

**Man-Portable Power Generation Devices:
Product Design and Supporting Algorithms**

by

Alexander Mitsos

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Chemical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2006

© Massachusetts Institute of Technology 2006. All rights reserved.

Author
Department of Chemical Engineering
June 28, 2006

Certified by
Paul I. Barton
Professor of Chemical Engineering
Thesis Supervisor

Accepted by
William M. Deen
Professor of Chemical Engineering
Chairman, Committee for Graduate Students

Man-Portable Power Generation Devices: Product Design and Supporting Algorithms

by

Alexander Mitsos

Submitted to the Department of Chemical Engineering
on June 28, 2006, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Chemical Engineering

Abstract

A methodology for the optimal design and operation of microfabricated fuel cell systems is proposed and algorithms for relevant optimization problems are developed. The methodology relies on modeling, simulation and optimization at three levels of modeling detail. The first class of optimization problems considered are parametric mixed-integer linear programs and the second class are bilevel programs with non-convex inner and outer programs; no algorithms exist currently in the open literature for the global solution of either problem in the form considered here.

Microfabricated fuel cell systems are a promising alternative to batteries for man-portable power generation. These devices are potential consumer products that comprise a more or less complex chemical process, and can therefore be considered chemical products. With current computational possibilities and available algorithms it is impossible to solve for the optimal design and operation in one step since the devices considered involve complex geometries, multiple scales, time-dependence and parametric uncertainty. Therefore, a methodology is presented based on decomposition into three levels of modeling detail, namely system-level models for process synthesis, intermediate fidelity models for optimization of sizes and operation, and detailed, computational fluid dynamics models for geometry improvement. Process synthesis, heat integration and layout considerations are addressed through the use of lumped algebraic models, general enough to be independent of detailed design choices, such

as reactor configuration and catalyst choice. Through the use of simulation and parametric mixed-integer optimization the most promising process structures along with idealized layouts are selected among thousands of alternatives. At the intermediate fidelity level space-distributed models are used, which allow optimization of unit sizes and operation for a given process structure without the need to specify a detailed geometry. The resulting models involve partial differential-algebraic equations and dynamic optimization is employed as the solution technique. Finally, the use of detailed two- and three-dimensional computational fluid dynamics facilitates geometrical improvements as well as the derivation and validation of modeling assumptions that are employed in the system-level and intermediate fidelity models. Steady-state case studies are presented assuming a constant power demand; the methodology can be also applied to transient considerations and the case of variable power demand.

Parametric programming provides the solution of an optimization problem, the data of which depend on one or many unknown real-valued parameters, for each possible value of the parameter(s). In this thesis mixed-integer linear programs are considered, i.e., optimization programs with affine functions involving real- and integer-valued variables. In the first part the multiparametric cost-vector case is considered, i.e., an arbitrary finite number of parameters is allowed, that influence only the coefficients of the objective function. The extension of a well-known algorithm for the single-parameter case is presented, and the algorithm behavior is illustrated on simple examples with two parameters. The optimality region of a given basis is a polyhedron in the parameter space, and the algorithm relies on progressively constructing these polyhedra and solving mixed-integer linear programs at their vertices. Subsequently, two algorithmic alternatives are developed, one based on the identification of optimality regions, and one on branch-and-bound. In the second part the single-parameter general case is considered, i.e., a single parameter is allowed that can simultaneously influence the coefficients of the objective function, the right-hand side of the constraints, and also the coefficients of the matrix. Two algorithms for mixed-integer linear programs are proposed. The first is based on branch-and-bound on the integer variables, solving a parametric linear program at each node, and the

second is based on decomposition of the parametric optimization problem into a series of mixed-integer linear and mixed-integer nonlinear optimization problems. For the parametric linear programs an improvement of a literature algorithm for the solution of linear programs based on rational operations is presented and an alternative based on predictor-continuation is proposed. A set of test problems is introduced and numerical results for these test problems are discussed. The algorithms are then applied to case studies from the man-portable power generation. Finally extensions to the nonlinear case are discussed and an example from chemical equilibrium is analyzed.

Bilevel programs are hierarchical programs where an *outer* program is constrained by an embedded *inner* program. Here the co-operative formulation of inequality constrained bilevel programs involving real-valued variables and nonconvex functions in both the inner and outer programs is considered. It is shown that previous literature proposals for the global solution of such programs are not generally valid for nonconvex inner programs and several consequences of nonconvexity in the inner program are identified. Subsequently, a bounding algorithm for the global solution is presented. The algorithm is rigorous and terminates finitely to a solution that satisfies ε -optimality in the inner and outer programs. For the lower bounding problem, a relaxed program, containing the constraints of the inner and outer programs augmented by a parametric upper bound on the optimal solution function of the inner program, is solved to global optimality. For the case that the inner program satisfies a constraint qualification, a heuristic for tighter lower bounds is presented based on the KKT necessary conditions of the inner program. The upper bounding problem is based on probing the solution obtained in the lower bounding procedure. Branching and probing are not required for convergence but both have potential advantages. Three branching heuristics are described and analyzed. A set of test problems is introduced and numerical results for these test problems and for literature examples are presented.

Thesis Supervisor: Paul I. Barton

Title: Professor of Chemical Engineering

To Eva, Evangelia, Baggelio, and Baggelitsa

Acknowledgments

I would like to begin my acknowledgments by thanking my thesis advisor Professor Paul I. Barton. He provided me with innovative suggestions and he routinely engaged into detailed discussions with me, but also granted me with the freedom of pursuing my own ideas. Furthermore, he gave me the opportunity to collaborate with other researchers as well as supervise two undergraduate students in research projects. Needless to say, I am also grateful for his efforts in securing my financial support for the bulk of my PhD program.

