterizing ideal routes. Top priority is to show that liftings of ideal routes are simple, so that our results about simple links will apply to them. At the same time we prove that an ideal route has vertices only where it bends around its barriers. Then we prove a key technical lemma: every straight half-cut for an ideal route is either trivial or semisimple. That fact enables us to prove that ideal routes are taut: wherever one turns, it is supported by a
straight, marginal, nondegenerate half-cut. Later sections will use these half-cuts to demonstrate that other ideal routes, which are evasive, cannot approach this one. And as we show in Section 6B, tautness implies that the ideal route cannot be made any shorter without becoming infeasible.

Getting off the ground

One easy result is that lifts of ideal routes are injective. For if $\beta$ lifts an ideal route $\alpha$ of $\omega$ and $\beta(s) = \beta(t)$, where $s \leq t$, then $\alpha_{0:s} \ast \alpha_{t:1}$ is an evasive route of $\omega$, and its arc length is $1 - t + s$ times that of $\alpha$ because $\alpha$ is canonical. Since $\alpha$ has minimum length among the evasive routes of $\omega$, and $|\alpha| > 0$, it follows that $s = t$. The first task is to prove that these liftings are piecewise linear, and therefore simple. We start with a technical lemma.

Lemma 5B.1. Let $\alpha$ be a link in a blanket. There are simple links $\beta, \gamma \in [\alpha]_L$ such that $\text{Im} \alpha \in \text{left}(\beta) \cap \text{right}(\gamma)$.

Proof. For each point $t \in I$, let $\alpha_t \in [\alpha]_L$ be a simple link that passes through $\alpha(t)$. By modifying $\alpha_t$ in the neighborhood of $\alpha(t)$, find links $\beta_t$ and $\gamma_t$ in $[\alpha]_L$ such that $\alpha(t) \in \text{left}(\beta_t) \cap \text{right}(\gamma_t)$. Write $L_t = \text{left}(\beta_t)$ and set $U_t = \alpha^{-1}(L_t)$; similarly, put $R_t = \text{right}(\gamma_t)$ and $V_t = \alpha^{-1}(R_t)$. Since $t \in U_t \cap V_t$ and the sets $U_t$ and $V_t$ are open, the collection $\{U_t \cap V_t\}$ is an open cover of $I$. Because $I$ is compact, it has a finite subcover

$$U_{i_1} \cap V_{i_1}, \ldots, U_{i_n} \cap V_{i_n}.$$  

Then $\text{Im} \alpha \subset \bigcup_{i=1}^n L_{i_t}$, and also $\text{Im} \alpha \subset \bigcup_{i=1}^n R_{i_t}$. By iterative application of the Detour Lemma (3c.3), there is a simple link $\gamma \in [\alpha]_L$ whose right-hand scrap contains those of $\gamma_{i_1}, \gamma_{i_2}, \ldots, \gamma_{i_n}$. A symmetrical argument yields a simple link $\beta \in [\alpha]_L$ whose left-hand scrap contains those of $\beta_{i_1}, \beta_{i_2}, \ldots, \beta_{i_n}$. It follows that $\text{Im} \alpha$ lies left of $\beta$ and right of $\gamma$. \hfill \Box

Turning points of ideal routes

The proof that shows ideal routes are piecewise linear, which we are about to begin, also characterizes the points at which they turn. Suppose that $\alpha$ is a simple link in a blanket $M$, and that $\alpha$ turns at $x \in (0, 1)$. Let $A$ and $B$ be the scraps of $M - \text{Im} \alpha$. One of these scraps, say $A$, contains points internal to the angle made by $\alpha$ at $x$. Let $C$ be a subset of $M$. If $C$ intersects $A$ but not $B$, then $\alpha$ turns toward $C$ at $x$; if $C$ intersects $B$ but not $A$, then $\alpha$ turns away from $C$ at $x$.

We prove the very intuitive fact that at each joint of an ideal route, it bends around the vertex of a barrier, and hence turns toward that barrier. The idea behind the proof is that where an ideal route is not constrained by the vertex of a barrier, it is elastic and hence linear. We have worked through a similar proof before: see Lemma 3d.5.

– 161 –
**Lemma 5b.2.** Let $\alpha$ be an ideal route with lift $\tilde{\alpha}$. The link $\tilde{\alpha}$ is simple, and if $\alpha$ is not straight at $x \in (0, 1)$, then some barrier $P$ for $\tilde{\alpha}$ has a vertex at $\tilde{\alpha}(x)$, and $\tilde{\alpha}$ turns toward $P$ at $x$.

**Proof.** Let $\Omega$ be a safe design on a sheet $S$, and let $\alpha$ be an ideal route of a wire $\omega \in \Omega$. Let $M$ be a blanket of $S$ with covering map $p: M \to S$. Let $Z$ be the union of the forbidden zones for $\tilde{\alpha}$. Because $\alpha$ is an evasive route of $\omega$, the link $\tilde{\alpha}$ avoids $Z$. By Lemma 5b.1, there are links $\beta$ and $\gamma$ in $M$ such that $\text{Im} \alpha \in \text{left}(\beta) \cap \text{right}(\gamma)$.

Suppose that $\tilde{\alpha}$ is not straight at the point $x \in (0, 1)$. Let $U$ be a neighborhood of $\tilde{\alpha}(x)$ that intersects neither $\beta$ nor $\gamma$. By Lemma 5a.4, the set $U \cap Z$ is the intersection of $U$ with finitely many barriers $P_1, \ldots, P_n$ of $\beta$ and $\gamma$. Also by Lemma 5a.4, these barriers are polygonal. Hence by restricting $U$ we may assume that $U$ contains no vertex of a barrier $P_i$ except those lying at $\tilde{\alpha}(x)$. Then the situation is as shown in Figure 5b-1. We may assume that $U \cap \text{Bd} M$ is empty and that $p(U)$ is convex. Because $\tilde{\alpha}$ is continuous and evasive, there is an interval $(s, t)$ containing $x$ such that $\tilde{\alpha}[s, t] \subset U - Z$.

![Figure 5b-1. Where an ideal link turns.](image)

Within a neighborhood small enough to include only one barrier vertex, a lifting $\tilde{\alpha}$ of an ideal route is either straight, or else it turns toward that barrier as shown here.

The straight path $\tilde{\kappa}$ between $\tilde{\alpha}(s)$ and $\tilde{\alpha}(t)$ must intersect $Z$. If not, let $\tilde{\alpha}'$ be the result of replacing the subpath $\tilde{\alpha}_{st}$ of $\tilde{\alpha}$ with $\kappa$. Then $\tilde{\alpha}'$ avoids $Z$, and hence $\alpha' = p \circ \tilde{\alpha}'$ is evasive. Furthermore, $\alpha' \approx_L \alpha$ because their lifts are link-homotopic. If $\alpha' \neq \alpha$, then $\alpha'$ is shorter, because a linear path is the only shortest canonical path between two points. (Compare Lemma 3d.3.) But $\alpha$ has the minimum length among all evasive routes of $\omega$. Therefore $\alpha = \alpha'$, so $\alpha$ is straight at $x$, contrary to assumption. Thus $\tilde{\kappa}$ must intersect some barrier $P_i$, and this barrier must have a vertex at $x$. Hence $\tilde{\alpha}$ is straight everywhere except the vertices of the barriers $P_i$. Since these vertices are finite in number, and $\tilde{\alpha}$ has finite arc length, we conclude that $\tilde{\alpha}$ is piecewise linear. In fact, because $\tilde{\alpha}$ is nonconstant and canonical, it is piecewise straight.

It remains to show that $\tilde{\alpha}$ turns toward $P_i$ at $x$. Let $\tilde{\sigma}$ be the linear path from $\tilde{\alpha}(s)$ to $\tilde{\alpha}(x)$, and let $\tilde{\tau}$ be the linear path from $\tilde{\alpha}(x)$ to $\tilde{\alpha}(t)$. Then $\tilde{\sigma}$ and $\tilde{\tau}$ do not intersect $Z$, so by the argument above, we must have $\tilde{\alpha}_{sx} = \tilde{\sigma}$ and $\tilde{\alpha}_{xt} = \tilde{\tau}$. We conclude that $\tilde{\alpha}$ turns at $x$. The barrier $P_i$ has points internal to the angle of $\tilde{\alpha}$.
at $x$, and does not intersect $\tilde{\alpha}$ because $\alpha$ is evasive. Therefore $\tilde{\alpha}$ turns toward $P_i$ at $x$. □

We can extend the notion of turning to the endpoints of a link. Let $\alpha$ be a simple link in a sheet $S$. We say that $\alpha$ turns at $e \in \{0,1\}$ if a straight subpath $\alpha_{se}$ of $\alpha$ makes an acute angle with an edge of the fringe containing $\alpha(e)$. A link in a blanket turns wherever its projection does. Let $\alpha$ be a simple link in a blanket $M$, and let $e \in \{0,1\}$ be a point at which its projection turns. Then we say $\alpha$ turns at $e$. Let $A$ and $B$ denote the scraps of $M - \text{Im} \alpha$, and suppose $A$ contains points internal to an acute angle made by $\alpha$ at $e$. We say that $\alpha$ turns toward a set $C \subset M$ at $e$ if $C$ intersects $A$ but not $B$. Then we obtain the following extension of Lemma 5b.2.

Lemma 5b.3. Let $\alpha$ be an ideal route with lift $\tilde{\alpha}$. If $\alpha$ turns at $e \in \{0,1\}$, then some barrier $P$ for $\tilde{\alpha}$ has a vertex at $\tilde{\alpha}(e)$, and $\tilde{\alpha}$ turns toward $P$ at $e$.

Proof. Assume without loss of generality that $e = 0$. Let $\nu$ be a straight path in the terminal containing $\alpha(0)$ such that $\nu \ast \alpha$ has an acute angle at $\alpha(0)$. Let $E$ be the fringe containing $\tilde{\alpha}(0)$, and let $\tilde{\nu}$ be a lift of $\nu$ satisfying $\tilde{\nu}(1) = \tilde{\alpha}(0)$. Then $\tilde{\alpha}$ turns toward $\tilde{\nu}(0)$ at 0. Let $A$ be the scrap of $M - \text{Im} \tilde{\alpha}$ that contains $\tilde{\nu}(0)$, and let $Z \subseteq A$ be the forbidden zone for $\tilde{\alpha}$ on the same side as $\tilde{\nu}(0)$.

Some barrier $P \subseteq Z$ for $\tilde{\alpha}$ has a vertex at $\tilde{\alpha}(0)$. For if not, then because no edge of a barrier in $Z$ can contain $\tilde{\alpha}(0)$, there is a neighborhood $U$ of $\tilde{\alpha}(0)$ that does not intersect $Z$. Let $\tilde{\sigma}$ be a straight path in this neighborhood from $E$ to $\alpha(s)$ that intersects $E$ perpendicularly. Then $\tilde{\sigma}$ is shorter than $\tilde{\alpha}_{0:1}$, and $\tilde{\sigma} \ast \tilde{\alpha}_{0:1}$ avoids the forbidden zones of $\tilde{\alpha}$. Projecting to the sheet, one thus obtains an evasive route of $\alpha$ that is shorter than $\alpha$, contradicting the assumption that $\alpha$ is ideal. □

Straight half-cuts for ideal routes

Now that we know ideal routes are piecewise straight, we can begin to apply our tools to them. One major property of a route that is ideal for a design $\Omega$ is that its nontrivial straight half-cuts respect $\Omega$. Actually, they have an even stronger property: they are semisimple in $\Omega$, which by Proposition 4e.6 implies that they are nondegenerate in $\Omega$ and respect $\Omega$ strongly.

Proposition 5b.4. If $\omega$ is an ideal route, then every straight half-cut for $\rho$ is either trivial or semisimple.

Outline of proof. Let $\sigma$ be a nontrivial straight half-cut for $\omega$ at $s$. Let $S$ be the relevant sheet, and let $\gamma$ be a straight cut of $S$ such that $\sigma = \gamma_{0:a}$ for some $a \in (0,1)$. Lift $\omega$ to a simple link $\tilde{\omega}$, and lift $\gamma$ to a straight link $\tilde{\gamma}$ such that $\tilde{\gamma}(a) = \tilde{\omega}(s)$. Let $(b,t)$ be the crossing of $\tilde{\gamma}$ by $\tilde{\omega}$ that minimizes $b$. The half-cut $\gamma_{0:b}$ for $\omega$ at $t$ is akin
to the half-cut \( \gamma_{0:a} \) for \( \omega \) at \( s \), which is \( \sigma \). Hence it suffices to prove the lemma in the case \( (b, t) = (a, s) \). We find a simple cut \( \chi \) and a necessary crossing \( (c, s) \) of \( \chi \) by \( \omega \) such that \( \sigma = \chi_{0:c} \). There are five cases to consider, of which four are easy and the fifth requires a further case analysis.

**Figure 5b-2.** Straight half-cuts for an ideal route. Given a nontrivial straight half-cut \( \sigma \) for an ideal route \( \omega \), we extend it to a straight cut \( \gamma \), and lift both the cut and the route to the blanket. The liftings are denoted \( \tilde{\gamma} \) and \( \tilde{\omega} \). In each of four cases, here labeled (1) through (4), we construct a link (dashed) starting at \( \tilde{\gamma}(0) \) that cuts \( \tilde{\omega} \). The projection of this bent link is a simple cut that makes \( \sigma \) semisimple. One case is missing here; it is like case (4) but \( \tilde{\omega} \) does not turn toward \( \tilde{\gamma}(1) \). For this case, see Figure 5b-3.

(1) The link \( \tilde{\omega} \) does not cross over \( \tilde{\gamma} \) at \( s \). Then \( \tilde{\omega} \) turns away from \( \tilde{\gamma}(0) \) at \( s \). Let \( P \) be a barrier for \( \tilde{\omega} \) having a vertex at \( \tilde{\omega}(s) \), such that \( \tilde{\omega} \) turns toward \( P \) at \( s \). Choose a bent half-link \( \theta \) in \( \text{Cl} \ P \) from the base of \( P \) to \( \tilde{\omega}(s) \). Then the link \( \alpha = \tilde{\gamma}_{0:a} * \theta_{1:0} \) is simple, and its endpoints lie on opposite sides of \( \tilde{\omega} \). Since \( \sigma \) and the projection of \( \theta \) are nontrivial half-cuts, neither \( \tilde{\gamma}(0) \) nor \( \theta(0) \) lies on a terminal of \( \tilde{\omega} \). Hence \( \alpha \) actually cuts \( \tilde{\omega} \). So \( \omega \) necessarily crosses \( p \circ \alpha \) at \( s \). We set \( \chi = p \circ \alpha \) and \( c = \frac{1}{2} \).

(2) The link \( \tilde{\omega} \) crosses over \( \tilde{\gamma} \) at \( s \), and crosses back at some point \( \tilde{\gamma}(b) = \tilde{\omega}(t) \), where \( b > a \). Assume \( (b, t) \) is chosen to minimize \( b - a \). Since the path \( \tilde{\gamma}_{a:b} \) is shorter than \( \tilde{\omega}_{s:t} \), it must intersect a forbidden zone of \( \tilde{\omega} \). Choose a half-link \( \theta \) in this forbidden zone such that \( \theta(1) = \tilde{\gamma}(e) \) for some \( e \in (a, b) \). Then \( \tilde{\gamma}(0) \) and \( \theta(1) \) lie on opposite sides of \( \tilde{\omega} \), and neither lies on a terminal of \( \tilde{\omega} \). Hence \( \alpha = \tilde{\gamma}_{0:e} * \theta_{1:0} \) cuts \( \tilde{\omega} \), and so \( \omega \) makes a necessary crossing with \( \alpha \) at \( \tilde{\omega}(s) \). Because \( \alpha \) is bent and its segments are not parallel, \( p \circ \alpha \) is simple. We put \( \chi = p \circ \alpha \) and define \( c \) by \( \alpha(c) = \tilde{\omega}(s) \).

It remains to consider situations in which \( \tilde{\omega} \) crosses over \( \tilde{\gamma} \) at \( s \) and does not cross back. Then \( \tilde{\gamma}(0) \) and \( \tilde{\gamma}(1) \) lie on opposite sides of \( \tilde{\omega} \), and \( \tilde{\gamma} \) either cuts or shares a
terminal with $\tilde{\omega}$. Since $\sigma$ is not trivial, the shared terminal cannot contain $\tilde{\gamma}(0)$. Thus we have the following cases.

(3) The link $\tilde{\omega}$ cuts $\tilde{\omega}$. Then the crossing $(a, s)$ of $\gamma$ by $\omega$ is necessary, and we simply put $\chi = \gamma$ and $c = a$.

(4) The link $\tilde{\omega}$ crosses over $\tilde{\gamma}$ at $s$, and $\text{cross}(\tilde{\gamma}, \tilde{\omega}) = 1$; for some $i \in \{0, 1\}$, the point $\tilde{\gamma}(1)$ lies on the fringe containing $\tilde{\omega}(i)$; and $\tilde{\omega}$ turns toward $\tilde{\gamma}(1)$ at a point in $(i, s)$. Then a forbidden zone of $\tilde{\omega}$ intersects the inside of $\text{Im}(\tilde{\omega}_i,s \ast \tilde{\gamma}_{1,a})$, which is a web of one thread. Because there are no fringes in this area, and $\tilde{\omega}$ avoids its forbidden zones, this zone must intersect $\tilde{\gamma}_{1,a}$. We construct $\chi$ as in case (2).

Figure 5b-3. *Straight half-cuts, continued.* The difficult case in Figure 5b-2 occurs when $\tilde{\gamma}$ and $\tilde{\omega}$ share a terminal $T$, and $\tilde{\omega}$ does not turn toward $\tilde{\gamma}(1)$. To handle this we replace $\tilde{\gamma}$ by a bent link $\tilde{\gamma}'$ that avoids $T$. Essentially the same cases arise, but $\tilde{\gamma}'$ and $\tilde{\omega}$ cannot share a second terminal $T'$ without falling into case (4').

The remaining case is the messy one. We can assume that $\tilde{\omega}$ crosses over $\tilde{\gamma}$ at $s$, that $(a, s)$ is the only crossing of $\tilde{\gamma}$ by $\tilde{\omega}$, that $\tilde{\omega}(i)$ shares a fringe with $\tilde{\gamma}(1)$, and that $\tilde{\omega}$ does not turn toward $\tilde{\gamma}(1)$ at any point in $(i, s)$. The situation is shown in Figure 5b-3. Let $F$ be the fringe containing $\omega(i)$, and let $\lambda$ be a line tangent to $F$ at $\gamma(1)$. Being convex, $F$ lies on the opposite side of $\lambda$ from $\gamma(0)$. Let $\tau$ be the half-cut of $\omega$ at $s$ such that $\tau$ is parallel to $\lambda$ and $\tau(0)$ lies on the opposite side of $\gamma$ from $\omega(1)$. Then $\tau$ does not intersect the fringe containing $\gamma(1)$. In addition, $\sigma \ast \tilde{\tau}$ crosses over $\omega$ at $\omega(s)$, because otherwise $\omega_i,s$ would have to turn toward $\gamma(1)$.

We perform another case analysis like that above, but with $\sigma \ast \tilde{\tau}$ in place of $\gamma$. The details are omitted. Because $\omega$ crosses over $\sigma \ast \tilde{\tau}$ at $s$, there is no case corresponding to case (1). The remaining cases correspond to (2), (3), and (4). No further problems arise. For if $\omega$ shares one terminal with $\gamma(1)$ and the other with $\tau(0)$, geometry dictates that it must turn toward them somewhere. □
Struts for ideal routes

The following definition and proposition are central to the analysis of ideal routes. We show that wherever an ideal route turns, it has a rigid cut or half-cut toward which the route turns. If a cut or half-cut \( \theta \) is nondegenerate and straight, and \( \text{margin}(\theta, \Omega) = 0 \), then we say \( \theta \) is rigid in \( \Omega \).

**Definition 5b.5.** Let \( \Omega \) be a design on a sheet \( S \), and let \( \omega \) route a wire in \( \Omega \). A strut for \( \omega \) at \( t \) is a rigid cut or half-cut \( \sigma \) for \( \omega \) at \( t \) with the following property: if \( \tilde{\sigma} \) and \( \tilde{\omega} \) are lifts of \( \sigma \) and \( \omega \) satisfying \( \tilde{\sigma}(1) = \tilde{\omega}(t) \), then \( \tilde{\omega} \) turns toward \( \tilde{\sigma}(0) \) at \( t \). The link \( \omega \) is taut if there is a strut for \( \omega \) at every joint of \( \omega \).

The proof that ideal routes are taut is fairly intuitive. Lemmas 5b.2 and 5b.3 say that ideal routes only turn at the vertices of barriers. And since points inside barriers correspond to nondegenerate, unsafe half-cuts, it stands to reason that the vertices of barriers correspond to nondegenerate, rigid cuts and half-cuts. Using the results on chains from Section 4F, we construct a rigid half-cut for each joint of an ideal route. The lifting of this half-cut, moreover, lies in the closure of the barrier that constrains that joint. Since ideal routes turn toward the barriers that constrain them, the half-cut turns out to be a strut.

**Proposition 5b.6.** Ideal routes are taut.

**Proof.** Let \( \Omega \) be a safe design on a sheet \( S \), and let \( \omega \) be an ideal route of a wire in \( \Omega \). Denote by \( M \) the blanket of \( S \), and let \( p: M \rightarrow S \) be the covering map. Lift \( \omega \) to a simple link \( \tilde{\omega} \) in \( M \). Suppose \( \omega \) turns at \( t \in [0,1] \). Then by Lemma 5b.2 there is a barrier \( P \) for \( \tilde{\omega} \) with a vertex at \( \tilde{\omega}(t) \) such that \( \tilde{\omega} \) turns toward the base of \( P \) at \( t \). Let \( B \) denote the base of \( P \), and as in Lemma 5a.4, let \( f \) denote the common flow across the half-cuts whose lifts are the forbidden half-links that define \( P \). By the geometry of barriers (see Lemma 5a.4), there is a straight path \( \tilde{\alpha} \) in \( \tilde{Cl} P \) from \( B \) to \( \tilde{\omega}(t) \) whose length is

\[
\|p \circ \tilde{\alpha}\| = f + \text{width}(p(B))/2 + \text{width}(\omega)/2. \tag{5-1}
\]

There is also a bent path \( \tilde{\sigma} \) in \( \tilde{Cl} P \) from \( \tilde{\alpha}(0) \) to \( \tilde{\omega}(t) \) whose middle lies in \( P \subset M - \text{Bd} M \). Put \( \sigma = p \circ \tilde{\sigma} \) and \( \alpha = p \circ \tilde{\alpha} \).

The cases \( t \in (0,1) \) and \( t \in \{0,1\} \) have to be distinguished, but they are essentially the same. If \( t \in (0,1) \), then \( \sigma \) is a bent half-cut for \( \omega \) at \( t \), and \( \alpha \) is the elastic chain for \( \sigma \). Also, \( \sigma \) is akin to a straight half-cut \( \theta \) for \( \omega \) (see Figure 5b-4), and hence by Proposition 5b.4, \( \sigma \) is semisimple (it cannot be trivial). Therefore (Proposition 4e.6) \( \sigma \) is nondegenerate and respects \( \Omega \). Similarly, if \( t \in \{0,1\} \), then \( \sigma \) is a bent cut, and \( \alpha \) is the elastic chain for \( \sigma \). In addition, \( \sigma \) is link-homotopic to an associated cut of a straight half cut \( \theta \) for \( \omega \), and again \( \sigma \) is nondegenerate.
and respects $\Omega$. Also $\sigma$ is nonempty. This follows from the fact that terminals are convex.

In both cases the results of Section 4F apply. Lemma 4f.3 gives us a bound on the capacity of $\alpha$; together with inequality (5-1), it gives

\[
\text{cap}(\alpha) \leq \text{cap}(\sigma) + \|\alpha\| - \|\sigma\|
= \|\alpha\| - \text{width}(p(B))/2 - \text{width}(\omega)/2
= f = \text{flow}(\sigma, \Omega),
\]

and the inequality is strict if $\alpha$ is degenerate. Proposition 4f.1 gives us the inequality $\text{flow}(\alpha, \Omega) \geq \text{flow}(\sigma, \Omega) - \text{gaps}(\alpha)$. If $\alpha_1, \ldots, \alpha_n$ are the links of $\alpha$, then we can conclude

\[
\sum_{i=1}^{n} \text{flow}(\alpha_i, \Omega) \geq \text{flow}(\sigma, \Omega) - \text{gaps}(\alpha). \tag{5-3}
\]

Subtracting this inequality from the equation $\sum_{i=1}^{n} \text{cap}(\alpha_i) = \text{cap}(\alpha) - \text{gaps}(\alpha)$, we get

\[
\sum_{i=1}^{n} \text{margin}(\alpha_i, \Omega) \leq \text{cap}(\alpha) - \text{flow}(\sigma, \Omega) \leq 0. \tag{5-4}
\]

The second inequality follows from (5-2). Lemma 4f.2 ensures that $n \geq 1$.

Now we deduce that the final major link in $\alpha$ is nondegenerate. Each $\alpha_i$ is a nondegenerate straight cut or half-cut for $\omega$. Since $\Omega$ is safe and $\omega$ is evasive, all the $\alpha_i$ have nonnegative margin. Hence inequality (5-4) holds with equality, and therefore none of the inequalities that led up to it can be strict. In particular, $\tilde{\alpha}$ cannot be degenerate; $\alpha_n$ must end at $\omega(t)$. Hence $\alpha_n$ is a nondegenerate straight cut or half-cut $\tau$ for $\omega$ at $t$. By (5-4) again, $\text{margin}(\tau, \Omega) \leq 0$, which implies that $\tau$ is rigid.
To check turning, let $\tilde{\tau}$ be a lift of $\tau$ satisfying $\tilde{\tau}(1) = \tilde{\omega}(t)$. To show that $\tau$ is a strut for $\omega$ at $t$, it remains to show that $\tilde{\omega}$ turns toward $\tilde{\tau}(0)$ at $t$. But this is easy, because $\tilde{\tau}(0)$ lies in $Cl P$. Since $\tilde{\tau}(0)$ is not on $\tilde{\omega}$, it lies on the same side of $\tilde{\omega}$ as $P$. And $\tilde{\omega}$ turns toward $P$ at $t$ by assumption. 

5C. Ideal Routes Form a Design

In this section we see the first fruits of our analysis of ideal routes. We show that the ideal routes of wires in a safe design are actually wires, and that they do not intersect. Hence they actually form an embedding of the safe design, and we call this embedding an ideal design.

Figure 5c-1. When lifts of ideal routes cross over. The links $\alpha$ and $\beta$ lift ideal routes (possibly the same one). The scraps of $\alpha$ are $A$ and $A'$; those of $\beta$ are $B$ and $B'$. Where they cross over, they form a simple loop $\lambda$, and at one of its internal angles, $\alpha$ turns away from the endpoints of $\beta$. Since $\alpha$ is taut, it has a strut $\sigma$ there whose lift $\tilde{\sigma}$ is shown. The half-link $\tilde{\sigma}$ crosses $\beta$, forming a half-link $\tilde{\tau}$ that ends on $\beta$. This half-link turns out to be forbidden to $\beta$.

A single technique is used both to rule out intersections between different routes and to rule out self-intersections. Assuming that two routes have an undesirable crossing, we first construct lifts of those routes that reflect this crossing. Each of these two links has its endpoints on the same side of the other. As shown in Figure 5c-1, one of the links has a joint whose strut has a lifting that crosses over the other link. We show that this strut contains an unsafe, nondegenerate half-cut for the other link. This contradicts the fact that ideal routes are evasive, and shows that the undesirable crossing could not have occurred.

The first step, finding an appropriate turning point, is handled mainly by the following lemma. Two links in a blanket cross over if the image of one contains points in both scraps of the other.
Lemma 5c.1. Let $\alpha$ and $\beta$ be coherent links in a blanket $M$. If $\alpha$ and $\beta$ cross over, then there is some $z \in (0, 1)$ such that, up to renaming of $\alpha$ and $\beta$,
(1) $\alpha$ turns away from $\beta(0)$ at $z$, and
(2) $\beta$ separates $\alpha(z)$ from $\alpha(0)$.

Proof. Let $A$ and $A'$ be the scraps of $M - \text{Im} \, \alpha$. By Lemma 4c.5, both endpoints of $\beta$ lie in one of these scraps, say $A'$. Let $B$ and $B'$ be the scraps of $M - \text{Im} \, \beta$, and assume that $B'$ contains the endpoints of $\alpha$. The links $\alpha$ and $\beta$ are simple because they cohere.

Suppose $\alpha$ and $\beta$ cross over, and choose a maximal interval $(t, t') \subseteq \beta^{-1}(A)$. Define $s$ and $s'$ by the equations $\alpha(s) = \beta(t)$ and $\alpha(s') = \beta(t')$. Then the path $\alpha_{sst'} * \beta_{tt'}$ is a simple loop $\lambda$ in $\text{Cl} \, A \cap \text{Cl} \, B$; the middle of $\alpha_{sst'}$ lies in $B$, and the middle of $\beta_{tt'}$ lies in $A$. (See Figure 5c-1.) Hence the inside of the loop $\lambda$ intersects both $A$ and $B$.

Corollary 3c.7 shows that $\lambda$ must have at least three internal angles of measure less than $\pi$. Two of these angles can lie at $\beta(t)$ or $\beta(t')$, but the third must lie in $\text{Mid} \, \alpha_{sst'}$ or $\text{Mid} \, \beta_{tt'}$. If this angle is at $\alpha(x)$, where $x \in (s, s')$, then $\alpha$ turns toward $A$, and hence away from $\beta(0)$, at $x$. Since $\alpha(x)$ lies in $B$ while $\alpha(0)$ lies in $B'$, conclusions (1) and (2) hold with $z = x$. If the angle is at $\beta(y)$, where $y \in (t, t')$, then $\beta$ turns toward $B$, and hence away from $\alpha(0)$, at $y$. Since $\beta(y)$ lies in $A$ while $\beta(0)$ lies in $A'$, conclusions (1) and (2) hold with $\alpha$ and $\beta$ interchanged, and with $z = y$. $\square$

A second technical lemma handles the construction of the unsafe half-cut within the strut. The strut is called $\sigma$, and the unsafe half-cut it contains is called $\tau$.

Lemma 5c.2. Let $\nu$ and $\omega$ be ideal routes of wires in a safe design $\Omega$. Let $\alpha$ and $\beta$ lift $\nu$ and $\omega$, respectively, and assume $\alpha \neq \beta$. Let $\sigma$ be a strut for $\nu$ at $z$, and let $\tilde{\sigma}$ be a lifting of $\sigma$ such that $\tilde{\sigma}(1) = \alpha(z)$ and $\alpha$ separates $\tilde{\sigma}(0)$ from the endpoints of $\beta$. Then $\tilde{\sigma}$ cannot intersect $\beta$.

Proof. We suppose that $\tilde{\sigma}$ does intersect $\beta$ and derive a contradiction. Because $\sigma$ is a strut, it is nondegenerate. Let $(s, b)$ be a crossing of $\tilde{\sigma}$ by $\beta$ that minimizes $s$. Then $\sigma_{sb}$ is a straight half-cut for $\omega$ at $b$. Call this half-cut $\tau$. Because $\tau$ is straight and $\omega$ is ideal, $\tau$ is either trivial or semisimple, by Proposition 5b.4. By assumption, $\alpha$ separates the terminal of $\tau$ (which is also the terminal of $\sigma$) from the endpoints of $\beta$. Hence for $\tau$ to be trivial, its terminal would have to be a terminal of $\alpha$ as well, making $\sigma$ trivial. But $\sigma$ is nontrivial, so $\tau$ is semisimple in $\Omega$. Proposition 4e.6 implies that $\tau$ is nondegenerate and that $\tau$ respects $\Omega$.

Now we show that $\tau$ is unsafe, contradicting the evasiveness of $\omega$. Because $\alpha$ and $\beta$ cohere, Corollary 4c.4 gives us a terminal of $\beta$ that is not shared by $\alpha$. Suppose that $\beta(e)$ lies on this terminal, where $e \in \{0, 1\}$. Let $\tilde{\chi}$ be the simple link
Routing a Safe Design

Chapter 5

\( \tilde{\sigma}_{0:z} \ast \beta_{b:z} \). The endpoints of \( \tilde{\chi} \) lie on opposite sides of \( \alpha \), and do not lie on either terminal of \( \alpha \). Hence \( \tilde{\chi} \) actually cuts \( \alpha \). If \( \chi \) denotes the cut \( p \circ \tilde{\chi} \), then there is a necessary crossing \((c, a)\) of \( \chi \) by \( v \). Applying Proposition 4d.2, we infer that

\[
\text{flow}(\chi, \Omega) \geq \text{flow}(\chi_{0:z}, \Omega) + \text{flow}(\chi_{1:z}, \Omega) + \text{width}(\omega).
\]

The link \( \chi \) is an associated cut for \( \tau \), and since \( \tau \) respects \( \Omega \), we have \( \text{flow}(\chi, \Omega) = \text{flow}(\tau, \omega) \) by Lemma 4d.3. Furthermore, \( \chi_{0:z} \) and \( \sigma \) are akin as half-cuts for \( v \), and hence have the same flow. We conclude that

\[
\text{flow}(\tau, \Omega) \geq \text{flow}(\sigma, \Omega) + \text{width}(\omega).
\]

Since \( \tau \) is shorter than \( \sigma \), it follows that \( \text{margin}(\tau, \Omega) \leq \text{margin}(\sigma, \Omega) - \text{width}(\omega) \), and the right-hand side is negative because \( \sigma \) is rigid. Therefore \( \text{margin}(\tau, \Omega) < 0 \), which means that \( \tau \) is unsafe in \( \Omega \). \( \square \)

Lemmas 5c.1 and 5c.2 are combined in the following proof.

**Proposition 5c.3.** Let \( v \) and \( \omega \) be ideal routes of wires in a safe design. If \( v(s) = \omega(t) \), then \( v = \omega \) and \( s = t \).

**Proof.** Let \( \Omega \) be the safe design, and let \( M \) be a blanket of its sheet \( S \) with covering map \( p: M \rightarrow S \). Suppose that \( v(s) = \omega(t) \). Lift \( v \) to \( \alpha \) and \( \omega \) to \( \beta \) so that \( \alpha(s) = \beta(t) \). Then \( \alpha \) and \( \beta \) are simple. If \( v \neq \omega \), then certainly \( \alpha \neq \beta \); if \( s \neq t \), then \( \alpha(s) \neq \alpha(t) \) because \( \alpha \) is simple, and hence \( \beta(t) \neq \alpha(t) \). In both cases \( \alpha \neq \beta \).

We use Lemma 5c.2 to derive a contradiction. It may be necessary to interchange \( v \) and \( \alpha \) with \( \omega \) and \( \beta \), but because of the symmetry between them, we only consider the case in which no exchange is needed. By Lemma 4c.5, the endpoints of \( \beta \) lie on the same side of \( \alpha \). Let \( A \) and \( A' \) be the scraps of \( M - \text{Im } \alpha \); name them so that \( \beta(0) \in A' \). Let \( B \) and \( B' \) denote the scraps of \( M - \text{Im } \beta \), and assume \( \alpha(0) \in B' \). Suppose we find a strut \( \sigma \) for \( \alpha \) at a point \( z \), and a lift \( \tilde{\sigma} \) of \( \sigma \) such that \( \tilde{\sigma}(1) = \alpha(z) \) and \( \tilde{\sigma}(0) \in A \). Since the endpoints of \( \beta \) do not lie in \( A \), Lemma 5c.2 will show that \( \tilde{\sigma} \) does not intersect \( \beta \). There are two cases to consider.

(A) If \( \alpha \) and \( \beta \) cross over, Lemma 5c.1 applies. Let \( z \) be the point given by Lemma 5c.1. By part (1), \( \alpha \) turns away from \( \beta(0) \) at \( z \), which means \( \alpha \) turns toward \( A \) at \( z \). Since \( v \) is taut, there is a strut \( \sigma \) for \( v \) at \( z \) and a lifting \( \tilde{\sigma} \) of \( \sigma \) such that \( \tilde{\sigma}(1) = \alpha(z) \) and \( \alpha \) turns toward \( \tilde{\sigma}(0) \) at \( z \). Hence \( \tilde{\sigma}(0) \) lies in \( A \), and Lemma 5c.2 implies that \( \beta \) does not intersect \( \tilde{\sigma} \). But by part (2) of Lemma 5c.1, \( \beta \) separates \( \alpha(z) \) from \( \alpha(0) \). As one can check, all the fringes of \( A \) lie in \( B' \), and hence \( \alpha(0) \) and \( \tilde{\sigma}(0) \) lie on the same side of \( \beta \). Thus \( \beta \) separates the endpoints of \( \tilde{\sigma} \), and so \( \tilde{\sigma} \) intersects \( \beta \), a contradiction.

(B) Suppose instead that \( \alpha \) and \( \beta \) do not cross over. Choose a maximal interval \([x, y] \subseteq \beta^{-1}(\text{Im } \alpha) \). Because \( \alpha \) and \( \beta \) are simple, we have \( \beta([x, y]) = \alpha([u, v]) \)
Figure 5c-2. Intersecting lifts of ideal routes. Figure 5c-1 does not cover the possibility that \( \alpha \) and \( \beta \) intersect without crossing over. But then at some point where \( \alpha \) and \( \beta \) touch, one turns away from the endpoints of the other, and essentially the same construction goes through.

for some interval \([u, v] \subset I\). (See Figure 5c-2.) There must be some point in \([u, v]\) at which \( \alpha \) turns toward \( A \), or some point in \([x, y]\) at which \( \beta \) turns toward \( B \). By symmetry, we may assume the former; say \( \alpha \) turns toward \( A \) at \( z \), where \( z \in [u, v] \). Because \( v \) is taut, there is a rigid half-cut \( \sigma \) for \( v \) at \( z \) such that if \( \sigma \) is lifted to \( \tilde{\sigma} \) with \( \tilde{\sigma}(1) = \alpha(z) \), then \( \tilde{\sigma}(0) \in A \). Now Lemma 5c.2 implies that \( \tilde{\sigma} \) cannot intersect \( \tilde{\beta} \). But \( \tilde{\sigma} \) intersects \( \beta \) at \( \tilde{\sigma}(1) \), again giving a contradiction. \( \square \)

By Lemma 5b.2 and Proposition 5c.3, the ideal routes of wires in a safe design are piecewise linear and injective, hence simple. And since they are link-homotopic to wires, their terminals are convex inner fringes. Therefore ideal routes are wires in their own right; we call them ideal embeddings or ideal wires. Proposition 5c.3 implies that the ideal wires form a design.

**Corollary 5c.4.** If every wire in a safe design is replaced by an ideal route, the result is a design. \( \square \)

We call it an ideal design. Because the flow across a cut is the same in all embeddings of a design, as is its capacity, a cut that is safe in a design is also safe in any embedding of the design. Furthermore, a cut that is major in a design is major in any embedding of that design. Therefore ideal designs are safe.

### 5D. Ideal Designs Are Properly Connected

The title of this section refers to Proposition 5d.4, the main result of this section: the articles of an ideal design have disjoint extents. This proposition goes a long way
toward showing that the ideal routes form a proper design, as defined in Section 4A. The most difficult part of Proposition 5d.4 is the claim that no two wires in an ideal design have overlapping extents. The method we use to prove this claim is similar to that used in Section 5C: given two ideal wires that are two close, we find a strut for one wire that gives rise to an unsafe, straight, nondegenerate half-cut for the other, contradicting the evasiveness of the second wire.

Figure 5d-1. When ideal wires approach too closely. As in Figure 5c-1, the links \( \alpha \) and \( \beta \) lift ideal routes, but this time they do not cross over. Instead, at a point of closest approach, \( \alpha \) turns away from \( \beta \). Since \( \alpha \) is taut, it has a strut \( \sigma \) at this angle whose lift \( \bar{\sigma} \) is shown. The straight path \( \bar{\tau} \) lifts a minimum-length mid-cut between \( \alpha \) and \( \beta \), and the bent half-link \( \bar{\sigma} \ast \bar{\tau} \) crosses over \( \alpha \). Together \( \alpha, \beta, \bar{\sigma}, \) and \( \bar{\tau} \) split the blanket into five scraps, here denoted \( A \) through \( E \).

Our analysis of ideal wires continues by examining the points at which they approach each other most closely. Figure 5d-1 illuminates the situation. If two nonintersecting taut wires are not parallel, there is a point at which the wires are closest and one turns away from the other, according to Lemma 5d.1 below. Concatenating the strut for that joint with a minimum-length mid-cut between the wires, one obtains a bent half-cut for the second wire that crosses over the first wire. We prove in Lemma 5d.2 that the flow across this bent half-cut is the flow across the strut plus the width of the first wire. If the two wires have overlapping extents, then the capacity of the bent half-cut exceeds the capacity of the strut by less than the width of the first wire. Hence the bent half-cut is unsafe. The technical difficulties arise in proving that it is nondegenerate and that it respects the design. Lemma 4f.6 then shows that the second wire has an unsafe, straight, nondegenerate half-cut, implying that it cannot be ideal.

Turning points, again

The first step is the geometric one of finding an appropriate joint. The result
we use is taken from [52].

**Lemma 5d.1.** Let \( \alpha \) and \( \beta \) be disjoint PL paths in \( \mathbb{R}^3 \). There are points \( \alpha(s) \) and \( \beta(t) \) such that \( \| \alpha(s) - \beta(t) \| \) is the minimum distance between Im \( \alpha \) and Im \( \beta \), and either

1. \( \alpha \) turns away from \( \beta(t) \) at \( s \); or
2. \( \beta \) turns away from \( \alpha(s) \) at \( t \); or
3. either \( s \in \{0,1\} \) or \( t \in \{0,1\} \). □

The proof is straightforward but messy; I refer the reader to [52].

One comment is in order about turning points in sheets and blankets. If \( \tilde{\alpha} \) lifts a link \( \alpha \) in a sheet, and \( \tilde{\alpha} \) turns toward a point \( z \) at \( x \), then \( \alpha \) turns toward the projection of \( z \) at \( x \) provided that some straight path \( \tilde{\tau} \) starting at \( z \) intersects Im \( \tilde{\alpha} \) only at \( \tilde{\alpha}(x) \).

**Construction of the bent half-cut**

The bulk of the technical work is performed by the following lemma. This lemma takes care to allow the two ideal wires to coincide, because we will also need this result to prove that ideal wires are self-avoiding. We say that a subcut \( \gamma \) is clean in a design \( \Omega \) if no wire in \( \Omega \) intersects the middle of \( \gamma \).

**Lemma 5d.2.** Let \( v \) and \( \omega \) be wires in an ideal design \( \Omega \), let \( \sigma \) be a strut for \( v \) at \( s \), and let \( \tau \) be a nondegenerate, clean, straight mid-cut between \( v \) at \( s \) and \( \omega \) at \( t \). If \( \sigma \cdot \tau \) crosses over \( v \) at \( \sigma(1) \), then \( \| \tau \| \geq \text{width}(v)/2 + \text{width}(\omega)/2 \).

**Proof.** Let \( \alpha, \beta, \tilde{\sigma}, \) and \( \tilde{\tau} \) be lifts of \( v, \omega, \sigma, \) and \( \tau \) that satisfy \( \tilde{\sigma}(1) = \alpha(s) = \tilde{\tau}(0) \) and \( \beta(t) = \tilde{\tau}(1) \). There can be no other intersections among these paths. First of all, \( \alpha \) and \( \beta \) cannot cross, and neither one can intersect \( \text{Mid} \tilde{\tau} \) because \( \tau \) is clean. Since \( \tilde{\sigma} \cdot \tilde{\tau} \) crosses over \( \alpha \) at \( \tilde{\sigma}(1) \), the link \( \alpha \) separates \( \beta \) and \( \text{Mid} \tilde{\tau} \) from \( \text{Mid} \tilde{\sigma} \). And finally, \( \alpha \) intersects \( \tilde{\sigma} \) only at \( \tilde{\sigma}(1) \) because \( \sigma \) is a strut for \( v \).

We now consider the bent half-cut \( \sigma \cdot \tau \) for \( \omega \) at \( t \). Let \( \tilde{\chi} \) denote the simple link \( \tilde{\sigma} \cdot \tilde{\tau} \cdot \tilde{\beta}_{1:1} \). Its projection \( \chi \) is a cut associated to \( \sigma \cdot \tau \), and in fact \( \text{flow}(\sigma \cdot \tau, \Omega) = \text{flow}(\chi, \Omega) \) by definition. Because \( \tau \) is nondegenerate, the terminals of \( \alpha \) and \( \beta \) are all distinct, and hence \( \tilde{\chi} \) cuts \( \alpha \). Define \( a \) by \( \tilde{\chi}(a) = \alpha(s) \). Then the crossing \((a, s)\) of \( \chi \) by \( v \) is necessary, and Proposition 4d.2 shows that

\[
\text{flow}(\chi, \Omega) \geq \text{flow}(\chi_{0:a}, \Omega) + \text{flow}(\chi_{1:a}, \Omega) + \text{width}(v).
\]

Now \( \chi_{0:a} \) is just \( \sigma \), and because \( \sigma \) is a strut, we have \( \text{flow}(\sigma, \Omega) = \text{cap}(\sigma, \Omega) \). Denote by \( X \) the fringe containing \( \sigma(0) \). Using the definition of capacity, we have

\[
\text{flow}(\sigma \cdot \tau, \Omega) \geq \text{cap}(\sigma, \Omega) + \text{width}(v)
= \|\sigma\| - \text{width}(X)/2 + \text{width}(v)/2.
\]

---

- 173 -
Suppose we can prove that $\sigma \ast \tau$ is safe. Then we can substitute $\text{cap}(\sigma \ast \tau)$ for $\text{flow}(\sigma \ast \tau, \Omega)$ in (5-5), obtaining the inequality

$$\|\sigma \ast \tau\| - \text{width}(X)/2 - \text{width}(\omega)/2 \geq \|\sigma\| - \text{width}(X)/2 + \text{width}(\nu)/2,$$

which implies the desired result $\|\tau\| \geq \text{width}(\nu)/2 + \text{width}(\omega)/2$.

The next step is to prove that $\sigma \ast \tau$ is nondegenerate in $\Omega$. Let $F$ denote the terminal of $\tilde{\sigma}$. If $\sigma \ast \tau$ were degenerate, then $F$ would be part of the same branch $B$ of $\Omega$ as the terminals of $\beta$. But $F$ and $\beta$ lie on opposite sides of $\alpha$. Hence that branch $B$ would intersect $\alpha$. Either $B$ would contain the terminals of $\alpha$, implying that $\sigma$ and $\tau$ are degenerate, or else $B$ would include a link $\eta$ that cut $\alpha$ and lifted a wire of $\Omega$. The latter is impossible, because $\alpha$ and $\eta$ would cohere. We conclude that $\sigma \ast \tau$ is nondegenerate.

To prove that $\sigma \ast \tau$ is safe, we use Lemma 4f.6. It implies that if $\sigma \ast \tau$ is an unsafe, nondegenerate, simple half-cut for $\omega$ that respects $\Omega$, then $\omega$ has an unsafe, nondegenerate, straight half-cut. Since $\omega$ is evasive, the latter is false. We already know that $\sigma \ast \tau$ is nondegenerate and simple, so it suffices to show that $\sigma \ast \tau$ respects $\Omega$, which is to say that its associated cuts respect $\Omega$. By symmetry, it is enough to show that $\chi$ respects $\Omega$. Let $\eta$ be any wire in $\Omega$, let $\tilde{\eta}$ and $\tilde{\eta}'$ be distinct lifts of $\eta$ in the same branch of $\Omega$, and suppose that $\tilde{\eta}$ cuts $\tilde{\chi}$. We must show that the terminals of $\tilde{\eta}'$ lie on the same side of $\tilde{\chi}$. I break the analysis into two cases.

1. Suppose first that $\tilde{\eta}$ is not in the branch of $\alpha$. It cannot be $\beta$, because $\beta$ does not cut $\tilde{\chi}$. Hence $\tilde{\eta}$ cannot intersect $\beta$, $\alpha$, or $\tilde{\tau}$ (since $\tau$ is clean). The terminals of $\tilde{\eta}$ must therefore be in the scraps $A$ and $E$ of Figure 5d-1. This means $\tilde{\eta}$ cuts the link $\tilde{\sigma} \ast \alpha_{s:1}$, whose projection is an associated cut of $\sigma$, and therefore respects $\Omega$ (since $\sigma$ does). Hence the terminals of $\tilde{\eta}'$ must lie on the same side of $\tilde{\sigma} \ast \alpha_{s:1}$; either they both lie in $A$ or both lie in $E$. In either case, they are on the same side of $\tilde{\chi}$.

2. Suppose now that $\tilde{\eta}$ is in the branch of $\alpha$. Since $\alpha$ cuts $\tilde{\chi}$, we may assume that $\tilde{\eta}$ is $\alpha$. Then $\eta'$ cannot intersect $\alpha$, $\beta$, or $\tilde{\tau}$. Furthermore, since $\tilde{\sigma}$ respects $\Omega$, the lift $\tilde{\eta}'$ cannot cut either $\tilde{\sigma} \ast \alpha_{s:0}$ or $\tilde{\sigma} \ast \alpha_{s:1}$. It follows that the terminals of $\tilde{\eta}'$ both intersect one of the five scraps $A$, $B$, $C$, $D$, and $E$. Hence $\tilde{\eta}$ does not cut $\tilde{\chi}$. For $\tilde{\eta}'$ to share a terminal with $\tilde{\chi}$, it would have to share a terminal either with $\tilde{\sigma}(0)$ or with $\beta$. Then $\alpha$ would be part of the branch containing either $\tilde{\sigma}(0)$ or a terminal of $\beta$. The former option is ruled out because $\tilde{\sigma}$ is nondegenerate; the latter option is ruled out because $\tilde{\tau}$ is nondegenerate.

Thus $\chi$ respects $\Omega$, and the proof is complete. □

Lemma 5d.2 represents the peak of technical difficulty in the entire thesis. It brings together all the concepts we have been studying: respect, degeneracy, safety,
struts, and more. There are some formidable foothills ahead, but if you have made it this far, you should be able to surmount them.

The extents of details

Proposition 5d.4 puts Lemmas 5d.1 and 5d.2 together to show that ideal wires have disjoint extents. It also shows that if two fringes in different articles have overlapping extents, then the design admits a major cut that is straight and unsafe; and if a wire's extent overlaps with that of a fringe other than one of its terminals, then the design contains a major cut or nondegenerate half-cut that is straight and unsafe. Neither of these things can happen in an ideal design. First we prove the most basic of these results.

**Lemma 5d.3.** If two fringes in a design have overlapping extents, then the design admits an unsafe, straight, nonempty cut. The cut is also nondegenerate if the fringes lie in different articles.

**Proof.** Let $\Omega$ be an ideal design on the sheet $S$. Let $A$ and $B$ be two different fringes of $S$, and suppose their extents overlap. Choose points $a \in A$ and $b \in B$ to minimize $\|a - b\|$, and $\sigma$ be the straight path $a \triangleright b$. Then we have

$$\|\sigma\| < \text{width}(A)/2 + \text{width}(B)/2$$

(5-6)

because the extents of $A$ and $B$ intersect. Neither $A$ nor $B$ touches $\text{Mid} \sigma$. If no other fringes of $S$ do, then $\sigma$ is the desired straight cut: it is nonempty because $A \neq B$, unsafe because inequality (5-6) implies $\text{cap}(\sigma) < 0$, and nondegenerate if $A$ and $B$ lie in different articles.

Now suppose $\sigma$ intersects a fringe $C \notin \{A, B\}$. We replace $\sigma$ by a shorter path $\tau$ with the same properties. By inequality (5-6), the set $\text{Im} \sigma$ is contained in the union of the extents of $A$ and $B$. Hence for some $D \in \{A, B\}$ the fringe $C$ lies within $\text{width}(D)/2$ units of $D$. Let $\tau$ be the shortest subpath of $\sigma$ that runs from $D$ to $C$. Then we have the analogue of inequality (5-6) for $\tau$, namely $\|\tau\| < \text{width}(D)/2 + \text{width}(C)/2$. Moreover, we may assume that $C$ and $D$ lie in different articles if $A$ and $B$ do. For if $C$ and $A$ are fringes of the same article, namely the terminals of some wire, then they have the same width, and we may choose $D = B$. Similarly, if $C$ and $B$ fall in the same article, we may choose $D = A$. Since $\tau$ intersects fewer fringes than $\sigma$, the lemma follows by induction on this quantity. ☐

Now for the real result.

**Proposition 5d.4.** The articles of an ideal design have disjoint extents.

**Proof.** Let $\Omega$ be an ideal design on the sheet $S$. Let $A$ and $B$ be two different details of $\Omega$, and assume they lie in different articles of $\Omega$. We say that $A$ and $B$ are
too close if the distance between them (measured in the wiring norm) is less than width(A)/2 + width(B)/2. If A and B have overlapping extents, then A and B are too close. Supposing that A and B are too close, we derive a contradiction. Let d denote width(A)/2 + width(B)/2.

Case 1. Suppose that A and B are features. Then by Lemma 5d.3, the design Ω has an unsafe, straight, nonempty, nondegenerate cut. Since a nonempty and nondegenerate cut is major, this cut makes Ω unsafe.

Case 2. Let A be a feature and B a wire ω that does not touch A. Let σ be a minimum-length linear path from A to B; say σ(1) = ω(t). We have ∥σ∥ < d. We show that σ contains an unsafe straight cut or an unsafe straight half-cut for ω. In either case, Ω cannot be an ideal design. If no fringe of S touches σ(1) or the middle of σ, then σ is a half-cut for ω at t; it is nondegenerate because A is not a terminal of ω, and it is unsafe because ∥σ∥ < d implies cap(σ) < 0. Suppose instead that a fringe C touches σ(1) or Mid σ. If C is a terminal of σ, then A and C are too close (because width(C) ≥ width(ω)) and Case 1 applies. Otherwise C is either too close to A or too close to ω, and we use the same type of induction as in Case 1.

Case 3. The interesting case is when A and B are both wires. We apply Lemma 5d.1 to these wires, call them u and ω. If some endpoint of u or ω lies within d units of the other wire, then so does the terminal containing that endpoint, and we reduce to the previous case. Otherwise, there are points s, t ∈ (0, 1) such that ∥u(s) − ω(t)∥ < d, and either u turns away from ω(t) at s, or ω turns away from u(s) at t. By symmetry, we may assume the former.

Now we apply Lemma 5d.2. Let σ be a strut for u at s; there must be one because u is taut. Let τ be the straight mid-cut from u(s) to ω(t). This path does not intersect any fringe of S, or we could reduce to the previous case. Similarly, we can assume that Mid τ intersects no wire in Ω. Thus τ is clean in Ω, and because it connects different wires in a design, τ is nondegenerate. Finally, σ ∗ τ crosses over u at u(s), because u turns away from τ(1) at s, but u turns toward σ(0) at s. Applying Lemma 5d.2 to u, ω, σ, and τ, we see that ∥τ∥ ≥ d, contrary to assumption. ☐

5E. Ideal Wires Are Self-Avoiding

This section completes the proof that ideal designs are proper by showing that the wires of ideal designs are self-avoiding. The technique we use involves some fairly messy geometry, illustrated in Figure 5e-1. Beginning with a divisive article, we increase its width until just before its extent divides the sheet. At this point the frontier of its extent consists of two or more polygone linked by simple paths. One of these polygons surrounds the others, and one of the inner polygons surrounds an
inner fringe of the sheet. Across one of the simple paths we find an unsafe, straight, nondegenerate subcut. If the article contains a wire, and the subcut is a cut, we prove the cut is major.

Figure 5e-1. When an article is divisive. The shaded region represents a fractional extent of the article $C$, just before it divides the sheet by separating the fringes $X$ and $Y$. It intersects no articles except $C$. Because this region has a vertex at $x$, and includes points on both sides of the path $\alpha$, the dark points and line segments (at right) must contain points of $C$. Hence there is a bent subcut $\eta$ for $C$ which, together with $C$, separates $X$ from $Y$. We use this fact to show that $\eta$ is nondegenerate. Also $\eta$ is unsafe; its capacity is negative. The straight subcut $\kappa$ has the same properties.

Fractional extents

To study self-avoidance, we adjust the widths of design details and examine the moment when an article first fails to self-avoid. Let $C$ be any article of a design, and suppose $\delta \geq 0$. The $\delta$-extent of $C$, denoted $T_{\delta}(C)$, is the extent that $C$ would have if the widths of its details were multiplied by $\delta$. (By convention, the 0-extent of $C$ is the intersection of $T_{\delta}(C)$ for $\delta > 0$.) The set $T_{\delta}(C)$ is open unless $\delta = 0$. Since $T_{1}(C)$ is just the extent of $C$, the article $C$ is self-avoiding if and only if $T_{1}(C)$ divides the sheet.

Given a divisive article $C$, we find a critical value of $\delta$ for which the $\delta$-territory of $C$ looks like that in Figure 5e-1. The following lemma assists the search for a critical value of $\delta$.

**Lemma 5e.1.** Let $D_0 \supseteq D_1 \supseteq D_2 \supseteq \cdots$ be a descending chain of closed, connected subsets of $R^n$. If $D_0 - \bigcap_{i=0}^\infty D_i$ is bounded, then $\bigcap_{i=0}^\infty D_i$ is connected.

**Proof.** Set $D = \bigcap_{i=0}^\infty D_i$, and suppose that $D$ is not connected. Let $C$ and $D - C$ be nonempty sets that are both open and closed in $D$. Since $D$ is closed, they are closed in $R^n$. Because $R^n$ is normal, there are disjoint open sets $U$ and $V$ containing $C$ and $D - C$, respectively. Write $X = D_0 - (U \cup V)$. Then $X$ is closed,
and because \( X \subseteq D_0 - D \), it is also bounded. Hence \( X \) is compact. Now each of the connected sets \( D_i \) contains points of \( U \) and \( V \), and hence must also contain points of \( X \). It follows that the collection of closed sets \( \{ D_i \cap X \} \) satisfies the finite intersection condition, because if \( M \) is any finite subset of the natural numbers, it has a maximum value \( m \), and

\[
\bigcap_{i \in M} (D_i \cap X) = D_m \cap X,
\]

which is nonempty. Because \( X \) is compact, the intersection \( \bigcap_{i} (D_i \cap X) \) must be nonempty. But that intersection is precisely \( D \cap X \), which is empty. This contradiction establishes the lemma. \( \square \)

And the next lemma gives us a value \( \delta \) with the desired properties.

**Lemma 5e.2.** If an article \( C \) of an ideal design is divisive, then there exists a number \( \delta \in (0,1) \) such that \( T_\delta(C) \) does not divide the sheet, but its closure does.

**Proof.** Let \( S \) be the sheet. By Proposition 5d.3, the extent \( T_1(C) \) of \( C \) does not intersect any fringes except those in \( C \). For \( C \) not to self-avoid means that \( T_1(C) \) divides \( S \). On the other hand, \( T_1(C) \) does not divide \( S \) for sufficiently small \( \epsilon \). Hence the quantity

\[
\delta = \inf \{ \epsilon > 0 : T_\epsilon(C) \text{ divides } S \}
\]

is positive, and at most 1.

We shows that \( \delta < 1 \) by proving that \( T_\delta(C) \) does not divide \( S \). For \( \epsilon < \delta \), the set \( T_\epsilon(C) \) does not divide \( S \), and hence all the fringes of \( S \) except those in \( C \) lie in a single component \( F_\delta \) of its complement. Furthermore, for \( n \geq 2 \) the sets \( F_{\delta - \epsilon \delta / n} \) form a descending chain of connected closed sets. Call their intersection \( F_\delta \). If \( C \) is not the outer fringe of \( S \), then the complement of \( F_\delta \) is bounded; otherwise \( F_{\delta / 2} \) is bounded, and in either case Lemma 5e.1 applies. It shows that \( F_\delta \) is connected. Since \( \bigcup_{n \geq 2} T_{\delta - \epsilon \delta / n}(C) = T_\delta(C) \), we have \( F_\delta \subseteq R^2 - T_\delta(C) \). And since \( F_\delta \) contains all the fringes of \( S \) except those in \( C \), it follows that \( T_\delta(C) \) does not divide \( S \).

Now we indicate why \( \text{Cl}\, T_\delta(C) \) divides the sheet \( S \). Write \( V = R^2 - \text{Cl}\, T_\delta(C) \). Then \( V \) is open, and because the wiring norm is polygonal, \( V \) is bounded by finitely many line segments. If all fringes except those in \( C \) lay in the same component of \( V \), we could connect them by paths in \( V \). The images of these paths, being compact, would lie some finite distance from \( \text{Cl}\, T_\delta \). Hence they would also exist in \( R^2 - T_{\delta + \epsilon}(C) \) for all sufficiently small \( \epsilon \). But by the definition of \( \delta \), the set \( T_{\delta + \epsilon}(C) \) divides \( S \) for arbitrarily small positive values of \( \epsilon \). \( \square \)

**Deriving unsafe subcuts**

Next we need a condition for a subcut to be nondegenerate. If \( \sigma \) is any subcut
whose endpoints lie in the same article $C$, a completion of $\sigma$ is any loop $\sigma \star \kappa$ where $\kappa$ is a path in $C$.

**Lemma 5e.3.** Let $\sigma$ be a degenerate subcut in the design $\Omega$. If the endpoints of $\sigma$ lie in an article $C$ of $\Omega$, then no simple completion of $\sigma$ separates two fringes that are not part of $C$.

**Proof.** Because $\sigma$ is degenerate, there is a path $\tau \in [\sigma]_P$ that lies entirely in $C$. This is true by definition if $\sigma$ is a cut. If $\sigma$ is a half-cut for $\omega$ at $t$, then $\sigma \star \omega_{t,1}$ is path-homotopic to a path $\tau$ in $C$. It follows from the groupoid properties of concatenation (Section 2A) that $\sigma \simeq_P \tau \star \omega_{1,t}$, and the right-hand path lies in $C$. A similar argument applies to mid-cuts.

Let $\sigma \star \kappa$ be any simple completion of $\sigma$. Then $\sigma \star \kappa$ and $\tau \star \kappa$ are path-homotopic, and enclose the same fringes. By Proposition 2c.5, the fringes enclosed by $\tau \star \kappa$ are precisely those lying inside $\sigma \star \kappa$. Now $\tau \star \kappa$ lies entirely in $C$. If $C$ contains a wire of $\Omega$, then it comprises two inner fringes of $S$ connected by a thread. Then the other fringes of $S$ all lie in the unbounded component of $R^2 - Im(\tau \star \kappa)$, whence by Proposition 2c.5, the loop $\tau \star \kappa$ does not enclose any of those fringes. Or if $C$ is a single fringe, then $\tau \star \kappa$ may enclose $C$, or no fringes, or all the fringes but $C$ (if $C$ is the outer fringe). In no case are there fringes $X$ and $Y$ not in $C$ such that $\tau \star \kappa$ encloses $X$ but not $Y$. \(\square\)

The last lemma of the chapter outlines the geometric construction suggested by Figure 5e-1.

**Lemma 5e.4.** If $C$ is a divisive article of an ideal design $\Omega$, then it has a clean, straight, unsafe, nondegenerate subcut $\kappa$. Furthermore, if $C$ includes a wire $\xi$, and if $\kappa$ is a mid-cut between $\xi$ at $s$ and $\xi$ at $t$, then $\xi$ turns away from $\xi(t)$ at $s$.

**Proof.** Let $S$ be the sheet of $\Omega$. First apply Lemma 5e.2 to the article $C$, and let $\delta \in (0,1)$ be the quantity defined by that lemma. Write $D$ for the open set $T_5(C)$. Because $\Omega$ is ideal and $D \subset T_1(C)$, Proposition 5d.4 implies that neither $D$ nor $ClD$ intersects any fringes of $S$ other than those in $C$. Because $ClD$ divides $S$, there are two fringes of $S$ that fall in different components of $R^2 - ClD$. Call these fringes $X$ and $Y$. Since $D$ does not divide $S$, there is a simple path $\alpha$ from $X$ to $Y$ in $R^2 - D$. (The set $R^2 - D$ is locally path-connected, being a finite union of polygons and line segments.) Clearly $\alpha$ must enter $ClD - D = FrD$. Let $x$ be the point of $R^2$ where $\alpha$ first enters $FrD$. Figure 5e-1 pictures the situation near $x$. The shaded region represents $D$.

We find a straight subcut $\kappa$ through $x$. Because $x$ lies in $FrD$, there is a point $p$ of $C$ such that $\|p - x\| = \delta \cdot \text{width}(p)/2$, where by width($p$) we mean the width of the detail containing $p$. In fact, there must be points of this sort on both sides of $\alpha$; call them $p$ and $q$. Then $p \triangleright x$ and $q \triangleright x$ contact $\alpha$ from both sides, and the bent
path \eta = (p \rightarrow x) \ast (x \rightarrow q) intersects C only at its endpoints; its middle lies in D. Let \kappa be the linear path p \rightarrow q. By the triangle inequality we have

\|\kappa\| \leq \delta \cdot \text{width}(p)/2 + \delta \cdot \text{width}(q)/2.  \tag{5-7}

I claim that Mid \kappa intersects no article of \Omega. We already know that Mid \kappa is disjoint from C, and if any other article of \Omega touched Im \kappa, then its extent would overlap that of C, contradicting Proposition 5d.4. So \kappa is a clean subcut in S, and its capacity is negative by inequality (5-7), since \delta < 1. Hence \kappa is unsafe. Furthermore, it is apparent from the geometry of Figure 5e-1 that if both endpoints of \kappa lie in the middle of some wire \xi \in \Omega, then at one of those two points, \xi turns away from the other.

It remains to prove that \kappa is nondegenerate. Choose a simple loop \kappa \ast \gamma that is a completion of \kappa. I claim that \eta \ast \gamma, which is also simple loop, separates X from Y. For \alpha crosses over \eta \ast \gamma at the point x and nowhere else. Since X and Y lie at the endpoints of \alpha and do not intersect Im \eta or the article C that contains Im \gamma, they lie in different components of \(R^2 - \text{Im}(\eta \ast \gamma)\). Now consider \kappa \ast \gamma. All points within the triangle \Delta pqx are within \delta \cdot \text{width}(p)/2 units of p or within \delta \cdot \text{width}(q)/2 units of q, and hence neither X nor Y intersects \Delta pqx or its inside. So \kappa \ast \gamma also separates X from Y, and neither X nor Y is part of C. Hence by Lemma 5e.3, the subcut \kappa is nondegenerate in \Omega. \qed

Conclusions

One consequence of Lemma 5e.4 is that every divisive fringe has an unsafe, straight, nondegenerate cut. Another is that ideal wires are self-avoiding. The proof simply combines Lemma 5e.4 with Lemma 5d.2 from the preceding section.

Proposition 5e.5. Ideal wires are self-avoiding.

Proof. Let \Omega be an ideal design on the sheet S, and let \omega be a wire in \Omega. Suppose \omega is not self-avoiding, meaning that its article C is divisive. Apply Lemma 5e.4 to C, and let \kappa be the resulting subcut for \omega; it is clean, straight, unsafe, and nondegenerate. There are three cases: \kappa can be a cut, a half-cut, or a mid-cut. If \kappa is a cut, then because \kappa is straight and terminals are convex, \kappa must connect the terminals of \omega. Hence \kappa is a nonempty, and therefore major, unsafe straight cut of \Omega, contradicting the safety of \Omega. If \kappa is a half-cut, then \omega is not evasive, contradicting the assumption that \Omega is ideal. The remaining case is the interesting one.

Suppose that \kappa is a mid-cut between \omega at s and \omega at t. By Lemma 5e.4, \omega turns away from \omega(t) at s. Because \omega is taut, it has a strut \sigma at s. We apply Lemma 5d.2 with \omega representing both wires and with \kappa in place of \tau. The conditions are easily
checked: Lemma 5e.4 says that $\kappa$ is clean, straight, and nondegenerate; and $\sigma \star \tau$ crosses over $\omega$ at $\sigma(1)$ because $\omega$ turns toward $\sigma(0)$ but away from $\kappa(1)$ at $\omega(s)$. The conclusion of Lemma 5d.2 is that $\|\kappa\| \geq width(\omega)$. But Lemma 5e.4 says that $\kappa$ is unsafe, and since $flow(\kappa, \Omega) = 0$, this means $cap(\kappa) < 0$. But the capacity of $\kappa$ is just $\|\kappa\| - width(\omega)$, which we have just shown to be nonnegative. This contradiction completes the proof. ⪫

Together, Propositions 5e.5 and 5d.4 show that ideal designs are proper. And since every safe design has an ideal embedding, by Proposition 5b.3 and Corollary 5c.4, we obtain the following result.

**Theorem 5e.6.** Every ideal design is proper, and every safe design is routable.
In particular, after the loop completes, we have $\ell(\alpha_n) + t_n \leq \sum_{j=0}^{n-1} \ell(\pi_j)$. Let $T$ be the skeleton of $G$. The concatenation of $\alpha_n$ and $\pi_n$, written $(\alpha_n \cdot \pi_n)$, is a path in $T$ from $u_n$ to $y$. Hence by Proposition 9d.4,

$$\text{DIST}(x, y) \leq t_n + \ell(\alpha_n \cdot \pi_n)$$
$$= (\ell(\alpha_n) + t_n) + \ell(\pi_n)$$
$$\leq (\sum_{j=0}^{n-1} \ell(\pi_j)) + \ell(\pi_n)$$
$$= \ell(\pi) = \text{flow}(\chi).$$

Thus the invariant implies the lemma.

**Figure 9d-2.** The inductive step in proving the correctness of Algorithm F. A minimal path from $x$ to $y$ is $\pi_0 \cdots \pi_n$ (on the right), and the greedy path found by Algorithm F is $\rho_0 \cdots \rho_n$ (on the left). The induction hypothesis is that $\ell(\rho_0 \cdots \rho_{i-1}) + \ell(\alpha) \leq \ell(\pi_0 \cdots \pi_{i-1})$.

It remains to prove the invariant, which we do by induction on $i$. The basis case $i = 0$ is trivial. Now assuming the invariant for $i$, we prove it for $i + 1$. See Figure 9d-2. The path $\rho_i$ represents a shortest path in $T$ from $u_i$ to a node bordering $\gamma_{i+1}$ from below, and the paths $\alpha_{i+1}$ and $\alpha'_{i+1}$ are the shortest paths in $T$ between the indicated nodes. Since $T$ is a forest, by Lemma 9d.2, the shortest path between two nodes in $T$ is the unique simple path between them. In particular, $\alpha'_{i+1}$ is the unique simple path between its endpoints. The nodes adjacent to $\gamma_{i+1}$ from below are connected in $T$, and hence every node along $\alpha'_{i+1}$ is adjacent to $\gamma_{i+1}$ from below. It follows that $\rho_i \cdot \alpha_{i+1}$ is a simple path—the shortest path between $u_i$ and the end of $\pi_i$—and thus we have

$$\ell(\rho_i) + \ell(\alpha'_{i+1}) \leq \ell(\alpha_i) + \ell(\pi_i).$$

Combining this inequality with the induction hypothesis, we obtain

$$(t_i + \ell(\rho_i)) + \ell(\alpha'_{i+1}) \leq \sum_{j=0}^{i} \ell(\pi_j).$$
The first term on the right is just $t_{i+1}$, and Lemma 9d.3 shows that $\ell(\alpha'_{i+1}) = \ell(\alpha_{i+1})$. We conclude that the invariant holds for $i + 1$. \qed

9E. The Abstract Compaction Algorithm

To prove the correctness of Algorithm C, the compaction algorithm, we proceed by way of an intermediate procedure called Algorithm A, the abstract compaction algorithm. The name derives from the fact that Algorithm A (which is not really an algorithm at all, but just a mathematical definition) abstracts the essential element of Algorithm C, namely the iterative definition of the subspace of configurations to be searched for a minimum sketch. Algorithm A defines a sequence $A_0, A_1, \ldots, A_m$ of increasingly restricted subsets of the configuration space. These sets will correspond to sets of configurations satisfying the constraint system $I$ at different stages of Algorithm C.

The first part of this section is devoted to the statement of Algorithm A and its preconditions. The rest of the section demonstrates the correctness of Algorithm A by proving the following theorem.

**Theorem 9e.1.** The output $A_m$ of Algorithm A is the connected component of $\{ c \in C(S) : S(c) \text{ is routable} \}$ that contains the initial configuration 0.

Section 9F will draw a correspondence between Algorithms C and A, and prove that $A_m$ is precisely the set of configurations that satisfy the final constraint system $I$ of Algorithm C. Together with Theorem 9e.1, this implies that the constraints generated by Algorithm C are both necessary and optimal, if only convex constraints are allowed. Finally, because Algorithm C finds an optimal configuration among those satisfying the constraint system, it will follow that Algorithm C is correct, and that it finds the best solution available to any algorithm of its type.

There are at least two reasons for taking this abstract approach. First of all, it simplifies the correctness proof by separating the mathematical from the algorithmic concerns. Second, and more important, it clarifies the assumptions on which the compaction algorithm relies. An understanding of these assumptions will allow Algorithm C to be easily modified.

**Assumptions**

The input to Algorithm A is a routable sketch $S$ together with a sequence $\Psi(S) = (\psi_1, \ldots, \psi_m)$ of potential cuts of $S$; the output is a set of configurations $A_m$. As a precondition of Algorithm A, the potential cuts $\Psi(S)$ must determine the routability of the modified sketches $S(d)$. Specifically, they must have the following property.
**Routability property.** Assume $S(0)$ is routable, and let $d \in C(S)$ be a configuration. If $d$ fails to protect some element of $\Psi$, then $S(d)$ is not routable. But if for all $t \in [0, 1]$ the configuration $td$ protects every $\psi \in \Psi(S)$, then $S(d)$ is routable.

The capacities of the potential cuts must also have a special property.

**Convexity property.** For each $\psi \in \Psi(S)$, the function $d \mapsto cap(\psi(d))$ is convex.

Actually a weaker property suffices, namely that for each line $L$ in configuration space, there is a point $c$ of $L$ at which the capacity $cap(\psi(c))$ is minimal, and $cap(\psi(d))$ is nondecreasing as $d$ moves away from $c$ along $L$. The simpler condition of convexity is general enough for my purposes, however.

In principle, my compaction method depends on only two further assumptions about the potential cuts $\Psi(S)$.

**Ordering property.** Let the configuration $d \in C(S)$ protect $\psi_i$ for all $i < k$. If $d$ lies on the frontier of $\{ c \in C(S) : \psi(c) \text{ is a cut} \}$, then every cut that is a subpath of $\psi_k(d)$ is either safe or empty.

**Boundary property.** The configuration space $C(S)$ is open in $R^n$, and there is a closed set $D \subseteq R^n$ such that all configurations in $C(S) - D$ fail to protect some potential cut in $\Psi$.

In practice, of course, we also desire that the sequence $\langle \psi_i \rangle$ be computable in polynomial time. As we show in Section 9F, the sequence of potential cuts examined by Algorithm $C$ has all these desirable properties.

The abstract algorithm

Before plunging into the algorithm, I shall provide a brief overview. Algorithm $A$ computes a sequence of polytopes in configuration space, each one contained in the last. The configurations in the $k$th polytope will protect the first $k$ potential cuts in $\Psi(S)$. To process $\psi_k$, the $k$th potential cut, the algorithm first determines whether $\psi_k$ is unsafe and nonempty in any configuration in the current polytope. If not, the algorithm ignores $\psi_k$. Otherwise, it defines a set of unacceptable configurations in which the capacity of $\psi_k$ falls below some critical value. This set contains all configurations in the current polytope that fail to protect $\psi_k$. Its complement consists of two half-spaces: one in which the lower endpoint of $\psi_k$ is far to the right of the upper endpoint, and one in which the situation is reversed. Because the initial configuration is always acceptable, it must fall into one half-space or the other; the $k$th polytope is determined by intersecting the $(k-1)$st polytope with the half-space that contains 0. Thus Algorithm $A$ eliminates configurations not reachable from the initial one.
Algorithm A. (Finds the set of acceptable modifications of a sketch.)
Input: a legal sketch $S$ with $n$ modules specified, and a sequence $(\psi_1, \ldots, \psi_m)$ of potential cuts of $S$ with the routability, convexity, ordering, and boundary properties.
Output: a subset of the configuration space $C(S)$.
Local variables: an integer $k$, polytopes $A_k$ of acceptable configurations, sets $U_k$ of unacceptable configurations, and inequalities $\Lambda_k$.
1. $A_0 \leftarrow C(S)$;
2. for $k \leftarrow 1$ to $m$ do
3. if some $c \in A_{k-1}$ does not protect $\psi_k$ then
4. $U_k \leftarrow \{ d \in R^n : \text{cap}(\psi_k(d)) < \text{flow}(\psi_k(c)) \}$;
5. if the endpoints of $\psi_k$ lie on the features $P_k$ and $Q_k$, write $U_k$ as
   \[ \{ d : \Delta^- < \Delta_{P_kQ_k}(d) < \Delta^+ \}. \]
   Either $0 \in (\Delta^-, \Delta^+]$ or $0 \in [\Delta^+, \infty)$.\!
6. $\Lambda_k \leftarrow \begin{cases} \Delta_{P_kQ_k}(d) \geq \Delta^+, & \text{if } 0 \geq \Delta^+; \\ \Delta_{P_kQ_k}(d) \leq \Delta^-, & \text{if } 0 \leq \Delta^-; \end{cases}$
7. $A_k \leftarrow \{ d \in A_{k-1} : d \text{ satisfies } \Lambda_k \}$
8. else $A_k \leftarrow A_{k-1}; U_k \leftarrow \emptyset;
9. return $A_m$.

Some remarks about Algorithm A are in order.

- The set $U_k$ is defined in terms of an arbitrary configuration $c \in A_{k-1}$ that fails to protect $\psi_k$. We will soon show that $U_k$ is independent of the choice of $c$.
- The constraint $\Lambda_k$ is a simple linear inequality between $d_{\mu(P_k)}$ and $d_{\mu(Q_k)}$, and hence defines a closed half-space in $R^n$. Since $A_0$ is convex, the set $A_k$ is therefore convex for each $k$.
- In the light of the following results, the definition of $A_k$ in lines 6–7 may be read as “$A_k$ is the component of $A_{k-1} - U_k$ that contains $0$”.

Core of the correctness proof

The following definition and lemma are fundamental to the correctness proof. The lemma’s proof reveals the purpose of the convexity and ordering properties.

Definition 9e.2. Two configurations, $d$ and $d'$, are equivalent with respect to a potential cut $\psi$ if for every configuration $b_t = (1 - t)d + rd'$ with $t \in [0, 1]$ the path $\psi(b_t)$ is a cut. This relation is written ‘$d \approx d'$ with respect to $\psi$’.

In configurations that are equivalent with respect to a potential cut $\psi$, the flow across $\psi$ is equal. To see why, suppose $d \approx d'$ with respect to $\psi$, and let
$H: \mathbb{R}^2 \times \mathcal{C}(S) \to \mathbb{R}^2$ be the map used to define the sketch $S(c)$ for $c \in \mathcal{C}(S)$. (We have $S(c) = H(\cdot, c) \circ S(0)$.) Because $H(\cdot, d)$ and $H(\cdot, d')$ are homeomorphisms, we have

- flow across $\psi(d)$ in $S(d) = \text{flow across } H(\cdot, d)^{-1} \circ \psi(d)$ in $S(0)$, and
- flow across $\psi(d')$ in $S(d') = \text{flow across } H(\cdot, d')^{-1} \circ \psi(d')$ in $S(0)$.

A bridge homotopy between the two cuts on the right is $(s, t) \mapsto \theta_t(s)$ where $\theta_t = H(\cdot, b_t)^{-1} \circ \psi(b_t)$. (Both $\psi$ and $H$ are piecewise linear.) Hence they have the same flow (congestion) in $S(0)$ by Corollary 8a.6. For similar reasons, either $\psi(d)$ and $\psi(d')$ are both empty or else both are nonempty.

**Lemma 9e.3.** Let $\psi$ be a potential cut in the sketch $S$, let $d$ and $d'$ be configurations in $\mathcal{C}(S)$, and put $b_t = (1-t)d + td'$ for $t \in [0, 1]$. Suppose that the capacity function $t \mapsto \text{cap}(\psi(b_t))$ is convex, and that whenever $b_t$ lies on the frontier of $\{ c \in \mathcal{C}(S) : \psi(c) \text{ is a cut} \}$, all cuts that are subpaths of $\psi(b)$ are safe or empty.

1. If $d'$ protects $\psi$ but $d$ does not, then $\text{cap}(\psi(d')) \geq \text{flow}(\psi(d))$.

2. If neither $d$ nor $d'$ protects $\psi$, then $d \approx d'$ with respect to $\psi$.

**Proof.** As $t$ varies from 0 to 1, the sketch $S(b_t)$ varies from $S(d)$ to $S(d')$, and the linear path $\sigma_t = \psi(b_t)$ is sometimes a cut, and sometimes it crosses features. Denote the flow across $\sigma_t$ by $f_t = \text{flow}(\psi(b_t))$, and the capacity (or "length") of $\sigma_t$ by $l_t = \text{cap}(\psi(b_t))$.

We first argue that the set $Z = \{ t \in [0, 1] : \sigma_t \text{ is a cut} \}$, considered as a subspace of the unit interval, is open. Let $\sigma_t$ be a cut; say it connects the features $P$ and $Q$. There is some positive distance between $\sigma_t$ and every feature but $P$ and $Q$; because $b_t \in \mathcal{C}(S)$, no other features can touch the endpoints of $\sigma_t$. And since $\sigma_t$ and the module positions in $S(b_t)$ are all continuous functions of $t$, there is some neighborhood $U$ of $t$ such that $\sigma_u$ is a cut whenever $u \in U$. So $Z$ is open, and hence it consists of disjoint intervals, each one open in $[0, 1]$.

Now let us focus attention on one of these intervals, call it $T$. For all $s, t \in T$ the configurations $b_s$ and $b_t$ are equivalent with respect to $\psi$. Hence the flow $f_t$ is a constant $f_T$ for all $t \in T$. And if $s$ lies on the frontier of $T$, considering $T$ as a subspace of $I$, then $\sigma_s$ is not a cut. The following claim is the crux of the argument.

**Claim:** If $t \in T$ and $s \in \text{Fr } T$, the configuration $b_t$ protects $\psi$ unless $l_t < l_s$.

Consider the sketch $S(b_s)$. At this point, one or more features have just contacted $\sigma_s$, and hence $\sigma_s$ is broken up into a sequence of cuts $\alpha_1, \ldots, \alpha_l$. Because $b_s$ is on the frontier of the set of configurations that make $\psi$ a cut, all the cuts $\alpha_i$ are safe or empty. If the cut $\sigma_t$ is empty, then $\psi$ is fixed with respect to the module that contains its endpoints, and so the cuts $\alpha_1$ and $\alpha_l$ must connect different modules. Thus $\alpha_1$ and $\alpha_l$ are safe, not empty; their capacities are nonnegative.
Hence \( l_s \geq 0 \) also. In this case \( f_T \leq l_s \) because \( f_T = 0 \). Now suppose that \( \sigma_t \) is not empty. If \( f_T \) were to exceed \( l_s \), one of these cuts \( \alpha_i \) would be unsafe and nonempty. One could prove this rigorously by passing to the design model and appealing to Proposition 4f.1 and Lemma 4f.3. We conclude that \( f_T \leq l_s \). If \( b_t \) fails to protect \( \psi \) then \( l_t < f_T \), so \( l_t < l_s \). This proves the claim.

The lemma is now straightforward. Both parts of the lemma assume that \( d \) fails to protect \( \psi \), so we may assume that \( \sigma_0 = \psi(d) \) is a cut, and that \( l_0 < f_0 \). Suppose first that \( d \) and \( d' \) are equivalent with respect to \( \psi \). Then \( f_1 = f_0 \), and neither \( \sigma_0 \) nor \( \sigma_1 \) is empty. If \( d' \) protects \( \psi \), then \( \sigma_1 \) is safe, and so \( l_1 \geq f_1 \). Thus \( l_1 \geq f_0 \), establishing (1). Conclusion (2) is trivial if \( d \approx d' \), so we now assume \( d \not\approx d' \) with respect to \( \psi \). Then there exists \( t \in (0,1] \) such that \( \sigma_t \) is not a cut. Let \( s \) be the smallest such value, and consider the interval \( T = [0,s] \). Since \( d = b_0 \) does not protect \( \psi \), the claim implies \( l_0 < l_s \). Now because the function \( t \mapsto l_t \) is convex, it has at most one local minimum in \([0,1]\). Because \( l_0 < l_s \), the minimum value of \( l_t \) must occur in the interval \((\infty,s)\). Hence \( l_t \) is nondecreasing on \([s,1] \), and we have \( l_1 \geq l_s \geq f_0 \). This proves conclusion (1), because \( l_1 \) is \( \text{cap}(\psi(d')) \) and \( f_0 \) is \( \text{flow}(\psi(d)) \). Now we prove (2) by showing that \( d' \) protects \( \psi \). If \( \sigma_t \) is a cut, let \( \beta \) be the largest value such that \( \sigma_{\beta} \) is not a cut. (One must exist, for we are assuming \( d \not\approx d' \).) Applying the claim to the interval \( T = (\beta,1] \), we find that \( b_1 \) protects \( \psi \) because \( l_1 \geq l_{\beta} \). Since \( b_1 = d' \), this proves statement (2). □

Body of the correctness proof

Lemma 9e.3 provides us with the following lemma, our main tool for proving Theorem 9e.1. We shall use this lemma frequently.

**Lemma 9e.4.** (Potential Cut Lemma) Suppose \( 1 \leq k \leq m \), and let \( d \) and \( d' \) be configurations in \( A_{k-1} \).

1. If \( d' \) protects \( \psi_k \) but \( d \) does not, then \( \text{cap}(\psi_k(d')) \geq \text{flow}(\psi_k(d)) \).
2. If neither \( d \) nor \( d' \) protects \( \psi_k \), then \( d \approx d' \) with respect to \( \psi_k \).

Statement (2) implies that any two configurations \( d,d' \in A_{k-1} \) that fail to protect \( \psi_k \) must satisfy \( \text{flow}(\psi_k(d)) = \text{flow}(\psi_k(d')) \). Thus Lemma 9e.4 shows that the sets \( U_k \) defined in line 4 of Algorithm A are uniquely determined.

The proof of Lemma 9e.4 depends on several facts about the set \( A_{k-1} \). In particular, the lemma makes no sense unless \( A_{k-1} \) is well defined. On the other hand, \( A_k \) is well defined only if the Potential Cut Lemma holds for \( A_{k-1} \). We must therefore prove Lemma 9e.4 in parallel with the following claim.

**Lemma 9e.5.** For \( 1 \leq k \leq m \), the following statements hold:

3. If \( \mu(P_k) = \mu(Q_k) \), then every configuration \( c \in A_{k-1} \) protects \( \psi_k \).
(4) The set \( A_k \) is well defined by Algorithm A.
(5) The point \( 0 \) lies in \( A_k \).
(6) Every configuration in \( A_k \) protects the potential cuts \( \psi_1 \) through \( \psi_k \).

**Proof of Lemmas 9e.4 and 9e.5.** The proof proceeds by induction on \( k \), with the inductive hypothesis being the conjunction of (4) through (5). A basis for this hypothesis is easily established at \( k = 0 \): the set \( A_0 \) is obviously well defined, \( 0 \in A_0 \) by definition, and condition (6) is vacuously true. So assume \( k \geq 1 \). The key step is the proof of (1) and (2), in Lemma 9e.4, from the inductive hypothesis.

(1,2) We apply Lemma 9e.3 to the configurations \( d \) and \( d' \) and the potential cut \( \psi_k \). The convexity property implies that the function \( b \mapsto \text{cap}(\psi(b)) \) is convex, and hence \( t \mapsto \text{cap}(\psi(b_t)) \) is convex. And since \( A_{k-1} \) is a convex set, the inductive hypothesis implies that every configuration \( c \in L \) protects the potential cuts \( \psi_1 \) through \( \psi_{k-1} \). This fact, combined with the ordering property, demonstrates the final assumption of Lemma 9e.3. The conclusion of that lemma is identical to the conclusion of Lemma 9e.4.

(3) Suppose \( \mu(P_k) = \mu(Q_k) \), and apply parts (1) and (2) to \( \psi_k \) with \( 0 \) in place of \( d' \) and \( c \) in place of \( d \). Since \( S(0) \) is routable, the routability property implies that \( 0 \) protects \( \psi_k \). Hence only part (1) can apply; it says that \( \text{cap}(\psi_k(0)) \geq \text{flow}(\psi_k(c)) \) if \( c \) fails to protect \( \psi_k \). But our assumption that \( \mu(P_k) = \mu(Q_k) \) implies that the capacity of \( \psi_k \) is independent of configuration. Therefore \( \text{cap}(\psi_k(c)) \geq \text{flow}(\psi_k(c)) \), and so \( \psi_k(c) \) cannot be unsafe. Therefore \( c \) protects \( \psi_k \).

(4) For \( A_k \) to be well defined, the set \( U_k \) defined in line 4 of Algorithm A must have the specific form \( \{ d \in \mathbb{C}(S) : \Delta^- < \Delta_{P_kQ_k}(d) < \Delta^+ \} \), for some \( \Delta^- \) and \( \Delta^+ \). Recall that \( U_k \) includes a point \( d \) if and only if the capacity \( \text{cap}(\psi_k(d)) \) of \( \psi_k(d) \) is less than the constant \( f = \text{flow}(\psi_k(c)) \). But by the definition of a potential cut, \( \psi_k(d) \) depends only on \( \Delta_{P_kQ_k}(d) \). Hence it suffices to show that the set

\[
\{ \Delta_{P_kQ_k}(d) : d \in \mathbb{R}^n \text{ and } \text{cap}(\psi_k(d)) < f \}
\]

is a nonempty open interval \( (\Delta^-, \Delta^+) \). By part (3), line 4 is only reached if \( \mu(P_k) \neq \mu(Q_k) \). Hence we may choose a line \( L \) through \( c \) on which \( \Delta_{P_kQ_k}(d) \) is not constant. The convexity property of \( \psi_k \) implies that the set \( \{ d \in L : \text{cap}(\psi_k(d)) < f \} \) is an open interval of \( L \); it is nonempty because it contains \( c \). Since \( \Delta_{P_kQ_k}(d) \) is a nonconstant linear function on \( L \), the set

\[
\{ \Delta_{P_kQ_k}(d) : d \in L \text{ and } \text{cap}(\psi_k(d)) < f \}
\]

is also a nonempty open interval. This is enough, because every value \( \Delta_{P_kQ_k}(d) \) is represented by some \( d \in L \).
(5) By the induction hypothesis, \(0 \in A_{k-1}\). If every \(c \in A_{k-1}\) protects \(\psi_k\), then \(0 \in A_k\) trivially. Otherwise, apply (1) to \(0\) and \(c\). (Because \(S(0)\) is routable, \(0\) protects \(\psi_k\) by the routability property.) So \(\text{cap}(\psi_k(0)) \geq \text{flow}(\psi_k(c))\), whence \(0 \notin U_k\). Because \(\Delta_{P_k} Q_k(0) = 0\), by definition, we have \(0 \notin (\Delta^-, \Delta^+)\). Thus \(0\) satisfies the constraint \(\Lambda_k\) defined at line 6, and so \(0 \in A_k\).

(6) Since \(A_k \subseteq A_{k-1}\), every configuration \(d \in A_k\) protects \(\psi_1\) through \(\psi_{k-1}\), by the induction hypothesis; it remains to show that every configuration \(d \in A_k\) protects \(\psi_k\). Suppose that \(d \in A_{k-1}\) fails to protect \(\psi_k\). Then \(U_k\) is nonempty, and is defined in terms of some configuration \(c\). By part (2), \(d \approx c\) with respect to \(\psi_k\), and in particular \(\text{flow}(\psi_k(d)) = \text{flow}(\psi_k(c))\). Because \(d\) does not protect \(\psi_k\), certainly \(\text{cap}(\psi_k(d)) < \text{flow}(\psi_k(d))\), and it follows that \(d \in U_k\). But the constraint \(\Lambda_k\) excludes all members of \(U_k\) from \(A_k\). Therefore \(d \notin A_k\). □

From the above lemma, most of Theorem 9e.1 follows quickly. First of all, the initial configuration \(0\) is a member of \(A_m\) by claim (5). Second, if \(d \in A_m\), then for all \(t \in [0, 1]\), the configuration \(td\) lies in \(A_m\), and hence protects every \(\psi \in \Psi(S)\) by claim (6). Therefore by the routability property, \(S(d)\) is routable for all \(d \in A_m\). It remains to argue that \(A_m\) is a single connected component of \(\{d \in \mathcal{C}(S) : S(d)\ \text{is routable}\}\). To do so, we make use of an elementary topological result. A subset \(X\) of a topological space is said to surround another subset \(Y\) if \(Y\) lies in the interior of \(X\), and the closure of \(Y\) is contained in \(X\). If \(X\) surrounds the nonempty set \(Y\), then \(Y\) is a connected component of the complement of \(X - Y\).

**Lemma 9e.6.** For \(0 \leq k \leq m\), the set \(A_m\) is surrounded by the region

\[
X_k = A_k \cup \left( \bigcup_{i=1}^{k} A_{k-1} \cap U_k \right).
\]

**Proof.** It suffices to show that \(A_m\) is closed and \(X_k\) is open, because clearly \(A_m \subseteq X_k\). First the former. By the boundary property, the configurations that protect all the potential cuts \(\Psi(S)\) lie within a closed subset \(D\) of \(\mathcal{C}(S)\). By claim (6) of Lemma 9e.5, all points of \(A_m\) protect every \(\psi \in \Psi(S)\). Therefore \(A_m\) is the intersection of \(D\) with the set of configurations that satisfy the inequalities \(\Lambda_k\). Each configuration \(\Lambda_k\) defines a closed subset of \(\mathbb{R}^n\). Therefore \(A_m\) is closed.

Now we prove by induction on \(k\) that \(X_k\) is open. The basis case, \(X_0 = A_0\), is guaranteed by the boundary property. Let \(k > 0\), and consider the nontrivial case when \(U_k\) is nonempty. From the definition of \(X_k\) we derive \(X_k = (X_{k-1} - A_{k-1}) \cup A_k \cup (A_{k-1} - U_k)\), which reduces to \(X_{k-1} - (A_{k-1} - U_k - A_k)\). The set \(B = A_{k-1} - U_k - A_k\) is the intersection of \(A_{k-1}\) with one of the closed half-spaces forming the complement of \(U_k\); it remains to show that \(B\) is closed in \(X_{k-1}\). But \(A_{k-1}\) is just the subset of \(X_{k-1}\) satisfying the constraints \(\Lambda_i\), for all \(i < k\), so \(B\) is
$X_{k-1}$ intersected with finitely many closed half-spaces. Therefore $X_k = X_{k-1} - B$ is open. □

Setting $k = m$ in Lemma 9e.6, we find that $\bigcup_{i=1}^{m} (A_i \cap U_i)$ disconnects $A_m$ from the rest of $R^n$. Hence the connected component of $\{ d \in C(S) : S(d) \text{ is routable} \}$ that contains $A_m$ cannot be a proper superset of $A_m$, unless it also contains a point in $A_{i-1} \cap U_i$ for some $i$. But if $d \in A_{i-1}$ corresponds to a routable sketch, then (by the routability property) it protects $\psi_i$, and statement (1) of the Potential Cut Lemma applies to $d$ and the configuration $c \in A_{i-1}$ used to define $U_i$. It shows that $\text{cap}(\psi_i(d)) \geq \text{flow}(\psi_i(c))$, which means that $d \notin U_i$. Therefore $d \in A_{i-1} \cap U_i$ implies that $S(d)$ is not routable. So $A_m$ is precisely equal to the component of $\{ d \in C(S) : S(d) \text{ is routable} \}$ that contains 0. This completes the proof of Theorem 9e.1.

9F. Implementing the Abstract Algorithm

In this section, we build upon the results of Sections 9D and 9E to prove the correctness of Algorithm C, the concrete compaction algorithm. The hard part of the proof is over: Algorithm A, which is an abstract description of the compaction algorithm, is proven correct by Theorem 9e.1 of the previous section. It remains to show that Algorithm C is just a special case of Algorithm A. There are two steps to this process: first, to identify the potential cuts that Algorithm C uses, and show that they satisfy the preconditions of Algorithm A; and second, to prove an explicit correspondence between the quantities computed by the two algorithms. The correctness of the compaction algorithm will then follow from the correctness of its subroutines (Algorithms F and R) along with Theorem 9e.1.

Preconditions of Algorithm A

Our first task is to show that the potential cuts used by Algorithm C satisfy the requirements of Algorithm A, namely the routability, capacity, ordering, and boundary properties. The potential cuts in question are of three types.

1. Horizontal potential cuts $\phi_{pq}$ where either $p$ or $q$ is a feature endpoint.

2. Diagonal potential cuts $\phi_{pq}$ where $p$ and $q$ are feature endpoints.

3. Critical potential cuts $\chi_{pQ}$ where $p$ is a feature endpoint.

Let $S$ denote the sketch input to Algorithm C, and let $\Psi(S)$ contain all the potential cuts for $S$ of types 1–3. We number these cuts $\psi_1, \ldots, \psi_m$ in the order that Algorithm C examines them. Since Algorithm C considers horizontal potential cuts first, the cuts of type (1) are $\psi_1, \ldots, \psi_h$ for some $h$. Next come the potential cuts of type 2 in order of height, and finally the potential cuts of type 3 in order of height.
We treat Algorithm C as if it processed all the potential cuts in $\Psi(S)$, although it actually ignores those that connect features in the same module. Part (3) of Lemma 9e.5 says that such potential cuts generate no constraints; hence Algorithm C is justified in ignoring them.

**Proposition 9f.1.** The sequence $\Psi(S)$ has the routability, convexity, ordering, and boundary properties.

**Proof.** The routability property is easiest. If a configuration $d \in C(S)$ fails to protect some potential cut $\psi \in \Psi(S)$, then $\psi(d)$ is unsafe and nonempty in the sketch $S(d)$. By Proposition 8b.3, then, $S(d)$ is unroutable. On the other hand, if every configuration $td$ with $t \in [0, 1]$ protects every potential cut in $\Psi(S)$, then in particular $d$ protects all the critical potential cuts of $S$. By Theorem 9c.2, the sketch $S(d)$ is therefore routable.

To check the convexity property for a potential cut $\psi \in \Psi(S)$, it is enough to show that the function $d \mapsto \|\psi(d)\|$ is convex on $C(S)$. Let $d_0, d_1$ be arbitrary configurations in $C(S)$, and for $t \in [0, 1]$ define $d_t = (1 - t)d_0 + td_1$. Say $\psi$ connects the features $P$ and $Q$. For each $t$ we have $\psi(d_t)(0) = p_t(d_t)$ and $\psi(d_t)(1) = q_t(d_t)$ for some $p_t \in P$ and $q_t \in Q$. Put $l_t = \|q_t(d_t) - p_t(d_t)\| = \|\psi(d_t)\|$. We must show that $l_t \leq (1 - t)l_0 + tl_1$. If $\psi = \phi_{pq}$ for some $p$ and $q$, then $q_t = q$ and $p_t = p$ for all $t \in [0, 1]$. Consequently the vector $q_t(d_t) - p_t(d_t)$ changes linearly with $t$, and so the convexity of $\|\cdot\|$ implies that $l_t \leq (1 - t)l_0 + tl_1$. Now suppose $\psi = \chi_{pQ}$ for some feature $Q$ and feature endpoint $p$. Then $p_t = p$ for all $t$, and $q_t$ has the property that for all $q \in Q$,

$$\|q_t(d_t) - p(d_t)\| \geq \|q(d_t) - p(d_t)\|. \tag{9-2}$$

Because $Q$ is a convex set, we may choose $q = (1 - t)q_0 + tq_1$. Then $q_t(d_t)$ is linear in $t$ and equals $q_0(d_0)$ or $q_1(d_1)$ if $t$ is 0 or 1, respectively. Of course, $p(d_t)$ is also linear in $t$. Hence by the convexity of $\|\cdot\|$, the right-hand side of (9-2) is at most $(1 - t)l_0 + tl_1$. The left-hand side of (9-2) is just $l_t$, so the length of $\chi_{pQ}$ is a convex function. Therefore $\Psi(S)$ has the convexity property.

Now we argue that the sequence $\Psi(S) = \langle \psi_1, \ldots, \psi_m \rangle$ has the ordering property. Let the configuration $d \in C(S)$ protect $\psi_i$ for all $i < k$, and suppose $d \in Fr\{c \in C(S) : \psi(c) \text{ is a cut} \}$. We show that every cut that is a subpath of $\psi_k(d)$ is either $\psi_i(d)$, for some $i < k$, or its reverse. Since $d$ protects $\psi_i$, such cuts are either safe or empty. For $d$ to lie in $Fr\{c \in C(S) : \psi_k(c) \text{ is a cut} \}$ means that the features interrupting $\psi_k(d)$ must do so at their endpoints, and furthermore that $\psi_k(d)$ is not horizontal. If $\psi_k = \phi_{pq}$ for some feature endpoints $p$ and $q$, then every cut that is a subpath of $\psi_k(d)$ begins and ends at feature endpoints, and has smaller height than $\psi_k$. All such cuts appear in the list $\langle \psi_1, \ldots, \psi_{k-1} \rangle$. The other case is only slightly harder. Suppose $\psi_k = \chi_{pQ}$ for some feature $Q$ and feature endpoint $p$. Let
α be a subcut of \( \psi_k(d) \) that ends on \( Q(d) \), if one exists. Then all cuts that are subpaths of \( \psi_k(d) \), except possibly \( α \) and \( ̂α \), are cuts between feature endpoints; they have the form \( \psi_i(d) \) for some \( i < k \). If \( α \) exists, it is a critical cut from the feature endpoint \( α(0) \) to \( Q(d) \), and has the form \( χ_{rQ}(d) \) where \( α(0) = r(d) \). As noted in Section 9C, the height of \( χ_{pQ} \) exceeds that of \( χ_{rQ} \), and hence \( χ_{rQ} \) appears in \( \{ ψ_1, \ldots, ψ_{k-1} \} \).

To check the boundary property, we must exhibit a closed set \( D \subseteq R^n \) such that all configurations in \( C(S) - D \) fail to protect some potential cut in \( ψ \). (That \( C(S) \) is open follows directly from its definition.) Let \( w \) denote the minimum of the widths of the elements of \( S \). The space \( C(S) \) was defined as the set of configurations \( d \) such that for all points \( p \) and \( q \) of \( S \) with \( p_x = q_x \) and \( p_z < q_z \), we have \( Δ_{pq}(d) > 0 \). We may assume that \( μ(p) \neq μ(q) \). Define \( D \) the same way, but replace the condition \( Δ_{pq}(d) > 0 \) by the constraint \( Δ_{pq}(d) ≥ w \). Clearly \( D \) is closed in \( R^n \). And if \( d \) is a configuration in \( C(d) - D \), then there are two features in separate modules of \( S(d) \) whose separation is less than \( w \). Choose features \( P \) and \( Q \) such that the horizontal separation between \( P(d) \) and \( Q(d) \) is minimal. The minimum separation is realized at a feature endpoint, so there are points \( p ∈ P \) and \( q ∈ Q \) such that \( φ_{pq} ∈ Ψ(S) \) and \( ||φ_{pq}(d)|| < w \). By the choice of \( P \) and \( Q \), no features intervene between \( p(d) \) and \( q(d) \), and hence \( φ_{pq}(d) \) is a cut. It is nonempty because it connects different islands, and is unsafe because its capacity is negative. Thus \( d \) fails to protect the potential cut \( φ_{pq} ∈ Ψ(S) \). □

Correspondence between the algorithms

The final phase of our proof strategy involves showing that the constraints computed by the concrete algorithm define the same space as the constraints \( Λ_k \) defined abstractly. This fact will imply that the compaction algorithm searches precisely the set \( A_m \) of acceptable configurations, and correctness will follow quickly. In order to state the correspondence, let \( C_0 \) denote the set of configurations satisfying the constraint system \( I \) defined at line 2 of Algorithm \( C \), and let \( C_k \) denote those configurations satisfying \( I \) after the \( k \)th iteration on the loop in lines 3–6.

Lemma 9f.2. For all \( k \) satisfying \( h ≤ k ≤ m \), the sets \( C_{k-h} \) and \( A_k \) are identical.

Proof. Recall that \( h \) is the number of horizontal cuts in the sequence \( Ψ(S) \). We prove the lemma by induction on \( k \), the basis case being \( k = h \). Any configuration in \( A_h \) is in \( C(S) \), because \( A_h ⊆ A_0 \), and also protects the horizontal potential cuts, according to part (6) of Lemma 9e.5. Therefore \( A_h ⊆ C_0 \). On the other hand, you may check that when the constraint \( Λ_k \) exists, for \( k ≤ h \), it corresponds to the potential cut in \( I_0 \) induced by \( ψ_k \). (Here we use Proposition 9d.5, which establishes the correctness of Algorithm \( F \).) Therefore \( C_0 ⊆ A_h \).
For the inductive step, suppose that $C_{k-h-1} = A_{k-1}$. We first draw a correspondence between the configurations $c$ found by Algorithms A and C. The key observation is that the configuration $c$ found by Algorithm C at line 4 minimizes the capacity $\text{cap}(\psi_k(c))$ over all $t \in C_{k-h-1} = A_{k-1}$. (Dijkstra's algorithm is applicable here, because according to Lemma 9.4.5, the initial configuration 0 satisfies the constraint system.) We wish to argue that if any $d \in A_{k-1}$ fails to protect $\psi_k$, then neither does $c$. Suppose to the contrary that $c$ protects $\psi_k$ but $d \in A_{k-1}$ does not. Then by the Potential Cut Lemma (9.4.4), statement (1), we have $\text{cap}(\psi_k(c)) \geq \text{flow}(\psi_k(d))$. But $\text{cap}(\psi_k(c)) \leq \text{cap}(\psi_k(d))$ by the choice of $c$, so $\text{cap}(\psi_k(d)) \geq \text{flow}(\psi_k(d))$, and $d$ protects $\psi_k$ after all. Thus line 7 of Algorithm C correctly implements line 3 of Algorithm A.

There are now two cases to consider. If the configuration $c$ does protect $\psi_k$, then so do all configurations in $A_{k-1}$. Therefore Algorithm A sets $A_k$ to $A_{k-1}$, and Algorithm C does not change $I$, so we have $C_{k-h} = A_k$ as desired. On the other hand, if $c$ does not protect $\psi_k$, then Algorithm C adds to $I$ the constraint derived from $\psi$ in the configuration $c$. This constraint is precisely $A_k$. ☐

We conclude that the configurations that obey the final constraint system $I$ in Algorithm C are precisely those in $A_m$. (If the design system adds extra constraints to $I$, some configurations in $A_m$ may be excluded.) Theorem 9.4.1, which characterizes $A_m$, now implies that every configuration obeying $I$ is routable, and that the constraints $I$ are optimal, unless the constraints are allowed to define a disconnected region of configuration space. Finally, line 10 of Algorithm C finds an optimal configuration obeying the constraint system $I$. The resulting sketch is guaranteed to be routable, and hence Algorithm R can regenerate the layout. This completes the proof that the compaction algorithm is correct.

Optimizations of Algorithm C

Both the time and space performance of Algorithm C can be improved by reducing the size of the adjacency graph. One therefore wishes to choose gates in such a way as to minimize the number of crossings between traces and gates. Although we required the gates to form a partition of the sketch, one can get by with fewer. If the routing region is connected, a minimal set of gates is such that the set of points in the routing region but not on any gate is simply connected. Equivalently, if islands and gates are considered as the nodes and arcs, respectively, of a graph, then this graph should be a tree. One must be careful, however, to keep track of the direction of every crossing among the traces, gates, and terminals. The removable nodes and edges of the intersection graph depend upon these directions of crossing in a somewhat complicated manner. In essence, one must ensure that when modifying the intersection graph, the traces can be rerouted to reflect the new structure.
A minimum-cost spanning tree algorithm can be used to find a set of gates that
cross as few traces as possible. Every horizontal cut between different islands is a
potential gate, but we may restrict our attention to horizontal cuts that are incident
on feature endpoints. There are at most $O(|F|)$ such cuts, and they can be thought
of as the arcs of a graph $H$ over the islands. The cost of a cut will be the number of
crossings of the cut by traces in the original sketch; costs can be computed efficiently
using a scanning algorithm as in Section 1D. The gates are chosen to be the arcs in
a minimum-cost spanning tree of the graph $H$.

Another way to speed up Algorithm C is to ignore potential cuts that cannot
generate constraints. For example, if a potential cut $\phi_{pq}$ has minimal capacity in
the initial configuration, it cannot generate a constraint. This observation follows
from statement (1) of the Potential Cut Lemma. More generally, if a potential cut
is occluded in such a way that it cannot become a cut before reaching a minimum
of capacity, then this potential cut may be ignored. Lemma 9e.4 (or more generally,
Lemma 9e.3) can be applied in many other ways to justify the omission of potential
cuts. For example, I showed in [29] that if the wiring norm is rectilinear—that is,
if $\|(x, y)\| = \max\{|x|, |y|\}$—and the features are all horizontal or vertical, then the
critical potential cuts may be omitted altogether.

None of these improvements affect the fact that Algorithm C requires $\Omega(|F|^3)$
time, not just in the worst case, but in almost every case. To reduce this amount,
one must avoid considering most of the potential cuts. Most constraints in practice
are likely to be local, so one can try to ignore all potential cuts of sufficiently large
height. If one solves the constraint system before evaluating all the potential cuts,
and the routing algorithm succeeds, then compaction may be terminated. If the
routing algorithm fails, more potential cuts must be considered. A good heuristic
for exploiting locality could reduce the average-case running time to quadratic or
less, though the leading constant might be large.

Ultimately, the slowness of Algorithm C is due to its generality. The islands in
a sketch compaction problem allows may be bound into modules in an arbitrary
way, whereas in many cases of interest only local connections are needed. When all
features are independent, as usually occurs in the compaction of routing channels,
simpler and faster techniques are available that still insert all useful jogs automatica-
ly [59].

Wire length minimization

Usually when performing compaction one would like to improve wire lengths as
well as layout area. Algorithm C minimizes trace lengths in a trivial sense, namely
that the wires make no unnecessary detours. Because it uses Algorithm R, the
lengths of traces are minimal given the positions of the features that it supplies. By
default Algorithm C moves each module as close to the left-hand wall as possible, which will often be far from optimal. But it can be modified to support whatever wire-length minimization technique you favor. The constraint graph constructed by Algorithm C specifies the set of acceptable output configurations. Solving the constraint system with a longest-path algorithm determine the minimum separation between the walls. One can add the constraint that the walls be separated by that distance, thus defining a smaller set of acceptable configurations. One may then choose a configuration in this set by any desired means. If one can estimate the effects of configuration on total wire length, then one can find a configuration that nearly minimizes wire length. The problem of finding a good heuristic for making this estimate is open, but probably not too difficult.

Summary

I have presented a polynomial-time algorithm for one-dimensional layout compaction with automatic jog insertion. It works whenever layouts can be partitioned into layers such that wires on two different layers interact only via modules that are present on both. The algorithm takes its input as a set of proper sketches, one for each routing layer, and produces output in the same form. (For practical purposes, this means the input must be a legal layout.) Algorithm C treats the special case of one routing layer, which is no easier to compact than many connected layers. Jog insertion is achieved by treating wires not as objects to be moved, but only as indicators of layout topology. Using the sketch routability theorem, the algorithm converts the wires into constraints on module positions that ensure that the wires have sufficient room to be routed. Having determined a new placement for the modules, it then invokes a single-layer router (Algorithm R) to restore the wires with as many jogs as necessary. The compactor thereby inserts all jogs that help to reduce the width of the layout. It may use more jogs than necessary, however.

The version of Algorithm C presented here substantially generalizes the compaction algorithms in my earlier papers [28, 29]. Those algorithms worked only in a grid-based wiring model, while Algorithm C allows features and traces to be other than rectilinear, to have different widths, and to be governed by an arbitrary piecewise linear wiring norm. These extensions were made possible, of course, by the theory of single-layer wire routing established in Chapters 1 and 8. Further extensions or reformulations of this theory, as we will discuss in the next chapter, should lead to further generalizations of Algorithm C. Unlike the reasoning that underlies Algorithm R, the correctness proof of Algorithm C is nearly modular. Having the requirements of Algorithm A spelled out in Section 9E means that changes to Algorithm C can be easily justified.
Chapter 10

Extensions of the Theorems and Algorithms

This chapter is all about the sketch model: the rationale behind it, the extensions and modifications it supports, how the sketch algorithms can be adapted to handle these extensions, and how sketches may be used to represent circuit layers. For the most part the proposed changes to the sketch model are orthogonal, meaning that they can be adopted or ignored independently. Since many facts about the sketch model will be illustrated by reference to the design model, in this chapter I use the terms ‘wire’ and ‘trace’ interchangeably.

Chapter outline

The chapter is divided into four major sections. The first chiefly concerns the representation of standard devices as parts of sketches. It presents a view of separation constraints based on the convolution of geometric regions, and relates it to the use of wiring norms to define which sketches are proper. It suggests how the sketch model may be modified so that separation constraints can be defined independently for all pairs of sketch elements. Finally, it describes how to change the Algorithms T and R so that the terminals of each trace are permitted to approach one another. Both these extensions are helpful for representing integrated circuit layers.

Section 10B examines the aspects of the sketch model that govern the shapes of traces: the wiring norm, the allowed shapes of features, and the fact that traces are not constrained to a grid. It first shows that the sketch model does, in fact, subsume the grid-based wiring model. If all the features of a sketch lie in a grid of unit pitch, measured in a rectilinear wiring norm, and all the elements of the sketch have width 1, then the traces may be constrained to the grid without affecting routability. Moreover, one can add a simple postprocessing phase to Algorithm R to ensure that every trace is routed within the grid.

Section 10B then considers wiring rules at the opposite extreme: curvilinear rules in which the wiring norm is not piecewise linear. The theory of single-layer routing does not change substantially if the wiring norm is, say, the euclidean norm, and if circular arcs are allowed as features. Even without appealing to a more
general theory of wire routing, we can show that the design routability theorem holds also for curvilinear wiring norms, provided that fringes remain polygonal. Hence the sketch routability theorem admits the same generalization. The trick we use involves approximating the curvilinear norm by a polygonal norm. It thereby allows us to apply Algorithm R to sketches with curvilinear wiring norms, although its performance declines and it cannot quite minimize wire lengths.

Section 10C steps farther out and considers some major extensions of the sketch model and the sketch problems. These include: allowing the terminals of a trace to merge or pass through one another during compaction; allowing terminals to be line segments or convex polygons as well as points; and allowing traces to have more than two terminals. My conclusion is that although the extensions seem to be possible, the sketch model is not well suited to them, particularly the addition of extended terminals and multiterminal nets. In Section 10D I propose a new model of wiring that incorporates extended terminals and multiterminal nets in an elegant way. I then discuss the prospects for adapting my theory of single-layer wire routing and its attendant algorithms to the new model.

Development of the sketch model

Before describing various generalizations of sketches, I should explain some of the reasons why the sketch model has the properties it does. My advisor Prof. Leiserson and I originated the sketch model as a generalization of the grid-based wiring model used Leiserson and Pinter [22] and many others. We wished to consider wiring rules more general than grid models, and so we quickly abandoned the common convention that terminals are points on the boundaries of modules. Instead we decided to separate terminals from the modules they helped interconnect. The reason was to avoid introducing spurious cuts that might falsely indicate unroutability; see Figure 10-1 for an example. The desire for a clean routability theorem was the major motivation for most of my decisions concerning the sketch model.

![Figure 10-1](image)

*Figure 10-1. Terminals are not points on other features.* If they were, some cuts in a routable sketch could be both nonempty and unsafe. Here the traces have width equal to the distance between adjacent dotted lines, and the unit polygon is square. The striped cut has a congestion of 5 but a capacity of only 4.5. Yet the traces can be routed.
Another peculiarity of sketches is their requirement that each trace be no wider than its terminals. Others have made such an assumption to simplify design-rule checking [37]. My reason comes from the sketch routability theorem. If the requirement were removed, this theorem would be false. Figure 10-2 shows the counterexample. The breakdown in the proof can be traced to Lemma 4f.3, which shows that the capacity of a major cut is no less than the capacity of its elastic chain. This lemma is used to prove Corollary 4f.5: that a safe sketch, whose major straight cuts are safe, has no unsafe, major, bent cuts. In Figure 10-2 this claim fails: the bent cut is unsafe, but the links of its elastic chain are safe. The reason is that in going from the bent cut to its elastic chain, the flow has decreased by the width of the wire, but the capacity has only decreased by the width of the its terminal, which in this example is smaller.

Self-avoidance

Perhaps the most puzzling aspect of the sketch and design models is the requirement that wires be self-avoiding. I am frequently asked why this condition exists, and whether for practical purposes it could be ignored. Unfortunately there are good reasons for being concerned with self-avoidance. The question is nevertheless a good one, because it seems that whenever a wire fails to be self-avoiding, a simple change to the topology would remove the offending loop of wire, improving the routing and avoiding design-rule violations. This hope is dashed by examples like Figure 10-3, which shows that two parts of a wire can approach too closely in a gap that is too narrow for the wire to be routed through. Programmers of computer-aided design tools have assured me that such situations should not be assumed to be absent in practical designs.

The first reason to require self-avoidance of wires is that the sketch and routing theorems depend upon it. Without it, the sketch of Figure 10-3 would be proper without being safe. A possible escape from this dilemma is to redefine the flow across a cut so that consecutive necessary crossings of a cut by the same wire contribute only one wire's thickness to the congestion. In order for this approach to work, one would have to allow wires to intersect themselves. Such a change would almost certainly cause more problems than it solved.
In any case, there are eminently practical reasons to insist that wires be self-avoiding. One can make a case for an even stronger condition. Let us call a wire strongly self-avoiding if the union of its territory with those of its terminals does not separate the plane into two or more components. (Ordinary self-avoidance requires only that this union not separate one island of the sketch from another.) An equivalent definition is that a wire is strongly self-avoiding if its extent is simply connected. If a wire is not strongly self-avoiding, then two parts of the wire violate the design rule concerning wire-to-wire separation. This raises the possibility that a short between these parts would form a loop in the layout, which (I am told) could function as an unwanted antenna; or that while the wires are being laid down, the thin piece of resist between the nearby wire parts could break off and foul the circuit. For these reasons, some self-avoidance condition must be imposed on wires, and possibly on routing obstacles as well.

The property on which my definition of self-avoidance is based, namely divisiveness of design articles, has the benefit that it can be tested by looking at nondegenerate straight cuts. (See Lemmas 5e.4 and 6a.2.) In contrast, there is no natural set of cuts whose safety determines whether the extent of an article separates the plane. Another benefit of my definition is that the self-avoidance of ideal embeddings of wires and ideal realizations of traces is relatively easy to verify. Ideal embeddings and ideal realizations are strongly self-avoiding, but the proof is fairly difficult.

10A. Representation Issues

As it stands, the sketch model is not very close to the models that circuit designers actually use. Although grid models, which the sketch model subsumes, are acceptable for channel routing problems, they are poorly suited for representing transistors, the primary components of integrated circuits other than wires. Problems arise when trying to map the geometric design rules onto the sketch model. Usually the rules are separation constraints and overlap constraints among regions on various circuit layers. Some of the regions have no natural counterparts in the
sketch model, and some regions must be represented as a set of sketch elements if that region needs to connect to wires. Nevertheless, with suitable extensions to the sketch model described here and in later sections, one can obtain approximations of real design rules. Though sketches cannot adequately represent most transistor structures, they can probably handle the interconnection of larger modules.

Convolution of regions

The rules governing proper sketches and designs are stated in terms of a global wiring norm. This approach has the virtue of simplicity, but it grew out of a more basic and more flexible view of geometric design rules, which I now describe. It begins with the assumption that wires, at least, are to be represented as paths, rather than regions of positive area, in order to define homotopy relations among wires. Hence we must relate the abstract wire, which we think of as a path or its image, to the region that the wire is to occupy in the circuit. We must also convert the design rules among these regions to design rules among the abstract wires.

A natural approach is to define the regions that wires occupy, and the regions that are forbidden to occupy, using the operation of convolution. For the purposes of this section, a region is a subset of the plane \( R^2 \). The convolution of two regions \( A \) and \( B \), which I denote \( A + B \), is the set of all vector sums of points in \( A \) with points in \( B \), namely,

\[
A + B = \{ a + b : a \in A, b \in B \}.
\]

We consider wires whose shape can be described as the convolution of a centerline, call it \( C \), with a region \( W \) that contains the origin 0, as shown in figure 10a-1. The region \( C + W \) occupied by the wire can be obtained by sweeping the brush \( W \) along the centerline \( C \), keeping the origin of \( W \) on \( C \). The required separation between wires may also be described using convolution. If \( R_1 \) and \( R_2 \) are the regions occupied by two wires, there may be a region \( S_{12} \) such that \( R_1 \) and \( R_2 \) are sufficiently separated if and only if \( R_1 + S_{12} \) does not intersect \( R_2 \). This kind of design rule is quite general. Each wire can have a different brush, and each pair of wires can have a different region defining their required separation. Self-avoidance can also be described using convolution; there may be regions \( S_{11} \) and \( S_{22} \) such that \( R_1 + S_{11} \) and \( R_2 + S_{22} \) are not allowed to divide the plane.

The convolution conditions become somewhat simpler if we assume that the regions \( W_i \) and \( S_{ij} \) have inversion symmetry. If \( B \) is any region, we define the set \(-B\) to be \( \{ -b : b \in B \} \). Suppose \( W_i = -W_i \) and \( S_{ij} = -S_{ij} \) for all \( i \) and \( j \). Then the conditions

\[
((C_1 + W_1) + S_{12}) \cap (C_2 + W_2) = \emptyset \quad \text{and} \quad (C_1 + W_1) \cap ((C_2 + W_2) + S_{12}) = \emptyset
\]
are equivalent, and are also equivalent to the condition

\[(C_1 + T_{12}) \cap C_2 = \emptyset \quad \text{where} \quad T_{12} = W_1 + S_{12} + W_2.\]

(Note that convolution is associative and commutative.) So the brushes \(W_i\) and separation regions \(S_{ij}\) may be discarded in favor of the regions \(T_{ij}\).

One can take the convolution idea a step further and consider the constraints that arise among the wires crossing a cut in a sketch. Let \(\overline{pq}\) be a cut between two obstacles \(P\) and \(Q\), which for simplicity we consider to be points. As we saw in Chapter 8, there is a definite sequence of traces that must cross \(\overline{pq}\); they have a definite ordering from \(P\) to \(Q\). Let \(C_1, \ldots, C_n\) denote the centerlines (i.e., images) of these traces, and put \(C_0 = P\) and \(C_{n+1} = Q\). For \(1 \leq i \leq n\), let \(T_i\) be the region defining the required separation between \(C_i\) and \(C_{i+1}\). If the \(i\)th and \((i+1)\)st traces are the same, then \(T_i\) determines the self-avoidance requirement for that trace. The centerline of the first trace must satisfy \((P + T_0) \cap C_1 = \emptyset\). Assuming that the sets \(T_i\) are well behaved (they should be simply connected and should contain the origin), the closest \(C_1\) can come to \(P\) is the edge of the region \(P + T_0\). Similarly, even if \(C_1\) wraps tightly around \(P + T_0\), the closest the second centerline \(C_2\) can come to \(P\) is the edge of \(P + T_0 + T_1\). (This conclusion is accurate only if \(P\) does not interact with \(C_2\) through \(C_1\).) Thus the \(i\)th centerline \(C_i\) is forbidden to enter the region \(P + T_0 + \cdots + T_{i-1}\). This region is a barrier for \(C_i\) in the sense of [49].

The cut \(\overline{pq}\) is safe if and only if the following condition holds:

\[(P + T_0 + T_1 + \cdots + T_n) \cap Q = \emptyset.\]  \hspace{1cm} (10-1)

One could probably build a theory of single-layer wire routing on this basis. I have chosen a simpler foundation to avoid making the proofs of the routability and routing theorems any more difficult than they already are.

**Relation to wiring norms**

Under certain common conditions the design rules defined via convolution can also be derived from a wiring norm. The basic requirement is that all the regions
Extensions of the Theorems and Algorithms

Figure 10a-2. Barriers as convolved regions. Here there are three necessary crossings of the cut from \( P \) to \( Q \), and for each crossing there is a barrier around \( P \). The minimum separation between wires is determined by convolving with a nonconvex polygonal region \( T_0 = T_1 = T_2 \). This approach can be used to model the routing of unit-width wires in a quarter-integer grid [49].

\( T_{ij} \) defining the separations between centerlines be multiples of a convex, open, symmetric region \( T \). If \( T \) is a region and \( r \geq 0 \), let \( rT \) denote the dilation of \( T \) by the factor \( r \): the region \( \{ r \cdot x : x \in T \} \). If \( T \) is convex, open, nonempty, and \( T = -T \), then we can define a norm \( \| \cdot \|_T \) by

\[
\| x \|_T = \inf \{ r \geq 0 : x \in rT \}.
\]

This norm has the property that \( rT \) is the region \( \{ x \in \mathbb{R}^2 : \| x \|_T < r \} \). As a consequence, the convolution \( rT + sT \) equals \((r + s)T\). If we suppose that each region \( T_i \) has the form \( r_iT \), then the condition \((C_1 + T_1) \cap C_2 = \emptyset \) is equivalent to the condition \( \| C_1 - C_2 \|_T \geq r_1 \). (The quantity \( \| C_1 - C_2 \|_T \) equals the infimum of \( \| x \|_T \) over all \( x \) in the convolution \( C_1 - C_2 \).) Condition (10-1) above is equivalent to \( \| P - Q \|_T \geq \sum_{i=0}^n r_i \). If every centerline \( C_i \) of a trace or obstacle can be assigned a width \( w_i \) such that the required separation \( T_{ij} \) between \( C_i \) and \( C_j \) is \( \frac{1}{2}(w_i + w_j)T \), then we are back to the sketch and design models, with \( \| \cdot \|_T \) as our wiring norm. The width of an element accounts not just for the size of its brush, but also for its required separation from other elements.

Only rarely will the use of a fixed wiring norm be too restrictive. Ideally the design rules would all be isotropic, and the design system would take full advantage of them by permitting circular arcs in centerlines and drawing components with circular brushes. In this case the wiring norm would be the euclidean norm. (Curvilinear wiring norms will be discussed in the next section.) But usually, for simplicity of programming and compatibility with manufacturing equipment, the design system deals only with polygonal regions, or only with rectangles. In this case the brushes and separation regions will all be multiples of a standard polygonal region, typically a square or an octagon, and the polygon bounding this region will be the unit polygon of the wiring norm.

But there is a problem in stipulating that the minimum separation between two components be purely a function of their widths. Consider a typical MOS technology that represents transistor gates by overlapping regions of diffusion and polysilicon. Although diffusion and polysilicon are usually thought of as different layers on the chip, for the purposes of routing they must be combined, since wires of the two
materials must not cross except where a transistor is to be placed. The minimum separation requirements between two polysilicon regions, between polysilicon and diffusion, and between two diffusion regions may all differ. Any design rule that was blind to differences between materials would have to be very conservative.

Components of differing materials

To treat the possibility that different components are made of different materials, the sketch model must be generalized. Instead of assigning each element a fixed width, a sketch will include a matrix of minimum distance constraints among the elements of the sketch. Let us denote the required separation between elements \(i\) and \(j\) by \(s(i, j)\). We assume that for all \(i, j,\) and \(k\) we have \(s(i, j) > 0\) (positivity), \(s(i, j) = s(j, i)\) (symmetry), and \(s(i, k) \leq s(i, j) + s(j, k)\) (the triangle inequality). No longer will each trace and island have a fixed territory. Instead two distinct elements \(i\) and \(j\) of a sketch will be considered properly separated if the distance between them (in the wiring norm) is at least \(s(i, j)\), or if one is a terminal of the other. Similarly, the trace \(i\) will be considered self-avoiding if the set of points lying \(\frac{1}{2}s(i, i)\) units or more from that trace has only one component that contains features. As before, a sketch is proper if its elements are properly separated and its traces are self-avoiding. We used to insist that the terminals of a trace have the same width, and that this width equal or exceed the width of the trace. This demand translates into the following: for each trace \(k\) with terminals \(i\) and \(j\), we have for each element \(l\) the relations \(s(i, l) = s(j, l) \geq s(k, l)\).

Some definitions concerning cuts must change also. The capacity of a cut will no longer account for the widths of its endpoints, since the contributions of those endpoints are uncertain. Instead we put the capacity of a cut equal to its arc length in the wiring norm. The congestion of a cut will now depend upon the sequence of traces that necessarily cross the cut. Let the endpoints of the cut lie on elements number \(e_0\) and \(e_{n+1}\), and suppose that the content of the cut is \((e_1, \ldots, e_n)\). Then the congestion of the cut is defined to be \(\sum_{i=0}^{n} s(e_i, e_{i+1})\). The cut is considered empty if \(n = 0\) and \(e_0 = e_1\), and safe if its congestion does not exceed its capacity.

I conjecture that if these changes are made to the sketch model, then the sketch routability and routing theorems continue to hold. There is strong support for this claim, I believe, from an observation concerning the proofs of the design routability and routing theorems. The key results concerning flow (Proposition 4d.2, Proposition 4f.1, Lemmas 5c.2 and 5d.2, and Proposition 6a.3) can all be reformulated in terms of content rather than flow. In other words, they never rely on the flow across a cut (in the usual design model) being independent of the ordering of the necessary crossings of that cut. To prove the conjecture, one would probably have to extend the design model by making changes corresponding to those I have suggested for the sketch model, and repeat the development of Chapters 4 through 8.
Extensions of the Theorems and Algorithms

(Some aspects of the design model that have no counterparts in the sketch model, such as the definition of the flow across a half-cut, would also need to change.) I have not attempted to carry out this program, but I have little doubt that it would eventually be successful.

If the design routing and routability theorems remain correct, then Algorithms T and R can be generalized to the new model. The required changes are simple and do not affect the worst-case performance bounds. In essence, one replaces the summing of element widths by the summing of element-to-element spacings. This replacement occurs in four places: in the construction of doorways by Algorithm R; in the determination of cable widths in the condensed RBE; in the data structure of Algorithm T that contains trace segments (which we called WS); and in the main loop of Algorithm F. Each cable in the condensed RBE must store, in addition to its width, the identities of the strands at the left and right edges of that cable, so that Algorithm T may compute the proper spacing between the strands of this cable and those of another. Processing this information still takes only constant time per cable. Likewise, the preprocessing for Algorithm F, which normally stores the lengths of the shortest paths between various nodes in the adjacency graph, must also keep track of the first and last traces along those shortest paths.

Nonlocal constraints

A further extension would remove the assumption that the minimum separations \( s(i,j) \) satisfy the triangle inequality. Such an extension may be necessary for handling complementary MOS technologies, in which there are large separation constraints between \( n \)-type and \( p \)-type transistors, far larger than the typical spacing between wires. Such nonlocal interactions probably cannot be handled at all if they involve traces. But if they involve only fixed devices, represented by islands, then one can divorce the question of whether devices are properly separated from the issue of routability. One would simply precede one’s routability test by a straightforward test, taking perhaps \( O(n^2) \) time, to ensure that each pair of features satisfies its minimum separation requirement. Routing would not be affected.

Representing devices in a sketch

A method of handling traces and features of differing materials would remove the major hindrance to accurate representation of integrated circuit layers by sketches. (In contrast, printed circuit board layers are much simpler, and the sketch model as given in Section 1A is probably adequate to describe them.) In what follows I assume that such a method is available.

By far the most common devices in an integrated circuit are contact cuts and transistors. On any particular layer, a contact cut is nothing more than a convex
region to which a wire may connect. This region is typically circular or square, depending on the wiring rules in effect, and in general it may be given the shape of the unit polygon (or circle) of the wiring norm. Thus it can be represented as a pointlike feature. Typically its width is greater than that of the attached wire; this is permitted by the sketch model. Some contacts, like the “buried” contacts in MOS technologies, connect two wires on the same routing layer. Like transistors, these must be represented as multiterminal devices.

Figure 10a-3. Representing transistors. At left, two polysilicon wires (dark shading) cross a diffusion wire (light shading) to form typical enhancement mode transistors. The situation can be represented in a sketch using four features per transistor: two of type diffusion and two of type polysilicon. This example uses Mead/Conway design rules [31] with \( \lambda \) equal to the grid spacing.

Transistors are built out of sets of features. A transistor is usually a three-terminal or four-terminal device, and so its representation must include at least three or four pointlike features to which traces can connect. In many systems the gate of a transistor can be determined implicitly by the crossover between polysilicon and diffusion wires, but since sketches prohibit crossings between wires on the same layer, such an approach is ruled out. Hence the sketch may need additional features to occupy the active area of the device, and prevent any other traces and features from approaching too closely. Finally, the transistor structure must be internally consistent; its islands must be properly separated. Figure 10a-3 shows how the simplest kind of transistor might be represented.

Unfortunately, transistor structures involving butting contacts, implantation regions, and so on are much harder to represent in sketches. To avoid design rule errors one is forced into a very conservative representation. Moreover, because terminals are points, the transistor structures are relatively inflexible. During compaction one probably cannot allow movement of the connection points relative to one another, even when such movement might be desirable; all the islands forming the transistor should be placed in the same module. Some flexibility can be regained using extended terminals, however; see Section 10C.

Terminal merging

Some representations of devices work well in the presence of wiring alone, but less well in combination with one another. The reason is that two devices may
sometimes approach more closely than their representations would indicate. For example, the two transistors in the left-hand panel of Figure 10a-3(a) are farther apart than necessary, but their terminals in the right-hand panel are too close. Figure 4 in the Introduction contains many other examples of modules overlapping. One would like to permit such overlaps, since they actually violate no design rules, but the sketch model prohibits it. The culprit is the requirement that the terminals of a trace have disjoint territories. If we remove this restriction, and allow the territories of each trace's terminals to merge, then we can route circuits like those depicted in the Introduction, without invoking special-purpose representations for groups of devices. I call this process terminal merging, because if we take the idea to its natural conclusion, it allows the terminals of a trace to coalesce during compaction.

We already have most of the machinery needed for terminal merging. By default the design model allows the terminals of a wire to be arbitrarily close, provided that the wire remains self-avoiding. In fact, when we began translating the results of the design model, in Chapter 7, we had to make special allowance for this difference between sketches and designs. We could just as well change the sketch model to permit terminal merging. This change would have only one major drawback: a complication of Algorithm C, the sketch compaction algorithm, and its proof of correctness. We discuss this issue further in Section 10C.

For now let us consider how terminal merging would affect routing and routability testing. As noted in Section 6C, whether terminals are permitted to approach one another has no affect on the ideal embedding of a design, and consequently it does not affect the ideal realizations of a sketch either. Thus Algorithm R is indifferent to terminal merging. Algorithm T, on the other hand, would have to avoid checking the degenerate cuts—those which correspond to degenerate cuts in a design. As it turns out, the only straight, degenerate cuts that are not also empty are straight cuts that coincide with rubber bands. Hence Algorithm T can be easily modified to permit terminal merging.

10B. Wiring Rules and Wiring Norms

In this section we consider four different wiring rules that may be attached to the sketch model. One asks that wires be composed of horizontal and vertical segments only. The next is even more restrictive: it requires that the wires run in a grid. In another the wiring norm is euclidean, or any other easily computable norm that is not piecewise linear, and wires are allowed to contain curves as well as line segments. The fourth is the same, but it also allows features to contain curved pieces. In each case we examine the effects on the sketch routing and routability theorems, and on
the performance of Algorithms T, R, and C. As usual, I shall characterize wiring norms by the locus of points of norm 1, the unit polygon or unit circle of the norm. Thus a piecewise linear norm is "polygonal", other norms are "curvilinear", and the norm attached to the grid model is "rectilinear".

Restrictions on wire segments

Suppose we require that the traces in a proper sketch consist only of horizontal and vertical segments. This requirement is necessary if the fabrication process or the design system can handle only rectangles whose sides are aligned with the coordinate axes. Let us assume, therefore, that all the line segments representing the features in our sketches are horizontal and vertical also. The appropriate wiring norm is rectilinear: define \( \|(x, y)\| \) to be \( \max\{|x|, |y|\} \).

Under these assumptions the sketch routability theorem still holds, and the basic sketch algorithms continue to work. Clearly it remains true that every unsafe sketch is unroutable. For the converse, I present a method for transforming the ideal realizations of the traces in a safe sketch, which may contain diagonal segments, into realizations consisting of horizontal and vertical segments only. This rerouting method appeared in an earlier paper with Leiserson on sketch routing [21], and can be added as a postprocessing phase to Algorithm R with a loss in performance of at most a constant factor. Consequently the other sketch algorithms need not be changed. Aside from using Algorithm R as a subroutine, Algorithm C relies only on the sketch routability theorem and basic properties of the sketch model. Algorithm T is likewise unaffected.

The sketch routing theorem, on the other hand, weakens somewhat. In general there are many feasible realizations for a trace that consist of horizontal and vertical segments and have minimal length under those conditions. In short, minimum-length feasible realizations are no longer unique. But the traces in a safe sketch still have minimum-length realizations that form a proper sketch. The modified Algorithm R computes such minimum-length realizations.

The rerouting process

Given the ideal realization of a sketch, the usual output of Algorithm R, we reroute each trace downward onto its struts as shown in Figure 10b-1. Only one trace \( \theta \) need be considered at a time. We first identify the joints of \( \theta \) that are stationary. Recall from Section 1D that an ideal trace is supported at each joint by a strut, which is part of a diagonal cut. With a square wiring norm the diagonal slopes are \( \pm 1 \), and hence each strut points either upward or downward and either leftward or rightward. A joint is stationary if either the strut supporting \( \theta \) there is upward, or a segment ending at that joint is horizontal or vertical. The stationary
joints divide $\theta$ into \textbf{flexible} subpaths, each of which is either a single segment or a chain of segments whose angles lie in the same quadrant: either all the segments in the chain point upward and leftward, or they all point upward and rightward, et cetera. (These facts follow from the results of Section 7D, because ideal traces are tracks.)

\textbf{Figure 10b-1.} \textit{Routing a flexible subpath onto its struts.} The flexible subpath (grey) is replaced by a rectilinear path (black). Only the struts (striped lines) of a particular diagonal slope are considered: that with opposite sign to the sign of the slopes of the segments in the subpath.

Each flexible subpath of $\theta$ is rerouted downward onto certain of its upward struts, keeping its endpoints fixed. A flexible subpath that contains only a single horizontal or vertical segment may be left alone. Consider now a flexible subpath whose segments all have positive slope. This subpath is to be rerouted onto struts that point upward and leftward. Which struts are they? The endpoints of the subpath, those that are not terminals, are supported by struts of slope $-1$. Hence this subpath passes through a particular portion of the corridor for $\theta$ corresponding to the diagonal slope $-1$. Consequently there is a well-defined sequence of struts of slope $-1$ that constrain this flexible subpath from below. If this flexible subpath consisted instead of segments of negative slope, we would reroute it onto struts that point upward and rightward in the same way.

Though I have no intention of providing any formal proofs in this chapter, it should not be difficult to believe that this method works, and works efficiently. First of all, there is no chance of changing the routing topology by moving a flexible subpath across a feature, because any such feature would give rise to a strut constraining that subpath. For the same reason, the new routings are actually traces—intersecting no features except their terminals—and they also remain properly separated from all features other than their terminals. Second, the rerouting causes no two flexible subpaths to collide. For suppose it did; suppose it pushed an upper subpath downwards onto or through a lower subpath, as shown in Figure 10b-2. By symmetry we may assume that the upper subpath originally consisted of segments of negative slope. Then the lower subpath would have an upward, rightward strut that intersects the other subpath as well. But the feature that gives rise to this strut also gives rise to a longer strut for the upper subpath. Hence the upper subpath could not be rerouted down onto the lower one. (This argument
is reminiscent of the proofs of Lemmas 5c.2 and 5c.3, and one could formalize it by adapting those proof techniques.) Finally, the rerouting is efficient; it requires only constant time per strut, and hence consumes no more time and space, up to constant factors, than Algorithm R normally does.

\textbf{Figure 10b-2.} No flexible subpaths collide. If one were somehow rerouted down through another, it would cross a strut for the lower one (striped path) that is part of a strut for the upper one.

One last claim is that the traces output by the rerouting process are as short as possible. This claim is harder to verify, because it depends on the unproven fact that no feasible realization of a trace is shorter than its ideal realization in any norm. Suppose we measure wire length with the taxicab or $L^1$ norm. The ideal realization of a trace has minimum length in this norm among all feasible realizations of that trace. Since the rerouting process does not affect arc length in the taxicab norm, the rerouted traces are still optimal in this sense.

\textbf{The grid model}

From the model in which a proper sketch consists only of horizontal and vertical segments, the grid model is but a short step away. Let $R$ denote the real line and $Z$ the set of integers. The grid relevant to single-layer routing is the set $\{(x, y) \in R^2 : x \in Z \text{ or } y \in Z\}$ of points with at least one integer coordinate. The lines it contains are called \textbf{gridlines}, and the points $Z \times Z$ where they intersect are called \textbf{gridpoints}. The grid model for sketches makes the following assumptions: all features in a sketch lie in the grid; all feature endpoints are gridpoints; the width of each element in a sketch is an odd integer; and the wiring norm is the rectilinear one. It mandates that a sketch is not proper unless each of its traces lies within the grid.

Despite the additional restrictions imposed by the grid model, it need not be treated any differently. The reason is that the rerouting stage of Algorithm R still produces proper realizations. Because the width of every sketch element is an odd integer, all the struts have integral length in the wiring norm, and so their endpoints are gridpoints. Consequently the rerouted traces all lie in the grid. The sketch routability theorem continues to hold, the sketch routing theorem holds in
its weak form (trace lengths can be simultaneously minimized, but not uniquely), and Algorithm T is unchanged.

Algorithm C continues to work because it never considers a nonintegral displacement for any module. The additional restrictions on sketches could only cause Algorithm C to err by producing an improper sketch as output. We show that this event never happens. The congestions and capacities of all cuts, as well as the coordinates of every feature endpoint, are integers. Hence in each constraint that Algorithm C adds to its constraint system, the constant is an integer. Therefore all paths through the constraint graph have integral length, and so the configurations that Algorithm C computes involve integer displacements only. Since the input sketch is assumed to be routable, its features must lie in gridlines and their endpoints on gridpoints. The same is true of the sketch that Algorithm C gives to Algorithm R, and so the output of Algorithm C is proper.

Curvilinear models

Having gone nearly as far as possible in restricting the traces in a proper sketch, from now on we consider allowing them some liberties that are lacking in the original sketch model. The simplest of these is the ability to contain arcs as well as segments. Such an ability is useless as long as features are line segments and the wiring norm is polygonal, so we will consider relaxing both of these assumptions. First we assume that the wiring norm is not polygonal.

Although we have no mechanism for dealing with curvilinear traces, there is a trick that converts a curvilinear norm to a polygonal norm for the purpose of routing. Using this trick we can prove that the sketch routability theorem holds for any wiring norm. For the sake of simplicity we take the wiring norm to be the euclidean norm, as there is probably no call for any other nonpolygonal norm.

Approximating the wiring norm

We construct the surrogate norm from the features in the sketch to be routed. Let $S^1$ be the unit circle $\{ x \in \mathbb{R}^2 : |x| = 1 \}$, and let $\Lambda$ be a set that contains, for each feature endpoint $p$ and each feature $Q$ having a cut to $p$, the line segment $\overline{pq}$ from $p$ to $Q$ that minimizes $|q - p|$. Let $C$ be a convex polygon, symmetric about the origin, which does not intersect the inside of $S^1$. Suppose further that for each line $L$ through the origin that is parallel to a line segment in $\Lambda$, two sides of $C$ are tangent to $S^1$ at the intersections of $S^1$ with $L$. Such a polygon is easily created, as shown in Figure 10b-3. Start with any symmetric, convex polygon whose inside contains that of $S^1$. Then for each segment $\lambda$ in $\Lambda$, intersect that polygon with the two lines perpendicular to $\lambda$ and tangent to $S^1$, yielding either one or three polygons; take the one that encloses the origin. The polygon $C$ that results from
this process is the unit polygon of a norm $\| \cdot \|$ defined as follows: for any point $x \in \mathbb{R}^2$, the quantity $\|x\|$ is the number $r \geq 0$ such that $x = rc$ form some $c \in C$. You may check that $\| \cdot \|$ is in fact a norm. It is stronger, or more restrictive, than the euclidean norm, in that $\|x\| \leq |x|$ for all points $x \in \mathbb{R}^2$.

**Figure 10b-3. Constructing the unit polygon.** The striped lines take the slopes of all the critical cuts and all the line segments that would be critical cuts if their middles crossed no features. We circumscribe about the unit circle $S^1$ a polygon (dark lines) that is tangent to $S^1$ wherever these lines intersect $S^1$.

The new polygonal norm has two key properties that allow it to substitute for the wiring norm. The critical cuts of the sketch are the same in both norms, and they have the same length in both norms. In each norm the critical cuts are those cuts that begin at a feature endpoint $p$ and travel to the closest point on another feature $Q$, with ties broken using the euclidean norm. Hence it suffices to show that if the point $q \in Q$ minimizes $|q - p|$, then it also minimizes $\|q - p\|$, and that the two distances $|q - p|$ and $|q - p|$ are equal. Let $\lambda \in \Lambda$ be the line segment $\overline{pq}$. Let $O$ be the circle $\{x : |x - p| = |q - p|\}$ centered at $p$ and passing through $q$, and let $P$ be the polygon $\{x : \|x - p\| = |q - p|\}$. By the construction of $C$, the polygon $P$ is tangent to $O$ at $q$. Thus $\|q - p\| = |q - p|$. If $T$ is the line through $q$ perpendicular to $\lambda$, then by the choice of $q$, the feature $Q$ does not intersect the side of $T$ that contains $p$. Also $T$ is tangent to $O$ at $q$, and hence is also tangent to $P$ at $q$. Since $P$ is convex, the only side of $T$ it intersects is that containing $p$. Hence no point of $Q$ is closer to $p$ than $q$ in the norm $\| \cdot \|$.

Now we can see why a sketch whose critical cuts are safe is still routable. Switching from the euclidean norm $| \cdot |$ to the polygonal norm $\| \cdot \|$, the critical cuts do not change, and neither do their lengths. Hence their capacities remain unchanged, as do their flows, emptiness, and safety. Thus the sketch whose critical cuts are safe in the norm $| \cdot |$ has the same property with respect to the norm $\| \cdot \|$, and hence is routable in the norm $\| \cdot \|$, by the sketch routability theorem. In other words, our sketch has a realization that is proper with respect to the new wiring norm. The unit polygon of the norm $\| \cdot \|$ circumscribes the unit circle of the norm $| \cdot |$, and hence every element of our sketch has larger extent in the former norm than in the latter. (The polygonal norm is stronger.) Therefore, as in Lemma 8b.1,
the realization that is proper in the polygonal norm is also proper in the euclidean norm.

A similar argument proves the other direction of the sketch routability theorem for the euclidean wiring norm. Given a sketch containing an unsafe, nonempty, straight cut \( \lambda \), critical or not, one can construct a weaker polygonal norm in which \( \lambda \) remains unsafe. Let \( L \) be the line through the origin parallel to \( \lambda \), and let \( P \) and \( Q \) be the islands containing the endpoints of \( \lambda \). It suffices to inscribe a convex, symmetric polygon \( C \) in \( S^1 \) whose intersections \( \pm x \) with \( L \) satisfy

\[
\text{flow}(\lambda) > \frac{|\lambda|}{|x|} - \text{width}(P)/2 - \text{width}(Q)/2.
\]

This condition ensures that in the norm whose unit polygon is \( C \), the cut \( \lambda \) remains unsafe. Hence by Proposition 8b.3, our sketch is unroutable in the new polygonal norm. Since this norm is weaker than the euclidean norm, meaning that it gives rise to smaller extents for sketch elements, any sketch that is proper in the euclidean norm is also proper in the polygonal norm. Consequently our sketch is unroutable in the euclidean norm also.

Because the sketch routability theorem carries over to curvilinear wiring norms, so do Algorithms T and C. The sketch routing algorithm, however, fares less well. The trick of replacing the wiring norm by a polygonal norm is computationally effective, but the complexity of the resulting norm slows down the routing process greatly. The time and space complexity of Algorithm R are both proportional to the number of diagonal slopes, which can be up to \( \Theta(n^2) \) if most features are visible from most other features. Hence Algorithm R could use up to \( O(n^4 \log n) \) time and \( O(n^4) \) space trying to route with the euclidean norm, and the result would not even minimize wire length.

Other approaches to curvilinear wires

A better approach to the routing problem is needed if routing with a euclidean metric is to be practical. Storb et al. [33] have recently developed a routing algorithm for sketches in the euclidean metric that combines the ideas of scanning over the rubber-band equivalent with the barrier construction methods of [52]. In a sense, their algorithm routes wires in the simply connected covering space of the routing region. It runs in time \( O(|F|^2 |T|) \) on a sketch \((F,T)\). Another approach is to use relaxation starting from the rubber-band equivalent, inflating wires to their full width one at a time, moving other wires out of the way, and keeping all the wires tight. This approach seems to work well in practice [36], especially if performed incrementally as the sketch is being input. The worst-case running time of this method is as yet unknown.
We conclude that the problem of finding a nearly optimal algorithm for sketch routing in a curvilinear wiring norm is still open. (Perhaps the best idea is to drop the euclidean norm in favor of some reasonable, prespecified polygonal approximation.) My theorems of single-layer routing probably do extend to arbitrary wiring norms and quite general feature shapes, however, as I now discuss.

**Arcs in traces and features**

Any statements I make concerning single-layer routing with nonpolygonal wires and features must be somewhat speculative. The only way to justify them on the basis of present knowledge about sketches would involve a limiting argument like that relating sketches to designs. As we know, such arguments are extremely tedious. Another approach, which is perhaps no shorter but requires no new ideas, is to strengthen the existing theory, replacing ‘piecewise linear’ with ‘piecewise smooth’ or some intermediate condition. In my view, nothing but a heap of technical detail stands in the way of this improvement, but those readers who are less than intimately familiar with Chapters 2 through 9 may not share my confidence. So the generalizations that I am about to propose must be taken as conjectures.

The design routability and routing theorems hold under any wiring norm, provided that wires are permitted to include canonical paths in sets of the form \( \{ x : \| x - F \| = c \} \), where \( F \) is a fringe and \( c > 0 \) is a constant. The same is true even if fringes are the images of arbitrary piecewise smooth loops. (As before, the inside of each terminal must be a convex set.) This claim is really more general and less obvious than necessary. One important special case may be easier to swallow: the design routability and routing theorems are true in the euclidean wiring norm if wires and fringes can contain circular arcs of arbitrary radii. These conjectures have counterparts in the sketch model as well. In particular, I claim that the sketch routability and routing theorems are true in the euclidean wiring norm if features and trace segments can be circular arcs.

The sketch algorithms fare more poorly than the theorems when the sketch model is generalized. As we know, the sketch routing algorithm is unable to operate with curved elements. Algorithm R is grounded in the idea of building ideal realizations out of partial realizations. When the wiring norm is not polygonal, this idea cannot be applied, except by approximating the wiring norm.

At least while features remain straight, critical cuts can still be identified. Hence Algorithm T continues to work, as does Algorithm C, at least until the point where the output sketch needs to be routed. When features are not straight, however, the critical cuts as currently defined need not be decisive. Proposition 8b.4, which shows for the standard sketch model that the exposed critical cuts are decisive, relies on features being convex. The decisiveness of critical cuts may be salvaged if
every curved feature is part of an island which has an inside and an outside, only one of which contains traces, and the curved feature bulges toward that side. In this case one can push Proposition 8b.4 through. Otherwise one must find a new set of decisive cuts using the methods of Section 6D. What's worse, the the rubber-band equivalent of a sketch is harder to compute when features are not straight, because Algorithm W no longer applies. Fortunately, one can always fall back to Algorithm F of Section 9B for computing congestion. That algorithm, while slower, is independent of the shapes of features and traces.

10C. The Terminals of Traces

In this section we consider more radical changes to the sketch model than merely altering the definition of what sketches are proper. The changes have two purposes: to allow sketches to represent wiring problems that they previously could not; and to give our sketch algorithms more freedom, so that they may find better and more compact realizations of the sketches given to them. Three extensions of the sketch problems come to mind. One, which was discussed briefly in Section 10A, allows the terminals of a trace to merge during compaction. Another allows terminals to be line segments or convex polygons, and allows trace endpoints to move along the boundaries of these extended terminals. The third attempts to remedy the most glaring defect in the sketch model by providing for multiterminal nets: wires that connect to three or more terminals.

All these extensions can, I believe, be incorporated into the sketch model, at the expense of complicating the sketch algorithms and their proofs of correctness. To justify terminal merging and the addition of extended terminals is not too difficult, since these ingredients are already present in the design model. It involves two things: rederiving the correspondence between sketches and designs, and upgrading the sketch algorithms. To add multiterminal nets is extremely hard, however, because it requires a major extension of the design model. In fact, my only reason for thinking that the design model can accommodate multiterminal nets is a strong faith in the proof techniques of Chapters 3 through 7, which have shown themselves in the course of my research to be remarkably adaptable. The main problem is to find the right definitions.

Merging terminals during compaction

Terminal merging was discussed briefly at the end of Section 10A. It begins with the notion that the terminals of each trace in a sketch should not be artificially kept apart; their extents should be allowed to overlap in a proper sketch. We saw
how to modify Algorithm T to test routability in the new sense: it must ignore all degenerate critical cuts. For if the extents of terminals may overlap, then the sketch routability theorem must be changed to read: A sketch is routable if and only if its nonempty, nondegenerate, critical cuts are safe. The definition of degeneracy here corresponds to that in the design model, and may be stated as follows. A bridge $\beta$ in a sketch $S$ is degenerate if there is a piecewise linear homotopy $B$ such that $B(\cdot, 0) = \beta$, for all $t \in (0, 1)$ the path $B(\cdot, t)$ is a bridge in $S$, and $B(\cdot, 1)$ is a path in an obstacle $\sigma^c S$ or in the image of a trace of $S$ and its terminals.

If these changes are adopted, then Algorithm C breaks down in two ways. First, if we retain the provision in the sketch compaction problem that prevents the sketch topology from changing, then the set of acceptable configurations can no longer be represented in as constraint graph. For if some two terminals in different modules can approach arbitrarily closely but cannot coincide, then the set of configurations that represent routable sketches is not closed. Second, protection of the critical cuts is no longer necessary for routability, because a unprotected critical cut can be degenerate and therefore irrelevant. Unlike some changes to the sketch model, this one cannot be accommodated by finding a new sequence of potential cuts with the routability, convexity, ordering, and boundary properties. The problem is that the sketch routability theorem, normally used to justify the routability property, has changed significantly. Fortunately, both breakdowns can be repaired in fairly obvious ways.

We solve the first problem by redefining the configuration space so that the terminals of each trace can merge or even cross over one another. This sort of topological change damages only a small part of the correctness proof of Algorithm C: the claim that the sketches corresponding to different configurations are homeomorphic. We used this claim in Corollary 9d.2 to prove that the adjacency graph of the sketch is independent of configuration, and thus a single adjacency graph could be used for computing flows in all relevant configurations. In fact Algorithm F still computes flow correctly, even when one terminal of a trace passes through the other. For as far as flow is concerned, one may pretend that the first terminal passed above or below the second terminal by a tiny distance.

The second problem may be addressed by changing the definition of protection and reworking the proofs in Sections 9E and 9F. Recall that a configuration $d$ protects a potential cut $\psi$ unless $\psi(d)$ is an unsafe, nonempty cut. Under the new definition $d$ protects $\psi$ unless $\psi(d)$ is an unsafe, nonempty, nondegenerate cut. The nondegeneracy condition causes no more trouble than the nonemptiness condition; its presence is felt only in Lemma 9e.3.

Only one major change is needed in Algorithm C. All potential cuts must be tested for degeneracy, including the horizontal ones that define the initial constraint set. A straight cut between different modules, which is the only kind Algorithm C
ever considers, is degenerate if and only if its trace code matches that of a trace between the same features as the cut. Since there is at most one trace between those features, the test for degeneracy is quick, at least compared to the computation of flow that precedes it. The other change is minor: when preparing the output sketch for routing, Algorithm C should eliminate all traces whose terminals coincide.

Extended terminals

The trouble with sketches is that their features have empty interior. Consequently our proof techniques, which rely heavily on lifting to a covering space, do not apply directly to sketches. All the theorems concerning sketches must be derived from corresponding theorems about designs. But what is a drawback in proving theorems is a virtue in designing algorithms: if all terminals are points, one never need worry where to place the endpoint of a trace. When we relax the restriction that terminals be points, we immediately face several problems in the construction and use of the rubber-band equivalent, and in routing, concerning the placement of trace endpoints. These are the same problems we sidestepped in Chapter 7. Everything I say about routing and testing routability in the presence of extended terminals applies equally well to the design model.

Any convex island in a sketch may be an extended terminal, provided that its trace contacts it from the outside. Thus terminals can be points, line segments, and convex polygons. The restrictions on extended terminals arise because the terminals of a wire in a design are convex, inner fringes.

Because extended terminals are an integral part of the design model, the correspondence between sketches and designs can easily admit them. A realization of a trace is any bridge-homotopic trace, and so one may move the endpoints of a trace along their respective terminals. Thus the notion of homotopy for traces is in line with that for wires. The sketch routability and routing theorems go through essentially as in Chapter 8, which is to say, with either a lot of handwaving or a lot of hard work. (Note: I have not actually done the hard work, so there is some chance that the extension contains a fatal flaw.) One difference is that the ideal realization of a trace is no longer necessarily unique.

Routing is also more difficult when extended terminals are present, due to the need to locate trace endpoints. I expect, however, that the following approach can be made to work. As in Algorithm R one first computes for every trace a corridor for each diagonal slope. But now these corridors should include doorways that pass right through the trace's terminals. The partial realizations of the trace are now the shortest paths through these corridors from one terminal to the other. Because terminals are convex, the partial realizations should not be too difficult to compute. Now one merges the partial realizations as before, with some minor extensions to determine from which partial realizations the ideal realization takes its endpoints.
Detecting trivial crossings

The other things that must change when extended terminals are present are the algorithms for computing congestion and necessary crossings. In particular, the rubber-band equivalent of a sketch must be treated somewhat differently. As explained in Section 7C, every crossing between a straight cut and a rubber band is either necessary or trivial. Informally speaking, a crossing is trivial if one of its corresponding half-cuts is homotopic to a path in a single island. See Figure 10c-1. The trivial crossings used only to occur at trace endpoints, but that is no longer true. Hence the RBE itself can no longer identify the trivial crossings; some additional computation is needed.

![Figure 10c-1. Trivial crossings.](image)

Fortunately, we have considerable freedom in choosing where on its terminals a rubber band should begin and end. Technically, the rubber band for a trace is the shortest path through a certain corridor that begins and ends at the endpoints of the trace. But we may replace the rubber band of a trace by the rubber band of any route for that trace without affecting the content of any cut. For starters we choose each rubber band so that neither its first nor its last segment lies within a terminal. Under this condition only the first and last crossings of a cut can be trivial.

A crossing may be tested for triviality as follows. For each terminal shared by the cut and the rubber band, consider the loop formed by the terminal and the portions of the cut and rubber band between the terminal and the crossing. If the inside of this loop is free of features, then the crossing is trivial. But if every such loop encloses a feature, then the crossing is nontrivial, and therefore necessary. If the trace and the terminal share no terminals, then nontriviality is automatic.

Testing whether a loop encloses some feature is generally difficult, but can be simplified by a judicious choice of rubber bands. We say that a rubber band $\rho$ for a trace $\theta$ is stiff if no subpath of $\rho$ that is not straight can be replaced by a straight path to yield the rubber band of a route for $\theta$. Every trace has some route whose rubber band is stiff, because one can keep eliminating joints of the rubber band until no further subpath can be straightened. For a crossing of a cut by a stiff rubber band $\rho$ to be trivial, it must occur in the first or last segment of $\rho$; otherwise one could straighten out a subpath of $\rho$. Hence if stiff rubber bands are used, the
loops we must test are essentially triangles. One can then apply any of the various retrieval or counting algorithms for triangles [7, 56] to test triviality of crossings. Good average-case performance, say $O(\log n)$ or $O(\log^2 n)$ per search, can probably be achieved with a quadtree structure.

Stiff rubber bands may be computed according to the following outline. One first uses Algorithm W to compute the "envelope" of the rubber bands of routes of a given trace, as shown in Figure 10c-2(a). The left-hand boundary is a rubber band that begins so far counterclockwise on the first terminal that its first segment lies on that terminal, and ends so far clockwise on the second terminal that its last segment lies on that terminal. The right-hand boundary is similar. If the two boundaries of this envelope do not intersect, then a straight rubber band exists. If they do intersect, then they intersect along a path that forms part of the middle of the desired rubber band. This path need only be augmented by straight paths from the first terminal and to the last terminal. If possible, these straight paths should be collinear with the segments to which they attach. See Figure 10c-2(b). I omit the details because I believe there are better ways of computing necessary crossings and congestion in the presence of extended terminals. One candidate method is mentioned in the following section.

Figure 10c-2. The envelope for a rubber band. In (a) the boundaries are separate, and the desired rubber band (grey) is straight. In (b) the two boundaries intersect in the middle (black segments) of the stiff rubber band.

Algorithm F of Section 9B needs modifications along the same lines. Like the RBE, Algorithm F essentially computes a minimal sequence of crossings of a cut, and if the endpoints of that cut lie on extended terminals, then the first and last crossings may be unnecessary, i.e., trivial. To test whether a crossing of a trace by a cut is trivial, one looks at each extended terminal they share, and compares the trace codes (gate lists) of the portions of the cut and trace from that terminal up to the crossing. If they are equal, then the crossing is trivial. To incorporate this test into the optimized version of Algorithm F is not trivial, but it can be done.

Multiterminal nets

Perhaps the most problematic weakness in the sketch model is its complete in-
ability to represent wires with more than two terminals. I now present an extension of the sketch model that may alleviate this problem. Of course, one can always break a multiterminal wire into two-terminal wires by introducing connector modules (groups of terminals) along the way. But doing so defeats the purpose of having flexible interconnections.

We may represent a multiterminal wire by a ring-shaped set of traces, as shown in Figure 10c-3(a). I call this set of traces a net. The traces in a net can intersect, and indeed they must intersect at their terminals unless extended terminals are present. But no two traces in the net may make a necessary crossing, so Figure 10c-3(b) is ruled out; and the loop that the traces form must enclose no features. Furthermore, the traces in a net must have the same width, as must the terminals of the net, and the width of the traces may not exceed that of the terminals. To route a net is to route each of its traces; the result is always a net. For technical convenience we assume that two-terminal wires, as well as wires with more terminals, are represented as nets. (A two-terminal net is just a pair of bridge-homotopic traces.)

![Figure 10c-3. Valid and invalid multiterminal nets. Part (a) shows a sketch with nets in place of traces. Each net is a loop, not necessarily simple, of traces (light paths), no two of which cross over. The paths in part (b) therefore do not form a net.](image)

A net need not represent the final form of a multiterminal wire. Instead, having routed the nets in a sketch, one can then replace them by more conventional tree-shaped wires. Each wire’s centerline should be placed within the net and the points it encloses. Unlike the traces of the net, however, this tree-shaped wire will not generally have its total arc length minimized; to minimize it involves solving a Steiner tree problem, which is NP-complete [13].

There are two fairly natural separation rules one might apply to nets, one stricter and one looser. Both insist that in a proper sketch no trace may cross over another, although two traces in the same net may coincide along part of their length. The strict rule treats the traces in a net as independent entities, each with its own extent; it says that a sketch is proper only if whenever two of its elements (traces and islands) have overlapping extents, they are either a trace and one of its terminals, or two traces that share a terminal. It further requires that each trace in a proper
sketch be self-avoiding. The other is more akin to the rule for terminal merging; it treats each net together with its terminals as a whole object. We define the extent of a net to be the union of the extents of its traces and terminals. Under the loose rule a sketch is proper only if the extents of its nets and obstacles (nonterminal islands) are disjoint and its nets are self-avoiding, meaning that no net has an extent that separates two islands of the sketch.

Unfortunately, only the strict rule is likely to give rise to a satisfactory routability theorem, and so we adopt this one. Under the loose rule there is no satisfactory definition of congestion. (Making sense of the loose rule requires allowing each trace to pass over terminals in its net, which the present framework absolutely forbids.) The conjunction of terminal merging and multiterminal nets must wait for the model presented in Section 10D. Under the strict rule, the congestion of a cut $\overline{pq}$ may be defined as the total width of the traces in the content of $\overline{pq}$ after eliminating every second trace wherever consecutive traces in the content are part of the same net. One must remove alternate traces because when a cut necessarily crosses a net, it usually makes necessary crossings with two traces of the net—one entering the inside of the net, and one leaving. Exceptions occur only at the endpoints of the cut, and then only if those endpoints are terminals.

**Impact of multiterminal nets**

Provided that all the mathematics works out, the sketch routing and routability theorems should continue to hold, and the sketch algorithms should change only slightly. The only major changes come in the computation of congestion by Algorithm F and the rubber-band equivalent, and the computation of doorways by Algorithm R. What changes in Algorithm R, of course, is that pairs of crossings by traces in the same net must be considered as single crossings for the purpose of computing the lengths of struts. Algorithm T now requires that each cable in the condensed RBE be assigned a width that reflects the total width of adjacent pairs of strands belonging to the same net, and must also record any strands left over so that Algorithm T can correct for duplication of crossings. Finally, when Algorithm F computes the content of a cut, it must also collapse pairs of consecutive traces when they fall in the same net. (Section 10A described similar modifications to Algorithms T and F. Like these, they arose from a definition of congestion in terms of content.)

The hard part, of course, is proving that the sketch algorithms remain correct. I see no other option than to extend the design model and generalize the whole theory of single-layer routing. If one stuck to the same outline, at the very least Chapters 4 through 6 would have to be overhauled. In discussing other extensions of sketch problems I have expressed confidence that the relevant theorems could be
strengthened to match, but here I cannot. Many of the proofs in these chapters use the assumption that a terminal intersects at most one wire. To eliminate this assumption in the presence of the new definitions may be easy, or it may be impossible. Whether the sketch model can accept multiterminal nets is really an open question.

10D. An Alternative to the Sketch Model

No discussion of sketches would be complete without a mention of alternative models. We begin by exploring the problems involved in representing how wires connect to their terminals. The difficulty of adding extended terminals and multiterminal nets to the sketch model, and the awkwardness of working mathematically with sketches, suggest that an entirely new model may be needed if our understanding of single-layer wire routing is to be advanced. In this section I present a pair of models for single-layer routing problems that may resolve these two issues. One, like the design model, is designed for mathematical convenience, while the other, like the sketch model, is intended for algorithmic use. By offering a new perspective on the connection of wires to their terminals, they handle extended terminals and multiterminal nets in an elegant manner. And because the two models are closer together than sketches and designs, they promise a smoother connection between the mathematical and algorithmic parts of the theory.

Modeling of terminals

The connection of wires to their terminals is the major stumbling block in the development of general models for single-layer routing. (A glance at the topics of the preceding section may help to convince you.) Terminals cause difficulties both in the technical development of the model and in its use. Because a wire has terminals, its endpoints must be treated differently from its middle, and its terminals must be treated differently from all other obstacles. For instance, because of the special status of terminals, the rubber-band equivalent of a sketch must distinguish between trivial and nontrivial crossings. This problem is particularly acute when extended terminals are present. In the design model, the burden of keeping track of respect and degeneracy for cuts and half-cuts can be traced to the possibility of a cut or wire winding around a terminal.

Slight changes in the way terminals are managed can make or break routability theorems and routing algorithms. As an example, suppose that we allowed a sketch to have extended terminals but fixed the endpoints of each trace at specific points on those terminals. To make things more plausible, we may permit the traces to
run along their terminals for some distance before departing. With this extension, my theory of routing collapses. Consider a trace that happens to wind once or more about one of its terminals. The middle of this trace can come arbitrarily close to its (fixed) endpoints, but cannot intersect them. Hence we must give up all hope of routing with minimum-length traces. Moreover, we can no longer expect to decide routability by the safety of certain cuts; some cuts will need capacities that strictly exceed their congestions. In other words, some of the routability conditions will go from closed to open. One can alleviate these problems by allowing traces to have self-intersections. The result is a model that is more complicated than the original without being any more expressive.

![Figure 10d-1. Compaction with multiterminal nets.](image)

The problems of terminal connections come to the fore when multiterminal nets are considered. Many seemingly natural ways of handling multiterminal wires simply do not work. One try that fails is the "loose" wiring rule described at the end of Section 10C. The stricter rule that we adopted is not an entirely satisfactory foundation for a study of routability either. It disallows routings that might often be desirable, by a prohibiting a trace to approach the terminals in its net except those to which it connects. Figure 10d-1 is only the simplest example. During routing or compaction one might greatly improve the layout by moving part of a net across one of its terminals. But this sort of topological change is foreign to the sketch and design models.

The network model

Now we come to the point of this section: a novel perspective on wire/terminal connections that gives rise to very pretty alternatives to sketches and designs. For concreteness I discuss the idea as an modification of the sketch model, and call the analogue of a sketch a network. We think of a wire as a net, a simple loop in the routing region that encloses its terminals rather than intersecting them. See Figure 10d-2. A net may not touch any terminal or obstacle, but it must enclose at least one terminal and may enclose more than two. Terminals may be islands of any shape. No two nets may intersect, and none may enclose another. To route a
net, we replace it by any other net that is homotopic as a map of the circle $S^1$ into the routing region.

![Diagram of network model]

**Figure 10d-2. The network model.** At left is a network of islands (made of dark points and lines) and nets (light curves). Each of these elements has a width, the number next to it. Between the two layouts are several copies of the unit polygon. At right is the unique proper realization of this network with minimum-length nets. Within these nets one can route the centerlines (grey lines) of multiterminal wires that are properly separated.

The design rules for nets are similar to those for sketches and designs, but are perhaps even simpler. Each net has a positive width and a corresponding territory, the set of points whose distance from its image is less than half its width. In the simplest case islands have zero width, and their territories are just themselves. A layout is proper if and only if its elements (nets and obstacles) have disjoint extents and its nets are self-avoiding. Thus the terminals of a net are treated no differently than any other obstacles. A net is self-avoiding if the complement of its extent has only two components that contain obstacles: one inside the net, and one outside. If desired, islands may be given positive widths and corresponding territories. The widths of a net’s terminals need have no special relation to one another or to the width of the net.

I conjecture that the sketch routability and routing theorems carry over to this “network model” in an obvious way. Cuts, capacity, safety, and emptiness remain

* There is an amusing parallel between my models for multiterminal connections and the models once discussed by particle physicists of the confinement of quarks in various subatomic particles. According to current theory, each nucleon and each meson is made of quarks bound together by a force that increases with distance, so that no quark can be isolated from the others. For the purpose of predicting properties of these particles, two models of quark confinement were introduced: a “string model”, which pictured the quarks as being held together by elastic strings, and a “bag model”, which represented the binding force as a flexible bag containing the quarks. One hears less of these models now, presumably because quark interactions have become better understood.
the same, as do the critical cuts. One must replace ‘trace’ by ‘net’ in the definition of congestion. Then the routability theorem reads: A network is routable if and only if its nonempty critical cuts are safe. The routing theorem remains at full strength: In a routable network, every net has a unique minimum-length feasible realization, and these realizations form a proper network. In other words, net lengths can be simultaneously minimized, and the layout that does this is unique. One can even modify the network model to permit the extents of the terminals of each net to overlap. As with sketches, the only change is to eliminate degenerate cuts from those that determine routability. Here a cut is degenerate if and only if (a) it can be collapsed into an island, or (b) some bridge-homotopic cut is entirely enclosed by a net.

Aside from the theorems it may support, the network model has two very nice properties in itself. First, it contains within it an isomorphic copy of the sketch model. To convert a sketch into a network, replace each trace by a net that surrounds its terminals, give that net half the width of the trace, and deduce the width of the trace from the widths of its terminals. I claim that the resulting network is routable if and only if the sketch was routable. This claim is not even very hard to prove. Second, the network model needs only minor changes to relate it to designs instead of sketches. One needs only replace the routing region by a sheet, and change some terminology: islands become fringes, territories become extents, and so on. The notion of net homotopy, in particular, needs no change. In fact, the only substantive change is that the routing obstacles grow to become polygons; and these were already allowed. So by mildly restricting the obstacles in the network model, we obtain a model that can be analyzed using covering spaces.

Benefits of nets

The network models may have many applications to single-layer wire routing, but it arrived too late in my research to have any influence on the bulk of this thesis. So close are the two network models to one another, and so naturally are the sketch and design models embedded in them, that they may actually form a better bridge between sketches and designs than the direct correspondence of Chapter 8. Even if not, the idea of converting wires and traces into nets could be of significant use in dealing with extended terminals. See Figure 10d-3. When computing flow, the first and last crossings of a cut would be the only trivial ones, and so no complicated test for triviality would be required. (This idea works only for two-terminal wires.)

But the network models are really intended to support a new and better theory of single-layer wire routing, all the way from basic topology to algorithms. If such a theory could be developed, it would have several advantages over that presented here.
Figure 10d-3. The rubber-band equivalent of a network. The loops shown here are the rubber bands for the nets in Figure 10d-2. The congestion of a cut equals the sum, over all crossings of that cut by rubber bands, of the width of the net corresponding to that crossing.

• It would treat all nets as fully flexible interconnections, regardless of the number of terminals they enclosed. In particular, it would solve the problem of Figure 10d-1: as terminals move or other obstacles intrude, each net deforms so as to connect its terminals in the best way.

• It would neatly decompose the routing problem for multiterminal wires and wires with nonconvex terminals. For such wires the problem of minimizing total arc length is generally difficult. When the net corresponding to a wire was routed, it would delimit the region in which that wire should run. Any reasonable heuristic could then be used to route the actual wire. In the case of nets with two convex terminals, the heuristic could be replaced by a fast algorithm that minimizes wire length.

• It would handle extended terminals with no penalty in efficiency or algorithm complexity. The problems of placing wire and cut endpoints would vanish, taking with them the need to distinguish path plans and link plans. Related technical difficulties, like the possibility that consecutive gates in an ideal wire’s tunnel intersect, would also disappear.

• The RBE data structure for computing flows and crossing sequences would become simpler. Every net has a unique rubber band, regardless of terminal shape, and every crossing of this rubber band by a cut corresponds to a necessary crossing of the net by the cut. Compare Figure 10d-3.

• The concept of respect would disappear entirely from the theory, thus simplifying many proofs. Degeneracy, also, would play a much smaller role, except where terminal merging is concerned.

• Finally, the task of relating the two network models would be much simpler than the task of relating sketches to designs. In particular, the algorithms for routing, testing routability, and compaction would be nearly identical in the two models.

In contrast, I can think of only a couple of disadvantages that would accrue to algorithms and theory in the network model. One is that the algorithms would
be slightly slower in the case of two-point nets, since there would be approximately twice as many crossings between nets and cuts as between traces and cuts. The path-finding algorithms might also need to run in two passes because they cannot initially identify any point that the output net should pass through. I am confident that no significant new algorithmic ideas would be needed. The only telling objection is mathematical. Because nets are essential loops, they cannot be lifted to a simply connected covering space of the routing region. A different covering space is needed for analytical purposes: the universal covering space among those to which all the nets can be lifted. Its properties are harder to derive than those of blankets. Consequently, the theory of the network model may depend on more advanced topological concepts than those I have employed.
Conclusion

A Critical Review

This dissertation did not begin the study of wire routing with homotopy constraints, and neither can it end it. At the risk of exhausting the reader's patience, therefore, I will say a few more words about the sketch problems. The first part of this Conclusion summarizes my main results on routing and compaction, puts them in perspective against the practical problems of wire routing, and then recognizes several related works, including several from which my papers have drawn and one to which my research has already contributed. The second part takes a careful look at the sketch model, and draws from that look several suggestions for further research.

A. Summary of Results

I have presented efficient algorithms for three problems of single-layer wire routing with homotopy constraints: sketch routability, sketch routing, and one-dimensional sketch compaction. These problems are natural abstractions of placement and routing tasks that circuit designers face when they distinguish flexible wires from rigid features. The tasks are these: determine whether a layout is routable under a given topology and layer assignment; if so, route it with wires of minimum length; and compact the layout horizontally, introducing jogs into wires in an optimal fashion. My solutions to the sketch problems work in a variety of wiring models, and they can be extended to handle a variety of useful constructs.

With the possible exception of the sketch compaction algorithm, all the algorithms are efficient enough to be used in practice. The algorithms for sketch routing and routability testing run in time $O(n^2 \log n)$ on input of size $n$. Sections 1E and 1F suggest that their average-case time requirements can be reduced to $O(n^{3/2} \log n)$ time or less. The sketch compaction algorithm runs in time $O(n^4)$, while its average-case performance is probably $O(n^3 \log^2 n)$ or better.

My algorithms employ two important data structures that help in computing the properties of cuts. One—the rubber-band equivalent of a sketch—is geometric, and
applies only to straight cuts. It supports fast computation of crossing sequences and congestion via scanning. It handles straight cuts only. The other—the adjacency graph of a sketch—is topological, and handles arbitrary cuts. It supports somewhat slower computation of congestion via graph search.

To justify and explain the sketch algorithms, I have developed a body of definitions and theorems that I refer to (somewhat pretentiously) as a theory of single-layer wire routing. In reality it is a partial theory of two particular models of single-layer routing: the sketch model and the design model. The centerpieces of the theory are two theorems that characterize the routability of a design and the optimal routing of a routable design in terms of the attributes of certain cuts and half-cuts. The design routability theorem states that a design is routable if and only if its major straight cuts are safe. The design routing theorem shows that every routable design can be routed so as to minimize the length of every wire, and it characterizes the optimal routing of each wire as the minimum-length route of that wire whose nontrivial straight half-cuts are safe. My proofs of these theorems give the first mathematically sound foundation to single-layer wire routing in multiply connected regions. I have also shown how to carry the design theorems over to the sketch model, and thereby provide the first rigorous treatment of algorithms for wire routing with homotopy constraints.

My treatment of the design model introduces a powerful tool for analyzing one-layer routing: lifting to a covering space of the routing region. None of the problems I encountered in studying the design model required me to step outside this framework. Covering spaces allow one to formalize and study many deceptively simple-looking concepts that play key roles in the routing problem. Some of these notions—flow, necessary crossings, and barriers—made sense when routing in simply connected regions, and via covering spaces can also be applied to designs. Others, such as the relations of respect and degeneracy between cuts and designs, have no such counterparts, and yet to overlook them can lead to disaster. (Before I hit upon the right definitions for these concepts, they caused persistent problems.) The use of covering spaces also allows one to treat cuts that are not simple, which opens the possibility of analyzing half-cuts in terms of their associated cuts.

Applications to practical problems

After all the algorithms are presented, analyzed, justified, and extended, the question of their practical utility remains. Not having implemented them in any form, I cannot answer with certainty. Nevertheless a short reply is possible. The sketch algorithms are limited in their practical applications mainly by the ability of sketches to represent active devices in integrated circuits. Where design rules are simple, the sketch algorithms—or algorithms derived from the same ideas—show
promise. Compared to modeling, performance is less of an issue. I have presented several ideas for improving the average-case behavior of the sketch algorithms, and there are undoubtedly many more to be found by implementing and experimenting with them. Moreover, the constants in their time and space bounds are small.

Most practical routing problems involve multiple layers, and do not specify the topology of the routing. I have little to say about such problems; my results concern provably efficient (polynomial-time) algorithms, while the problems of greatest practical interest are all NP-complete. Nonetheless, there are several ways in which my results might be applied to multilayer routing. Most obviously, a multilayer routing problem can be reduced to single-layer problems by first choosing rough routings of wires, and then assigning them to layers and placing the vias where they change layers. Good heuristics are known for the first problem, which is called global routing. Unfortunately, few heuristics are known for layer assignment and via placement (but see [41]).

A more robust approach allows for local topological changes, such as moving a wire across an obstacle or another wire. Starting with an infeasible assignment of wires to layers, one can identify the unsafe cuts, and try to shift wires away from them to reduce their congestion. This process is facilitated if on each layer the wires run in a preferred direction, as is common in integrated circuits and multilayer printed circuit boards. At least one PCB router was implemented using such a technique [9]. Its designer noticed that its “heuristic” method of testing whether a layer was routable, namely checking that no cut had greater congestion than capacity, never seemed to err. This suggestive piece of experimental evidence provided the initial motivation for my work.

Although this dissertation emphasizes routing, my compaction algorithm is considerably more powerful than my routing algorithm, and is more likely to prove useful. The reason is simple. In compaction, one may reasonably assume that the topology of the layout and the layer assignment are given, while in routing, most of the problem lies in choosing a proper topology and layer assignment. This observation makes the theory of routing no less important, however, for the compaction algorithm uses the routing algorithm as a subroutine.

The sketch compaction algorithm may also have applications to routing problems. The idea, which has been tested on channel routing with excellent results (see Acknowledgements, in the Preface), is as follows. One first expands the layout so that a conventional routing program, which may have difficulty with crowded layouts, can succeed. One then applies a compactor with the ability to insert arbitrary jogs in order to compact the layout to its proper size.

Related work

This thesis synthesizes and generalizes results from three primary sources. One
is the paper by Tompa [52] that solves the problem of river routing in a rectangular channel with terminals along its top and bottom. This paper first introduced the notion of the barriers for a wire. It showed that every routable channel can be routed by choosing a minimum-length barrier-avoiding routing for each wire. That demonstration formed the outline for my proof that the ideal embedding of a design is safe. Another source is the algorithm of Leiserson and Pinter [22] that uses routability conditions to compact a river routing channel horizontally. The third source is the paper of Cole and Siegel [6] that solves Pinter’s problem called ‘DRH’ [41], which is essentially the sketch routability problem in the grid model. That paper first claimed the equivalence of routability and safety—what I would call the sketch routability theorem for grids—but without a detailed proof. I relied upon that result in an earlier paper [21] when my own attempts at proving the routability theorem were failing, but my present proof is independent of their result. The algorithms and theorems in this thesis subsume those in the sources just mentioned, but of course the special-case algorithms are faster and easier to prove correct.

Recently Storb et al. [33] have developed an algorithm that appears to solve the sketch routing problem in an arbitrary wiring norm. Building on the algorithms for constructing the rubber-band equivalent of a sketch and for testing routability, they propose a routing algorithm for the euclidean wiring norm with a worst-case execution time of $O(n^3)$. If this result is borne out, it will complement my routing algorithm, which is faster (time $O(n^2 \log n)$) for polygonal wiring norms and slower (time $O(n^4 \log n)$) for other norms. Their method involves sweeping through a with scan lines perpendicular to each rubber band, constructing barriers and routing through them as in [52]. They, like I, call upon a simply connected covering space to understand how separate parts of a wire interact. How difficult their algorithm is to implement how complicated a correctness proof it will need, and how quickly it can solve problems of practical size—the answers to these questions are as yet unknown. The same authors point out that if the input to the routing problem includes not only rough routings of the wires, but also a planar graph in which their realizations must run, then the problem of routing with minimum total wire length becomes NP-complete. (Wire length can be minimized in certain fixed graphs, such as grids.)

All the works I have just described refer explicitly or implicitly to the dependence of routability on the congestions and capacities of cuts. This phenomenon occurs in other routing problems as well, notably the problem of routing edge-disjoint paths in a planar graph between pairs of terminals on its outer face [3, 17, 32]. The algorithms for this problem and its various special cases are all derived from a theorem [40] concerning the existence of such paths. That theorem states that there exist edge-disjoint paths connecting the terminals if the free capacity (margin) of every cut is nonnegative and even. (Here a cut is not a path, but rather a partition.
of the vertices into two sets.) This routing problem is rather different in character from mine, however, because it allows paths to cross.

B. Directions for Future Research

I close with a critical look at my models and a discussion of open problems. The deficiencies of the sketch model, in particular, suggest several directions for further investigation. My conclusion is that a great deal of work remains to be done in all the areas I have touched upon—the application of topology to prove routing theorems, the design of efficient algorithms and data structures, and the implementation of those algorithms.

A critique of the sketch model...

The many extensions presented in Chapter 10 may convince some readers that the sketch model is a robust one, and in some ways it is. But the reader who has looked carefully at the amount of work needed to justify the extensions may come to a different conclusion. Indeed, one might question whether my treatment of the unadorned sketch model is adequate, given the amount of handwaving in Section 8C. All my results concerning sketches rest not only on the detailed theory of the design model, but also on a complicated limiting process that relates this model to the sketch model. Three of the extensions I have proposed—elements of differing materials, curvilinear sketch elements, and multiterminal nets—involve strengthening both supports of the sketch results. What’s worse, there is no sure way to tell whether these extensions would somehow interfere, except by carrying through the proofs in as general a model as possible. Though generalizing a theory is usually much easier than constructing it from scratch, in this case the sheer mass of technical material makes it a daunting task. Still, in my opinion there is no substitute for a rigorous correctness proof of an algorithm, except perhaps an extensively tested, practical implementation.

But there are troubles on the practical side as well. Because sketches are so abstract, there is no straightforward way to convert a more conventional representation of a integrated circuit layer into a sketch for the purpose of routing or compaction (although the reverse is easy). On the other hand, even with all envisioned extensions, sketches may be too simple to serve as the primary geometric denotation of circuit layers in a CAD system. Even if they could, their ability to represent devices and multiterminal nets is disturbingly inflexible. We saw the difficulty of representing transistors in Section 10A, and the problems with multiterminal wires were made apparent in Section 10D. Sketches make more sense as an abstraction
of the interconnections among large circuit blocks than of the wiring within "leaf cells".

To be fair, the sketch abstraction may be perfectly adequate for printed circuit board layers. The components of a PCB layer, terminals (vias) and traces, are easily identified and correspond directly to sketch elements. More permissive wiring rules than grid-based rules are common, and fit nicely into the sketch model. Finally, optimal treatment of multiterminal nets may not be crucial. But except for the sketch routability theorem and its implications for routability testing, the application of my results to circuit boards is not particularly interesting. The most powerful sketch algorithm is Algorithm C, and it would find little use in compacting circuit board layouts, since modules (in this case, chips) are not generally free to move. In any case, their placement is dictated more by the physical volume they occupy above the routing layers than by routability constraints.

...and an apologia

So the sketch model sits in an awkward position. It is more abstract and simplistic than a practical user would like, yet it is mathematically inconvenient. For these reasons I have come to regard the sketch model as deficient. Why, then, did I choose it? The reasons are mainly historical. It was a natural outgrowth of the grid models previously in vogue among theorists, and I had previously published algorithms using it [21, 28]. These were the algorithms I set out to justify. I could have taken up these algorithms in the design model, for example, rather than the sketch model. In a sense the two models are on equal footing: while the design model is flawed for representing real designs, the sketch model flawed for mathematical analysis. But because the terminals in a sketch are points, the algorithms are simpler in the sketch model, and this fact tipped the balance. The algorithms dictated the model, not the other way around.

Of course, none of the flaws in the sketch model means that study of sketches cannot provide valuable insight into the issues of routing, routability testing, and compaction in more complex models. But if insight is all that we carry away, why spend so much time analyzing the technical aspects of one model? The answer is simply that some model must be chosen; the technical details come with the territory. Surely there are models better suited for practical use. But building a practical system was never an objective of my research. The real objective was to build technical tools, where none were previously known, for solving problems of single-layer routing, routability testing, and compaction in the presence of homotopy constraints. This I have done by developing the theory of routing in the design model, and showing how results from one model can be brought over to another.
Directions for future research

One way to build upon the ideas in this thesis would be to redevelop them in a model that avoids some of the defects of the sketch model. The sketch model is weak in two general areas: representing devices and modules in integrated circuits, and representing wiring structures like multiterminal nets and extended terminals. I see two corresponding directions for model changes: toward greater accuracy and faithfulness in representing practical circuits, and toward greater versatility and mathematical cleanliness. Regarding cleanliness, I have no solution to the problem that the use of covering spaces requires features to have positive diameter, which precludes the attachment of wires to terminals of the same size. Consequently, I see no good way to analyze the models I am about to discuss except by passing to an auxiliary model (like the design model) whose routing region is a manifold, and carrying back the results via convergence arguments. But we can hope to make this process easier, as in the network models of Section 10D.

As we noted in Section 10A, the sketch model has trouble representing the variety of different materials that interact on the lowest layers of a chip. The root of the problem is that the only objects that can coexist or coincide in a sketch are traces and their terminals. A real chip has many types of elements whose extents can overlap, not just wires and terminals. Typically a wire can overlap or approach regions in the device to which it is connected, but wires that are not connected to that device must stay away. Several good ideas for representing wires and devices in a simple and quickly accessible form may be found in [37]. Some of them might be incorporated into the sketch model. Most useful would be a way of relaxing the separation constraints among the parts of a device and the wires that connect to it.

Another possibility is to return to a view of modules as polygons with terminals on their boundaries. Point modules (isolated terminals) should also be permitted. One would have to confront directly the complicated problem of module interconnection: how to represent preexisting cells of a design in a compact form that facilitates routing and checking of design rules among modules. An advantage of solid modules over collections of scattered obstacles and terminals is that they can hide what might be design-rule violations in the context of routing, but are actually proper due to the function of the device. Module boundaries could include pointlike terminals, extended terminals, and other features. The layout should assign materials to all wires and features should be assigned materials, and could potentially mandate a different separation constraint between each pair of materials.

Two main technical issues would arise in a model based on modules: what paths should be considered cuts—paths landing too close to a terminal would not qualify—and what restrictions must be placed on the separation distances and the composition of module boundaries. The goal, of course, would be to prove routabil-
ity and routing theorems like those for sketches, and adapt the sketch algorithms to the new model. I contend that the concepts developed in the design model will be useful in new models as well. In particular, congestion can be measured by relating it to a flow-like quantity defined using covering spaces.

Open problems

In addition to the general problem of finding better models for single-layer wire routing, there are several specific questions that future research could aim to answer. Those concerning extensions of the sketch problems were discussed in Chapter 10; I list the most significant of them below.

- Can a single sketch incorporate wires of differing materials, and hence differing separation requirements?
- Can the sketch theorems and algorithms be enriched to provide for extended terminals and multiterminal nets?
- Does the network model (see Section 10D) support efficient algorithms for routing, routability testing, and compaction?

I conjecture that the answer is yes in each case. The remaining questions are larger and harder, and I make no predictions about them.

- How efficiently can the sketch routing problem be solved when the wiring norm is not polygonal?
- How fast can the sketch routing and compaction algorithms be made to run on practical examples? In what cases are they superior to algorithms that treat wires as objects to be moved?
- Can Algorithm C be extended to handle an unroutable initial configuration? How should the sketch compaction problem be defined in this case?
- How can routability constraints be applied to two-dimensional layout compaction? What data structures might be used for computing congestion when features were moving in all directions?
- What mathematical tools can assist the study of the network model? Does this models indeed support strong routability and routing theorems?

And for the mathematically adventurous, there is the following question:

- Can the results of the design model be generalized to higher dimensions? (Can one route surfaces in $R^3$ among toroidal terminals and obstacles?)

Further studies of routing problems with homotopy constraints, even those with little or no relevance to practice, may prove fruitful by clarifying the general principles at work.
Glossary

Since this thesis spans three areas that are only weakly related—topology, algorithms, and circuit layout—and introduces a substantial amount of new terminology, I have collected here the definitions of terms that are likely to be unfamiliar. Due to software limitations, however, I have not included pointers to the places where the terms are first used.

**absolute retract:** A space $A$ is an absolute retract if whenever a normal space $X$ has a closed subspace $B$ homeomorphic to $A$, then $B$ is a retract of $X$. The fact that $I$ and $R^1$ are absolute retracts follows from the Tietze Extension Theorem [38, p. 212].

**adjacency graph:** A data structure used by Algorithm C, the sketch compaction algorithm, for computing congestions of straight cuts. See Sections 7C and 9B.

**akin:** Subcuts are akin if their liftings connect terminals in the same way. See Definition 4d.1. Subcuts that are akin have link-homotopic associated cuts, and therefore share properties such as respect for a given design. Two crossings of links (or chains for links) are akin if those links can be lifted to reflect the crossings such that corresponding liftings share their terminals. Two plans (crossing sequences) are akin if they have the same length and corresponding crossings are akin.

**angle:** In Sections 7D and 7E, a point of the unit polygon of the wiring norm. The angle at which a path $\sigma$ travels is the normalization of the vector $\sigma(1) - \sigma(0)$, assuming that $\sigma$ is not a loop.

**arc length:** The arc length of a path $\alpha$ in the norm $\| \cdot \|$ is defined as follows. If $\alpha$ is piecewise linear, then its arc length is the sum over all its segments $\tau$ of the quantity $\| \tau(1) - \tau(0) \|$. Otherwise the arc length of $\alpha$ is the supremum of the arc lengths of the polygonal approximations to $\alpha$, that is, piecewise linear paths $\beta$ from $\alpha(0)$ to $\alpha(1)$ such that $\alpha(s) = \beta(s)$ for each joint $s$ of $\beta$. By default we measure arc length in the euclidean norm.

**arrangement:** An arrangement on a sheet $S$ is a finite set of disjoint simple cuts in $S$.

**article:** A connected set of details in a design: either a nonterminal fringe, or the image of a wire together with the wire's terminals.

**aspect ratio:** The ratio of a rectangle's longer dimension to its shorter dimension.
Glossary

**associated cut**: A cut formed from a half-cut or mid-cut by (1) extending it along its link(s) to form a link, and (2) applying a link homotopy to obtain a cut. In large measure, the associated cuts determine the properties of a half-cut or mid-cut.

**barrier**: In general, a barrier for a wire is a connected area that no feasible realization of the wire can enter. In the design model, a barrier is a subset of a forbidden zone that constrains the lifting of a wire, rather than constraining the wire directly. See Section 5A.

**base**: A fringe that contains the set from which a barrier grows (cf. Lemma 5a.4). Also, the range of a covering map.

**base point**: A distinguished point of a space where the loops that define its fundamental group begin and end.

**basis**: A basis for a topological space $X$ is a collection of open sets of $X$ that contains “arbitrarily small” neighborhoods of every point of $X$. Specifically, for every point $x$ of $X$, and for every open set $U$ containing $x$, the collection must include a neighborhood of $x$ lying within $U$.

**bent**: A bent path is a simple path having at most two segments.

**blanket**: A simply connected cover of a sheet.

**border**: A node of the adjacency graph borders the gates across which it has edges. It borders on a point $\alpha(0)$ in the direction of $\alpha$ if the first piece of the partition that $\alpha$ enters contains the region represented by that node.

**borders**: The borders of a piece $P$ in a pattern for the sheet $S$ are the components of $P \cap \text{Bd} S$. A border for the pattern is a border for any piece of the pattern.

**boundary**: I use this term only for manifolds. The boundary $\text{Bd} M$ of an manifold $M$ is the set of points of $M$ that have boundary patches. See Definition 2d.1. The boundary of an $n$-manifold is an $(n-1)$-manifold whose boundary is empty.

**boundary patch**: A patch $h: U \to H^n$ about a point $x \in U$ such that $h(x) \in \mathbb{R}^{n-1} \subset H^n$.

**boundary property**: A set of potential cuts $\Psi$ for a sketch $S$ has the boundary property if all the configurations in $C(S)$ that protect every potential cut of $S$ lie within some closed subset of $C(S)$.

**bounding obstacle**: In a sketch, a polygonal obstacle that encloses all the other features and the traces of the sketch. We add a bounding obstacle to a sketch for the purpose of relating it to designs.

**branch**: The branches of a design are the components of the inverse images of the design’s articles (under the covering map). Two fringes in a blanket are in the same branch if and only if a link connecting them is degenerate. Branches are to blankets what articles are to sheet.

**bridge**: In a sketch, a path whose endpoints lie on features but whose middle intersects no feature. Traces are bridges, and the cuts of a sketch are the images of linear bridges.

**bridge homotopy**: A piecewise linear homotopy between bridges that moves their endpoints along their respective islands and moves their middles through the routing
region. Two bridges are bridge-homotopic if there is a bridge homotopy that takes one to the other.

cable: In the RBE of a sketch, a group of rubber band segments (strands) with common endpoints.

CAD: Computer-aided design.

canonical: Also called ‘parameterized by arc length’. A path $\alpha$ is canonical if the euclidean arc length of each subpath $\alpha_{st}$ is just $|t - s|$ times the arc length of $\alpha$.

capacity: The capacity of a cut (or subcut) measures the amount of wiring space that the cut affords. It is defined as the arc length of the cut, measured in the wiring norm, decremented to account for the widths of the objects that contain the cut’s endpoints.

chain: Any path in a manifold is a chain; it contains zero or more links. A path for a path $\alpha$ is a chain that is path-homotopic to $\alpha$.

channel: A simply connected routing region, usually rectangular. Also used informally to mean the routing space between two islands in a sketch.

chip: See integrated circuit.

clean: Making crossings only at its endpoints. A path in a sheet is clean in a design if it intersects the articles of the design at its endpoints alone.

closure: The closure of a subset $A$ of a space $X$, denoted $\text{Cl} A$, is the minimal closed set of $X$ that contains $A$.

coherent: Simple links in a blanket are coherent if they lift wires (or routes thereof) in the same design. See Definition 4c.2.

collapsible: Given a design $\Omega$ and an arrangement $\Gamma$, we say that a deviation $\omega_{st}$ of a wire in $\Omega$ across a subpath $\gamma_{a,b}$ of a cut in $\Gamma$ is collapsible if $\omega_{st}$ is clean in $\Gamma$ and $\gamma_{a,b}$ is clean in $\Omega$.

compact: Compactness is a very important topological property. A topological space is compact if every collection of open sets that covers the space has a finite subset that also covers the space. The compact subspaces of $\mathbb{R}^n$ are the closed and bounded sets.

compaction: See sketch compaction and layout compaction.

component: Also called ‘connected component’. The components of a topological space are its maximal connected subspaces. Two points of a space $X$ lie in the same component of $X$ if some connected subspace of $X$ contains both points.

computational geometry: The study of algorithms that manipulate geometric objects.

concatenation: Formally, the concatenation of two paths $\alpha$ and $\beta$ is the path $\gamma = \alpha \star \beta$ such that $\gamma_{0:t} = \alpha$ and $\gamma_{t:1} = \beta$, with $t = \frac{1}{2}$. Informally, we allow $t$ to be any point in $(0, 1)$.

condensed RBE: A form of the rubber-band equivalent in which the strands within each cable are not represented as separate entities. Instead, the condensed RBE records only the total width of the strands within each cable.
configuration: A vector of horizontal displacements of the modules in a modular sketch. The configuration \( \mathbf{d} = (d_1, \ldots, d_n) \) for a modular sketch \( S \) corresponds to a sketch \( S(\mathbf{d}) \) in which module \( i \) has been shifted right by a distance \( d_i \).

configuration space: The configuration space of a sketch is the set of configurations that preserve its topology. See Section 9A.

conform: A link \( \omega \) conforms with an arrangement \( \Gamma \) if for every cut \( \gamma \in \Gamma \), every crossing of \( \gamma \) by \( \omega \) is necessary and no two are similar. A design \( \Omega \) conforms with \( \Gamma \) if every link in \( \Omega \) conforms with \( \Gamma \).

congestion: The congestion of a cut in a layout measures the minimum amount of wiring that must cross the cut, regardless of how the wires (or traces) are routed. In most cases this quantity is equal to the flow across the cut.

connected: A topological space is connected if it cannot be partitioned into two disjoint, nonempty open sets. See path-connected.

constraint: In the context of compaction, an inequality relating the positions of two modules.

constraint graph: A edge-weighted, directed graph in which each vertex denotes a variable and each edge denotes a simple linear inequality between two variables. If \( x_k \) is the variable represented by vertex \( k \), an edge of weight \( a_{ij} \) from vertex \( i \) to vertex \( j \) represents the constraint \( x_j - x_i \geq a_{ij} \). By computing longest paths in the constraint graph, one can assign values to the variables so as to satisfy the constraints.

content: The sequence of wires, traces, or rubber bands that a cut necessarily crosses.

contractible: A space is contractible if it can be shrunk to a point within itself. The homotopy that does this is called a ‘contraction’. Contractible spaces are simply connected.

convex: A subset \( X \) of \( \mathbb{R}^n \) is convex if for every pair of points in \( X \), the line segment between them also lies in \( X \). A function \( f: X \to \mathbb{R}^1 \) is convex if \( X \) is a convex subset of \( \mathbb{R}^n \) for some \( n \), and for every two points \( x, y \in X \) and every point \( t \in [0,1] \), we have

\[
f(tx + (1-t)y) \leq tf(x) + (1-t)f(y).
\]

convexity property: A set of potential cuts \( \Psi \) has the convexity property if for each \( \psi \in \Psi \), the capacity of \( \psi(\mathbf{d}) \) is a convex function of the configuration \( \mathbf{d} \).

convolution: The convolution of two planar regions is the set of all vector sums of a point one region with a point in the other.

corridor: A sequence of line segments, called doorways, through which a path must pass; the input to Algorithm W. See Section 1B.

covering map: This one is hard to explain. See Definition 2b.1 and Figures 2b-1 and 2b-2.

covering space: The domain of a covering map; also called ‘cover’. A space that looks locally like the space it covers, but whose parts may be connected together differently.

covering transformation: Also called ‘deck transformation’. A covering transformation is a homeomorphism of a covering space with itself that preserves the covering map. For
Critical: A critical cut in a sketch is one that begins at a feature endpoint and travels to the closest point (in the wiring norm) on another feature. The critical cuts of a sketch are decisive in the sense that their safety and emptiness determine the routability of the sketch.

A critical potential cut is a potential cut \( x_{PQ} \), where \( p \) is a feature endpoint and \( Q \) is a feature, such that \( x_{PQ}(d) \) is a critical cut from \( p(d) \) to \( Q(d) \) whenever one exists. See Definition 9c.1. Algorithm C uses critical potential cuts to generate routability constraints for sketch compaction.

crossing: Informally, a place where two paths meet. Formally, a pair \((s, t)\) such that the image of \( s \) under the first path equals the image of \( t \) under the second path. In the design model, we allow \( s, t \in [0, 1] \), while in the sketch model, we require \( s, t \in (0, 1) \).

crossing sequence: In the sketch model, the crossing sequence of a ray \( pq \) at \( p \) is the sequence of rubber bands that cross over that ray at \( p \), as defined in Section 1B. For the meaning of crossing sequence in the design model, see plan.

cross over: Two simple links in a blanket cross over if the image of one contains points in both scraps of the other.

curvilinear: Said of wiring norms: not piecewise linear.

cut: A path (or its image), often linear, used to test whether a layout is routable. The most important property of a cut is its safety or lack thereof.

cut: A simple link in a blanket cuts another link in the blanket if the terminals of the second link lie wholly one opposite sides of the first. See Definition 4b.1.

decisive: "Deciding routability". A set of cuts in a sketch is decisive if a sketch with the same set of features is routable if and only if all those cuts are either empty or safe in that sketch. By the sketch routability theorem, the critical cuts form a decisive set. A similar notion called "\( \tau \)-decisiveness" is defined for sheets; see Definition 6d.1.

deformation retract: A subspace \( A \) of a space \( X \) is a deformation retract if \( X \) can be shrunk down to \( A \) without moving any point of \( A \). The homotopy that does the shrinking is called a "deformation retraction".

degenerate: A cut is degenerate in a design if it is path-homotopic to a path in a single article of the design. This definition applies also to half-cuts and mid-cuts for wires in the design. More generally, a half-cut or mid-cut is degenerate if one of its associated cuts is degenerate. Degeneracy of cuts in sketches is similar, and is defined in Section 10C.

design: The more mathematical of my two basic representations of a circuit layer. Designs are defined in Section 4A. See also sketch.

design rules: Guidelines for the design of integrated circuits, intended to prevent unwanted behavior in the fabricated devices. For example, the design rules mandate a minimum separation between wires on the same layer, lest inaccuracies in the fabrication process cause the wires to short together.
Glossary

details: The details of a design are its wires and fringes.
detour: A detour of a link around a barrier is a link that does not intersect the barrier, and that is formed by splicing in pieces of barrier's frontier into the original link. See Definition 5a.5. To find an evasive route of a wire in a safe sketch, we lift it and make detours around the barriers for the lifting.
deviation: A subpath $\omega_{ast}$ of a wire is a deviation across a subpath $\gamma_{a:b}$ of a cut if $\omega_{ast} \simeq_P \gamma_{a:b} \cup \omega_{i:s} \star \gamma_{a:b}$ is a trivial link.
diagonal: Diagonal cuts are those that most strongly constrain the traces of a sketch; they have minimal capacity for cuts of their euclidean length. Formally, a cut is diagonal if its slope is diagonal and one of its endpoints is the vertex of a feature or fringe.
diagonal angles: The angles that correspond to the diagonal slopes.
diagonal slope: The wiring norm $\parallel \cdot \parallel$ defines the diagonal slopes: the slope of a line in $\mathbb{R}^2$ is diagonal if for every two points $p$ and $q$ on the line, $q$ is a vertex of the polygon $\{ x : \| x - p \| = \| q - p \| \}$. 
discrete: A topological space $X$ is discrete if every point of $X$ is open in $X$. For example, the integers form a discrete subspace of the real line.
divide: A planar region $X \subset \mathbb{R}^2$ divides a sheet if two fringes of that sheet fall in different components of $\mathbb{R}^2 - X$.
divisive: An article of a design is divisive if its extent divides the design's sheet. Divisive articles are undesirable, for they may represent unwanted loops in the layout.
dominant: A set of cuts in a sheet is dominant if it dominates the set of all nontrivial straight cuts in that sheet. Dominant cut set are $\mathfrak{d}$-decisive, by Corollary 6d.4.
dominate: One set of cuts in a sheet dominates another if every cut in the second set is either weak or can be reduced to a cut in the first set by a homotopy that does not increase its length. See Definition 6d.2. We exploit the relation of dominance to find small $\mathfrak{d}$-decisive cut sets.
doorway: In a safe sketch, each necessary crossing of a cut by a trace has a nonempty doorway. The doorway is the portion of the cut where a feasible realization of the trace may locate that crossing.
dual graph: The dual of an embedded planar multigraph is the graph whose nodes are the faces of that graph, and which has an arc between two faces for each edge of the original graph that borders on those faces.
ECE: See elastic-chain equivalent.
edging: An edging for a sheet $S$ is a finite set of convex polygons and line segments in $\mathbb{R}^2 - (S - Bd S)$ whose union contains $Bd S$. See Definition 6d.7.
elastic: A canonical path is elastic if it has minimum euclidean arc length among all paths in its path-homotopy class.
elastic-chain equivalent: A set of chains obtained from a design by replacing each wire in the design by the elastic chain for some route of the wire. In the "standard" ECE, one replaces each wire by its own elastic chain.
element: An element of a sketch is a feature or trace in the sketch.

embedded: An embedded planar graph is one that comes with a specific drawing in the plane.

embedding: A map that is a homeomorphism onto its image. Also refers to a wire that is link-homotopic to a given wire, or a design that results from “re-embedding” (routing) the wires in another design.

empty: A cut is empty if its flow is zero and its endpoints lie on the same fringe or island. Even if a empty cut is unsafe, we can ignore it.

enclose: A loop in the plane encloses a set if it cannot be shrunk to a point without touching that set.

entanglement: The entanglement of a wire (or trace) with a cut is the minimum number of crossings of the cut by any route for the wire (or trace). It counts the crossings that cannot be “untangled” by routing the wire. Compare winding.

equivalent: Two covering spaces of the same base space are equivalent if they are homeomorphic in a way that leaves the covering maps unchanged. See Proposition 2b.7. Two configurations of a modular sketch are equivalent with respect to a potential cut \( \psi \) if in moving linearly from one to the other, \( \psi \) is always a cut.

essential: Not path-homotopic to a constant loop. This definition is not entirely consistent with standard terminology, which defines ‘essential’ as “not homotopic to a constant map”. I do not need the latter concept, however.

euclidean: The euclidean norm \( \| \cdot \| \) is defined by \( |(x, y)| = \sqrt{x^2 + y^2} \).

evasive: Avoiding its barriers. In the design model, a route of a wire is evasive if it has no unsafe, straight, nontrivial half-cuts.

eventually: Section 8A defines for each suitably restricted sketch a family of sheets and designs parameterized by a positive real number \( \epsilon \). A statement involving \( \epsilon \) holds eventually if it holds for all \( \epsilon_0 \) less than some \( \epsilon_0 > 0 \).

exposed: A cut \( \alpha \) in a sketch is exposed if the corresponding cut \( \alpha^b \) eventually satisfies \( \| \alpha^b \| = \| \alpha \| - 2\epsilon \).

extent: Essentially a synonym for territory. The details of a design have extents, whereas the elements of a sketch have territories. Anyway, the extent of a detail of width \( d \) is the set of points closer than \( d/2 \) units to that detail, as measured in the wiring norm.

face: The faces of an embedded planar graph are the regions into which the edges of that graph divide the plane. The “outer” face is the unique unbounded one.

feasible: In general, a realization of a wire in a routing problem is feasible if it is part of a correct solution to the routing problem. Thus, an embedding of a wire in a design is feasible if some proper embedding of the design contains it.

feature: An inflexible object in a sketch. Every feature is a point or line segment.

flat: A n-manifold is flat if it comes with a local embedding into \( \mathbb{R}^n \). Flat manifolds include sheets, blankets, and scraps of blankets.
flow: The flow across a cut is a weighted sum of the necessary crossings of that cut by wires, where each crossing is weighted by the width of its wire. (Actually, flow counts equivalence classes of necessary crossings, rather than the crossings themselves.) Flow and congestion are equal for simple cuts, but flow is the deeper and more important of the two concepts. The notion of flow makes sense in all the routing problems I consider, although I define it formally only for the design model.

forbidden: Said of half-links in a blanket: contributing to a barrier. A half-link \( \sigma \) is forbidden to a wire lifting \( \bar{\omega} \) if to route \( \bar{\omega} \) through \( \sigma(1) \) would keep \( \omega \) from being evasive. See Definition 5a.1.

forbidden zone: The union of the left-hand or right-hand barriers for a wire lifting.

free: A path is free in a pattern if no seam in the pattern contains either endpoint of the path.

fringe: A component of the boundary of an \( n \)-manifold. A fringe is a path-connected \( (n-1) \)-manifold, closed in its parent manifold. The fringes of a sheet form the terminals and routing obstacles of the designs on that sheet.

frontier: The frontier of a subset \( A \) in a space \( X \), denoted \( Fr A \), is \( Cl A - Int A \): the set of points in the closure of \( A \) not in the interior of \( A \).

full plan: The full plan of \( \alpha \) in an arrangement \( \Gamma \) is the plan containing all the crossings of the cuts in \( \Gamma \) by \( \alpha \), sorted by position along \( \alpha \). It makes sense only when the crossings of \( \alpha \) in \( \Gamma \) are discrete.

fundamental group: An algebraic structure on the path classes of loops in a space at a given base point. See Definition 2a.3. The fundamental group of a space is an important topological invariant, part of the study of algebraic topology.

gap: A portion of a chain between two major links of the chain.

gate: A straight path forming part of a tunnel or a partition of a sketch.

gate arc: In the adjacency graph of a sketch, an arc representing adjacency across a gate.

gate list: The sequence of gates that a path crosses over, whether in the routing region of a sketch or in its adjacency graph.

graph: A mathematical structure comprising a set of "vertices", also called 'nodes', and a set of "edges", also called 'arcs', each of which is "incident" on exactly one or two vertices. Often the edges and vertices have additional information attached to them.

grid: The set of points in the plane which have at least one integral coordinate. The lines in this set are called 'gridlines', and the points where these lines intersect are called 'gridpoints'.

grid-based: Refers to a wiring model in which wires are constrained to run in a grid of horizontal and vertical lines.

half-cut: A half-link between a fringe and a route of a wire, used to measure the flow between the fringe and the wire.

half-link: A path \( \alpha \) in a manifold that touches the manifold's boundary at \( \alpha(0) \) only.

half-thread: The image of a simple half-link.
**Glossary**

**Hausdorff**: In a Hausdorff space, every two distinct points have disjoint neighborhoods. All the spaces I consider are Hausdorff.

**height**: The height of a potential cut $\phi_{pq}$, whose endpoints do not move vertically, is the difference between the $y$-coordinates of $p$ and $q$.

**homeomorphism**: A continuous, bijective function with a continuous inverse.

**homotopy**: A ‘continuous deformation’ or ‘continuous family’ of topological maps. See Definitions 2a.1 and 2a.6.

**IC**: See integrated circuit.

**ideal**: An ideal route of a wire is canonical, evasive, and as short as possible. Ideal embeddings are wires, and form a design; anything associated with this design is also called ideal. We route every wire in a safe design by means of its ideal embedding. By analogy with designs, we also apply the term ‘ideal’ to sketches; every trace in a routable sketch has an ideal realization, and these form a proper realization of the sketch.

**inner**: Said of fringes in a sheet: an inner fringe is one whose inside is not part of the sheet. Every sheet has at least one inner fringe. Compare *outer*.

**inside**: Every simple loop in a blanket or in the plane has an ‘inside’ and an ‘outside’. The inside of a blanket loop includes no part of any fringe.

**integrated circuit**: An electronic device made by depositing materials in and on a wafer of semiconducting material in precisely controlled patterns. Often called ‘chips’ or (in the popular press) ‘microchips’, integrated circuits are the computational elements at the heart of every modern digital computer.

**interior**: The interior of a subset $A$ of a space $X$, denoted $\text{Int} A$, is the maximal open set of $X$ contained in $A$.

**intersection graph**: The intersection graph of a sketch and a partition of that sketch is the graph whose nodes are features and the line segments where traces and gates intersect, and whose arcs are the subpaths of features and traces that connect these regions.

**island**: A maximal connected group of features in a sketch.

**jog**: A *joint* of a wire or trace.

**jog point**: A point at which at which a wire is allowed to develop a jog during compaction.

**joint**: A point (in the unit interval $I$) at which a piecewise linear path is not linear.

**kinship**: See *akin*.

**layout**: In general, the geometric structure of a circuit design. I use the term ‘layout’ to refer to an instance of a wire-routing problem, such as a sketch or design.

**layout compaction**: In general, the problem of minimizing the area of a circuit layout by altering its geometry.

**leaf cell**: The simplest modules in a VLSI design aside from transistors and other basic devices.
Glossary

lift: Also called 'lifting'. In the context of a covering map $p: M \to X$, a lift of a map $g: C \to X$ is any map $\tilde{g}: C \to M$ such that $p \circ \tilde{g} = g$. Outside of Chapter 2, the covering map $p$ is always taken to be the covering of a sheet by its blanket.

lifting: The process of converting maps into a base space into maps into its covering space.

line segment: A line segment is the image of a straight path.

linear programming: A classical and very important optimization problem: maximize a given linear function of real-valued variables subject to specified linear inequalities. Linear programming is solvable in polynomial time.

link: A path in a manifold that touches the manifold's boundary at its endpoints alone.

link code: The sequence of cuts in an arrangement necessarily crossed by a link or a chain for a link. See Definition 7b.1.

link homotopy: A homotopy between links that moves their endpoints along their respective fringes; or the relation of being link-homotopic. Two links are link-homotopic if there is a link homotopy (in the first sense) between them.

link plan: A sequence of crossings that a link (or a chain for a link) is forced to make with cuts in an arrangement, given that its link class is fixed. See Definition 7b.1.

list: A sequence of paths that a given path crosses over. See, for example, seam list.

local: A property of topological spaces is usually said to hold locally in a space $X$ if it holds within arbitrarily small neighborhoods of every point of $X$. (For properties that open sets do not normally have, such as compactness, the definition has to be modified somewhat.) For example, a space is locally path-connected if it has a basis of path-connected sets.

local embedding: The map $f: X \to Y$ is a local embedding if $X$ has a basis of open sets $U$ such that $f|_U$ is an embedding.

local homeomorphism: The map $f: X \to Y$ is a local homeomorphism if $X$ has a basis of open sets $U$ such that $f(U)$ is open in $Y$ and $f|_U$ is an embedding.

locally minimal: A linear path between two fringes of a sheet is locally minimal if its length (in the wiring norm) cannot be reduced by moving its endpoints along their respective fringe edges. The path need not be a chain; it can leave the sheet.

loop: A path whose endpoints coincide. A loop of $k$ links is...

major: Neither empty nor degenerate (said of cuts and links).

manifold: A topological space that is locally homeomorphic to $\mathbb{R}^m$ for some $m$. See Definition 2d.1.

margin: The margin of a cut is the difference between its capacity and its flow. Safe cuts are those with nonnegative margin (of safety). A subcut whose margin is zero is called 'marginal', or 'marginally safe'.

maze: A collection of tunnels, indexed by pairs $\pm \delta$ of diagonal angles, which begin and end at the same points. Every gate in the tunnel corresponding to the angles $\pm \delta$ must be a subpath of a linear path of angle $\pm \delta$. 

- 348 -
Glossary

**metric**: Also called 'distance metric', a metric on a set $P$ is a function $d$ from $P \times P$ to the nonnegative real numbers, satisfying three axioms: (1) $d(p, q) = 0$ if and only if $p = q$; (2) $d(p, q) = d(q, p)$ for all $p, q \in P$; and (3) the triangle inequality, $d(p, q) + d(q, r) \geq d(p, r)$ for all $p, q, r \in P$. The metric $d$ gives rise to a topology on $P$; a basis for this topology is the collection of sets $\{ q : d(p, q) < \epsilon \}$ for $p \in P$ and $\epsilon > 0$. In other words, a subset $S$ of $P$ is open in the metric topology if for every point $p \in S$ there is a number $\epsilon > 0$ such that $S$ contains the set $\{ q : d(p, q) < \epsilon \}$.

**metric space**: A topological space whose topology is given by a metric. All the spaces considered in this thesis are metric spaces.

**mid-cut**: A mid-link between two routes of wires, or between two points on the same route.

**mid-link**: A path in a manifold that does not intersect the manifold’s boundary.

**middle**: The middle of a path $\alpha$ is the set $\alpha((0, 1))$.

**minimal**: A minimal path from a compact region $P$ to a compact region $Q$ is a linear path from $P$ to $Q$ whose arc length, measured in the wiring norm, is the distance $\|P - Q\|$ from $P$ to $Q$.

**minor**: Either empty or degenerate (said of cuts and links).

**modular**: A modular sketch is a sketch together with a grouping of its islands into modules.

**module**: A set of sketch islands that move as a unit during compaction.

**multigraph**: A graph in which each pair of nodes can have multiple arcs between them.

**multiply connected**: Not simply connected.

**necessary**: Informally, a crossing of a cut by a wire (or trace, bridge, or link) is necessary if it cannot be removed by applying a homotopy (of the appropriate type) to the wire. The design model provides a formal definition (4b.2).

**neighborhood**: A neighborhood of a point or set is an open set that contains it.

**net**: A set of terminals to be connected, or a wire that connects them. Usually appears as 'multiterminal net', to contrast with the usual two-terminal nets. In the sketch model a net is a loop of traces that do not cross over and enclose no features. In the network model a net is a loop whose terminals are the islands it encloses.

**network**: A collection of nonintersecting nets and islands; an instance of a proposed wiring model (see Section 10D).

**norm**: A norm provides a uniform way of measuring distances in a vector space. A map $\| \cdot \|$ from a vector space to the nonnegative real numbers is a norm if three conditions hold: (1) $\|x\| = 0$ if and only if $x$ is the zero vector; (2) $\|tx\| = |t| \cdot \|x\|$ for all vectors $x$ and real numbers $t$; and (3) $\|x + y\| \leq \|x\| + \|y\|$ for all vectors $x$ and $y$. The distance between $x$ and $y$ in the norm $\| \cdot \|$ is just $\|x - y\|$. See also wiring norm.

**normal**: Said of topological spaces. In a normal space, every two disjoint closed sets have disjoint neighborhoods. All metric spaces are normal.

**obstacle**: An island of a sketch that is not a terminal.
Glossary

ordering property: A property required of the sequence of potential cuts input to Algorithm A, my abstract compaction algorithm. See Section 9E.

outer: Said of a fringe in a sheet. Every sheet has exactly one outer fringe, within which the rest of the sheet lies.

outside: The outside of a simple loop in a space \( X \) consists of every point in \( X \) that is neither on the loop nor inside it.

partial realization: A partial realization of a trace is minimum-length path through the gates for that trace of a particular diagonal slope. Partial realizations are constructed and used by Algorithm T.

partial route: A partial route for a maze is a minimum-length path through one of the tunnels of the maze.

partition: A partition of a sketch is a set of straight, horizontal cuts in the sketch that slice each component of its routing region into simply connected pieces.

patch: A patch about a point \( x \) in an \( n \)-manifold is a homeomorphism of a neighborhood of \( x \) with an open set in the half-space \( H^n \).

path: A continuous function with domain \( I = [0, 1] \). See the beginning of Chapter 2 for definitions related to paths.

path class: An equivalence class under the relation of path homotopy.

path code: In general, the sequence of cuts in an arrangement necessarily crossed by a path. See Definition 7b.1. When the arrangement is a pattern, the path can be constructed by reducing the seam list of the path in that pattern.

path component: The path components of a topological space are its maximal path-connected subsets. Two points lie in the same path component of a space \( X \) if there is a path in \( X \) from one to the other.

path-connected: A topological space is path-connected if every pair of its points can be connected by a path. Every path-connected space is connected, but not vice versa. If a space is connected and locally path-connected, however, then it is path-connected.

path homotopy: A homotopy between paths that fixes their endpoints; or the relation of being path-homotopic. See Definition 2a.1. Two paths are path-homotopic if there is a path homotopy between them.

path plan: A sequence of crossings that a path is forced to make with cuts in an arrangement, given that its path class is fixed. See Definition 7b.1.

pattern: A set of straight cuts called seams that divide a sheet into simply connected pieces for the purpose of determining which paths are path-homotopic. See Definition 7a.1.

PCB: See printed circuit board.

piecewise: In general, a property holds piecewise for a map \( f: X \to Y \) if \( X \) can be "triangulated" (divided into simplices) such that \( f \) has this property when restricted to each simplex. See the next entry.
**Glossary**

**piecewise linear**: A map $f : X \to Y$ is piecewise linear if $X \subset \mathbb{R}^n$ for some $n$, and $X$ can be chopped into simplices (points, line segments, triangles, tetrahedra, etc.) such that $f$ is linear on each simplex, and only finitely many simplices meet at each point. The composition of piecewise linear maps is piecewise linear, and the inverse of a piecewise linear map is piecewise linear.

**pivotal**: The pivotal cuts in a sketch are the diagonal cuts and the cuts between feature endpoints. Like the critical cuts, they are decisive.

**PL**: An abbreviation for piecewise linear.

**placement problem**: A problem that involves positioning inflexible objects (modules) as well as flexible ones (wires).

**plan**: A plan for a path $\omega$ is a finite sequence of triples $(\gamma, a, t)$ such that $\gamma(a) = \omega(t)$. Usually the paths $\gamma$ are taken from some arrangement $\Gamma$. See also full plan, path plan, and link plan.

**planar**: A graph is planar if its vertices and edges can be drawn in the plane without crossovers.

**pointlike**: A pointlike feature in a sketch is one that intersects no other features in the sketch and consists of a single point.

**polygonal**: A subset of the plane is polygonal if it lies within the union of a polygon with its inside, and contains the inside of the polygon. A wiring norm is polygonal if the set of points of norm 1 is a polygon.

**potential cut**: A linear path between two features of a sketch that moves in a continuous manner as those features move, depending only on their relative position. In any particular configuration, a potential cut may or may not give rise to a cut; hence the name. See Section 9C.

**printed circuit board**: A support and connector for electronic devices, made by plating metal wires onto layers of insulating material.

**proper**: Representing a valid circuit layout: “design-rule correct”. A sketch is proper if its traces are self-avoiding, and whenever two elements of the sketch have overlapping territories, they are a trace and one of its terminals. The corresponding property of designs is denoted by the term ‘$\|^1$-proper’. A design is proper if its wires are self-avoiding and its articles have disjoint extents.

**protect**: A configuration $d$ protects a potential cut $\psi$ for the sketch $S$ if in the sketch $S(d)$, the path $\psi(d)$ is either a safe cut or not a cut at all.

**quotient space**: A space obtained from another by identifying or “gluing” some points to some others. Formally, $Y$ is a quotient space of $X$ if there is a surjective map $f : X \to Y$ such that the open sets of $Y$ are those sets $U \subseteq Y$ for which $f^{-1}(U)$ is open in $X$.

**rail**: A rail of a track $\omega$ is a segment of $\omega$ that is either (1) supported at only one end, or (2) supported at both ends by ties of the same slope.

**RBE**: See rubber-band equivalent.
Glossary

reachable: One sketch is reachable from another if it can be obtained from the other sketch by a continuous motion of modules and wires that shifts modules horizontally and maintains the routability of the sketch.

realization: A trace or sketch that is the result of a routing process; it may or may not be feasible.

rectilinear: Composed of horizontal and vertical segments. The rectilinear norm on $R^2$ is defined by $||(x,y)|| = \max\{|x|, |y|\}$.

reduced seam list: See path code.

reduced intersection graph:

reflect: Two paths in a blanket reflect a crossing between their projections if they make that crossing themselves.

region: Usually refers to a subset of the plane.

respect: A relation that may obtain between a cut (or subcut) and a design; see Definition 4c.1. Respect and weak respect (Definition 4c.6) are the main technical conditions that permit us to relate the flows of different subcuts. A half-cut or mid-cut respects a design (strongly or weakly) if all its associated cuts respect the design (strongly or weakly).

restrain: A sheet $S$ (or a gate $\gamma$) restrains a path $\alpha$ at $x$ if for all sufficiently small open intervals $(s,t)$ containing $x$, the path $\alpha(s) \triangleright \alpha(t)$ leaves $S$ (or fails to intersect $\text{Im } \gamma$).

restricted route: An alternate definition of partial route; see Section 7E.

retract: A subspace $A$ of a space $X$ is a retract if there is a map $f: X \rightarrow A$ that fixes every point of $A$. The map $f$ is called a ‘retraction’.

rigid: Straight, nondegenerate, and marginal (a property of subcuts).

river routing: Refers to wire-routing problems in which wires do not change layers. Thus all single-layer routing problems may be considered river routing problems, but I prefer to reserve the term ‘river routing’ for situations in which each component or layer of the routing region is simply connected.

roots: With respect to a pattern in which $\alpha$ is free, the roots of a path $\alpha$ are the borders of that pattern that contain the endpoints of $\alpha$.

rough routing: A path that indicates the path class of a wire to be routed.

routable: An instance of a routing problem (e.g., a sketch or design) is routable if it has a proper routing (realization, embedding). Similarly, a design is $\uparrow$-routable if it has a $\uparrow$-proper embedding.

routability conditions: Necessary and sufficient conditions for a layout to be routable.

routability property: A set $\Psi$ of potential cuts for a sketch $S$ has the routability property if (1) the failure of a configuration $d$ to protect all elements of $\Psi$ implies unroutability of $S(d)$, and (2) the routability of $S(d)$ is guaranteed if all configurations $t \uparrow d$ with $t \in [0, 1]$ protect all elements of $\Psi$. 

- 352 -
route: A route for a trace is any bridge, not necessarily a trace, that is bridge-homotopic to that trace. A route for a wire is any link, not necessarily a wire, that is link-homotopic to that wire.

route: Also refers to a path through a tunnel. If Ω is a tight track through a maze and δ is a diagonal angle, then the shortest path through the δ-tunnel of this maze is called the δ-route of Ω.

routing region: In an instance of a routing problem, the space through which the wires are to be routed.

rubber band: The rubber band for a trace in a sketch is the shortest path that is a limit of routes for that trace.

rubber-band equivalent: A standard form for a sketch; the input to my routing and routability testing algorithms. The rubber-band equivalent (RBE) represents the features of the sketch and the rubber bands that result from shrinking each trace to its minimum length. The RBE data structure is optimized for computing the sequences of necessary crossings of cuts in the sketch.

safety: The central concept in the routability theorems concerning single-layer routing. A cut is safe if and only if its congestion (or flow) does not exceed its capacity. (Where flow and congestion are both defined, we use flow to determine safety.) A sketch (or design) is safe (or \( f \)-safe) if and only if all its nonempty straight cuts are safe. A design is safe if and only if all its major straight cuts are safe.

scanning: A fundamental algorithmic technique in computational geometry. A scanning algorithm constructs its output by sweeping a scan line across the objects in its input, processing each object as it enters and leaves the scan line.

scrap: A simply connected, open submanifold of a blanket.

seam: One of the straight cuts in a pattern.

seam list: The sequence of seams in a pattern that a piecewise linear path crosses over.

segment: The segments of a piecewise linear path \( \alpha \) are its maximal linear subpaths \( \alpha_{st} \) with \( s < t \). Consecutive segments of a PL path can be collinear if the path is not canonical.

self-avoiding: A wire in a design is self-avoiding if its article does not divide the sheet. Similarly, a trace in a sketch is self-avoiding if its territory, together with those of its terminals, does not separate any two of the sketch’s islands. The requirement that wires be self-avoiding is one of the complications of wire routing in multiply connected regions.

semisimple: Semisimplicity is a desirable attribute of half-cuts and mid-cuts. The subpaths of a cut between its necessary crossings by wires are semisimple subcuts for those wires. All subcuts akin to these are semisimple as well. See Definition 4e.5.

separable: A separable space is one that has a countable dense subset.

settle: Section 8A defines for each suitably restricted sketch a family of sheets and designs parameterized by a positive real number \( \epsilon \). A function \( f \) of \( \epsilon \) settles at a function \( g \) of \( \epsilon \) if the equality \( f(\epsilon) = g(\epsilon) \) holds for all \( \epsilon \) less than some \( \epsilon_0 > 0 \).
Glossary

shadow: The shadow cast by a point \( r \in \mathbb{R}^2 \) with respect to a point \( p \) is the set of points \( q \) such that \( \|p - q\| = \|p - r\| + \|q - r\| \).

shadowed: A cut \( \overline{pq} \) in a sketch is shadowed if there is a point \( r \) on a feature of the sketch such that \( q \) is in the shadow of \( r \) with respect to \( p \).

sheet: The routing region for a design; the result of removing one or more (but finitely many) polygonal holes from a closed polygonal region in the plane.

side: A simple link in a blanket separates it into two scraps, one on its left and one on its right. These scraps are the two sides of the link.

similar: Two crossings between paths in a sheet are similar if the liftings that reflect one also reflect the other. Equivalently, the crossings are similar if the subpaths that connect them are path-homotopic. See Definition 4b.2.

simple linear inequality: In the context of linear programming, an inequality \( x_j - x_i \geq a_{ij} \) in which \( x_i \) and \( x_j \) are variables and \( a_{ij} \) is a constant.

simple loop: A piecewise linear loop that would be injective but for the coincidence of its endpoints.

simple path: A piecewise linear and injective path.

simply connected: A topological space is simply connected if (1) it is path-connected, and (2) every loop in that space can be continuously shrunk to a point. For a formal definition, see Definition 2a.4.

skeleton: The subgraph of an adjacency graph obtained by omitting gate arcs.

sketch: One of my two basic representations of a circuit layer, discussed in Section 1A.

sketch compaction: Given a routable sketch, the problem of finding and routing a reachable sketch of minimum width. See Section 9A.

sketch routability: The problem of determining whether a given sketch is routable.

sketch routing: Given a routable sketch, the problem of finding a proper realization that minimizes the euclidean arc length of every trace.

space: A topological space: a set with a system of neighborhoods (open sets) closed under finite intersection and arbitrary union.

span: A cut set \( \Gamma \) spans the sheet \( S \) if for some edging \( \Delta \) of \( S \) and for every two elements \( P, Q \in \Delta \) such that the minimal cuts from \( P \) to \( Q \) are all cuts in \( S \), the set \( \Gamma \) a minimal path from \( P \) to \( Q \).

stable: A design is stable with respect to an arrangement if wherever a wire in the design intersects a cut in the arrangement, it intersects transversely, crossing over the cut at that point.

starlike: Also called ‘star-convex’. A subset \( P \) of a flat manifold is starlike about \( x \in P \) if for every point \( y \in P \), the linear path \( x \rightarrow y \) exists and lies in \( P \). A convex set is one that is starlike about each of its points.

straight: A path in a flat \( m \)-manifold is straight if its projection to \( \mathbb{R}^m \) is linear and nonconstant.
**strand**: One *segment* of a rubber band. (Rubber bands are piecewise linear.)

**string**: A finite sequence over a fixed alphabet. Path codes and link codes are strings over a pattern.

**strut**: A rigid cut or half-cut around which a wire route bends. See Definition 5b.5 and Section 1D. The struts of an ideal embedding are the constraints that force it to be as long as it is.

**subcut**: A cut, half-cut, or mid-cut.

**sublink**: A subpath of a link; any path in a manifold whose middle does not intersect the manifold’s boundary.

**submanifold**: A subset of a manifold that is itself a manifold of the same dimension.

**subpath**: A subpath of a path $\alpha$ is any path of the form $\alpha_{st}$ for $s, t \in I$. The definition of $\alpha_{st}$ is $\alpha_{st}(x) = \alpha((1 - x)s + xt)$.

A *track* has certain special subpaths, called $\delta$-subpaths, for each diagonal angle $\delta$. A $\delta$-subpath of a track $\omega$ is a path $\omega_{st}$ with $s < t$ such that either $s = 0$ or $\omega$ has a tie of angle $\pm \delta$ at $s$, and either $t = 1$ or $\omega$ has a tie of angle $\pm \delta$ at $t$.

**subspace**: A subset $A$ of a topological space $X$ with the inherited topology: the open sets in $A$ are the intersections of the open sets of $X$ with $A$.

**substring**: A contiguous subsequence of a string.

**support**: A straight path $\sigma$ in $R^2$ supports a piecewise linear path $\omega$ at $s \in (0, 1)$ if $\sigma(1) = \omega(s)$ and $\omega$ turns toward $\sigma(0)$ at $s$. If $\omega_{ri}$ and $\omega_{st}$ are segments of $\omega$, we also say that $\sigma$ supports these segments.

**tangent**: A straight path $\alpha$ in $R^2$ is tangent to a straight path $\sigma$ if the line containing $\alpha$ intersects the polygon $P(\sigma)$ at $\sigma(1)$, but does not intersect inside($P(\sigma)$).

**taut**: A route of a wire is taut if it has a strut at each of its joints. Ideal routes are taut (Proposition 5b.6).

**taxicab**: The taxicab norm on $R^2$ is defined by $||(x, y)|| = |x| + |y|$.

**terminal**: In general, the terminals of a wire (or trace) are the fixed objects to which that wire must connect. The terminals of a link or half-link are the fringes that contain its endpoints.

**terminal merging**: Refers to a modification of the sketch model in which the terminals of each trace are permitted to have overlapping territories and to coalesce during compaction.

**territory**: The territory of an object (feature, trace, fringe, or wire) is a region of the plane that represents the space allocated to it on its layer. It accounts not only for the physical dimensions of the object, but also for the necessary separation between objects. In other words, it encapsulates the geometric design rules for that object; two objects are assumed to interact if and only if their territories overlap.

**thread**: The image of a simple link.

**tie**: A tie for a track $\omega$ is a straight path $\sigma$ whose angle is diagonal, and which supports two segment of $\omega$, both of which are tangent to $\sigma$. 

---

- 355 -
Glossary

tight: A piecewise linear path \( \alpha \) is tight in a sheet \( S \) if \( S \) restraints \( \omega \) at each of its joints. Similarly, \( \alpha \) is tight in a tunnel or maze if for each joint \( x \) of \( \alpha \), some gate in that tunnel or maze restraints \( \alpha \) at \( x \).

topological property: A property that is preserved by homeomorphisms; what topology is about.

trace: A flexible object in a sketch. The metallized lines on a printed circuit board are (or used to be) called ‘traces’.

trace arc: In the adjacency graph of a sketch, an arc representing adjacency across a gate.

trace homotopy: A bridge homotopy that fixes the endpoints of the bridge.

track: A piecewise straight path with a tie at every joint.

trivial: Of paths in sheets, path-homotopic to a path in a single fringe. In a sheet \( S \), a crossing \((c,r)\) of a cut \( \chi \) by a chain \( \rho \) is trivial if for some \( i,j \in \{0,1\} \) the path \( \chi_{it} \star \rho_{rj} \) homotopic to a path in \( Bd \ S \).

tubular neighborhood: An especially nice neighborhood of a simple sublink; see Definition 3b.3.

tunnel: A sequence of gates through which a path must pass. Through any tunnel there is a unique minimum-length path. Tunnels are similar to corridors, but are more precisely defined; see Definition 7d.5.

turning: A piecewise linear path \( \alpha \) turns at \( s \in (0,1) \) if \( \alpha \) has two segments \( \alpha_{s1} \) and \( \alpha_{st} \) which either overlap or form an angle. If \( \alpha \) is a link in a sheet, then \( \alpha \) turns at \( x \in \{0,1\} \) if it forms an acute angle with a fringe there. If \( \alpha \) is a path in the plane, it turns ‘toward’ some points and ‘away from’ others. If \( \alpha \) is a link in a blanket, it turns ‘toward’ one of its scraps and ‘away from’ the other.

uniform convergence: A sequence of functions \( \langle f_n \rangle \) into a metric space with metric \( d \) converges uniformly to a function \( f \) if for every \( \epsilon > 0 \) there is an \( N \) such that \( d(f(x), f_n(x)) < \epsilon \) for all \( n \geq N \) and all \( x \).

unit polygon: For a piecewise linear norm, the analogue of the unit circle: the set of vectors of norm 1.

unsafe: See safety.

via: A connection between wires on different layers of a chip or printed circuit board. In an integrated circuit, also called ‘contact cut’.

visibility graph: The visibility graph of a sketch is a function of the features in the sketch. Its nodes are the feature endpoints and its arcs are the features and the cuts between feature endpoints.

VLSI: Stands for Very-Large-Scale Integration; refers to the technology that allows millions of electrical devices to be fabricated on a single chip.

wall: When compacting a sketch horizontally, we assume that the sketch is bounded at the left and right by vertical lines. These lines are called walls, and are treated as features.
weak: If a straight cut in a sheet can be reduced to a straight chain by link and path homotopies that do not increase its length, and this chain contains either two or more links or an entire fringe edge, then the cut is weak. See Definition 6d.2. One may ignore weak cuts when testing routability.

weak respect: See respect (and Definition 4c.6).

web: A web of k threads is the image of a loop of k links in a blanket.

width: Every feature and trace in a sketch, and every fringe and wire in a design, has a width that indicates how much area it requires. See extent and territory. In the RBE of a sketch, a crossing sequence or a cable has a width equal to the sum of the widths of the rubber bands it involves.

winding: The winding of a cut and a wire is the number of similarity classes of necessary crossings between them. Winding is to entanglement as flow is to congestion.

wire: Something to be routed. In the design model, ‘wire’ has a technical meaning: a wire in a sheet is a simple link whose terminals are convex and inner.

wiring model: The set of rules (design rules and others) that determine how the wires in a routing problem may be routed. More specifically, the definition of what constitutes a proper solution of a routing problem.

wiring norm: Part of a wiring model: the norm || . || used to measure widths, extents, and separations of layout components, and the capacities of cuts. Normally the wiring norm is an arbitrary piecewise linear norm, which means that the set of points of norm 1 is a convex polygon.
References


References


References


References