I would like to express my gratitude for financial support for my PhD program, through the Arch Chilton Scurlock Fund, the Department of Chemical Engineering at MIT, and the Martin Family Society of Fellows for Sustainability. This work was supported by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the Army Research Office under Grant DAAD19-01-1-0566.

I am indebted to my thesis committee members, who showed a great interest in my research and were always available for guidance and advice in academic and professional matters. Professor George Stephanopoulos provided me with different perspectives and insights on the issues of process synthesis, from which I benefited a lot. I especially thank him for constantly reminding me to stay focused on the most important topics and to always question my approaches and final goals. Professor Jefferson W. Tester gave me a thorough understanding of energy issues at different scales, through classes and meetings, and with his questions influenced the presentation of the design methodology. Finally, Professor Klavs F. Jensen held a dual advisory role for me, primarily as the principal investigator of a Multidisciplinary University Research Initiative (MURI) at MIT studying hydrogen generation and electricity production at the microscale, and secondarily, as a thesis committee member. His guidance and support shaped directly and indirectly the design methodology.

The formulation of the methodology was greatly influenced by the interactions within the MURI group, and especially with Leonel R. Arana, Ole M. Nielsen and Steven Weiss. Also the interactions with professors and students at MIT, and particularly at the Department of Chemical Engineering, has taught me a lot about technical and nontechnical matters. I would also like to thank George John Gesslein II for his helpful comments regarding the implementation of rational operations.

The working environment at the Process Systems Engineering Laboratory at MIT (PSEL@MIT) is extremely stimulating. It is impossible to capture all interactions here, but I would like to at least express my gratitude to the senior members of PSEL@MIT who helped me a lot at the beginning of my thesis work, especially Adam B. Singer and Cha Kun Lee. Also I would like to thank Ajay Selot for his hard work administering the PSEL@MIT computers in the final years of my thesis work. I had the luck to participate in a variety of fruitful collaborations within PSEL@MIT. In particular, I had a head-start to my thesis project by the work on system-level modeling of Ignasi Palou-Rivera, a postdoctoral fellow. Over a period of two years I had the great opportunity to collaborate with postdoctoral fellow Benoît Chachuat on the intermediate-fidelity modeling; not only is the result of our collaboration reflected in the outcome of this thesis, but also I enjoyed a lot working with Benoît and learned a great deal of things. The work on bilevel programs in the final year of my thesis work benefited a lot from the interactions with Panayiotis Lemonidis and in particular I would like to thank him for his encouragement in pursuing this project. Also, I have greatly enjoyed our collaboration on relaxation based bounds for semi-infinite programs. Finally, I would like to thank Mehmet Yunt for his efforts to design portable power generation devices under variable power demand.

During my third and fourth year as a doctoral student, I supervised two undergraduate students in research projects during their sophomore year; both Michael M. Hencke and Ruth Misener combine great skills with hard work and enthusiasm. It was a pleasure to work with them and I want to thank them for their work and congratulate them for their achievements. The main focus of Michael's project was on extensions of systems-level modeling, but he also performed modeling at the computational fluid dynamic level; part of his work is directly incorporated in this thesis. Ruth performed numerical experiments for a class of Partial-Differential Algebraic Equations; her work is outside the scope of this thesis, but very important for transient considerations.

My many and good friends at the PSEL, the department of chemical engineering, MIT, Cambridge and the world are extremely important to me, but I find it impossible to give justice to their friendship in a few sentences. I therefore thank them collectively and express the hope that they know what they mean to me. It would be even harder to express my feelings and gratitude for my family, so I will not attempt this either.

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction and Overview | 23 |
| 1.1 | Overview | 23 |
| 1.2 | Product Design Methodology | 23 |
| 1.3 | Parametric Optimization | 25 |
| 1.4 | Bilevel Programming | 26 |
| 2 | Product Design Methodology for Micropower Generation | 27 |
| 2.1 | Introduction | 27 |
| 2.2 | Scope of Methodology | 31 |
| 2.3 | Literature Review | 33 |
| 2.4 | Methodology Overview | 34 |
| 2.5 | Product Specifications | 37 |
| 2.6 | System-Level Analysis | 40 |
| 2.6.1 | Alternatives Considered | 42 |
| 2.6.2 | Integrated Layout and Thermal Management | 48 |
| 2.6.3 | Chemical Equilibrium Considerations | 53 |
| 2.6.4 | Simulation-Based Case Studies | 54 |
| 2.6.5 | Parametric Optimization-Based Case Study | 76 |
| 2.7 | Detailed Modeling for Justification of Modeling Assumptions | 80 |
| 2.7.1 | Uniform Temperature at Steady-State | 81 |
| 2.7.2 | Uniform Temperature in the Transient Case | 84 |
| 2.7.3 | One-Dimensional Species Balance | 89 |
| 2.8 | Computational Fluid Dynamics for Geometry Improvement | 97 |

| | | |
|----------|--|------------|
| 2.8.1 | CFD Model | 99 |
| 2.8.2 | Reduced Model | 102 |
| 2.9 | Intermediate Fidelity Modeling | 103 |
| 2.9.1 | Modeling | 104 |
| 2.9.2 | Optimal Operation and Design | 109 |
| 2.10 | Conclusions | 115 |
| 2.11 | Future Work | 117 |
| 3 | Parametric Optimization | 121 |
| 3.1 | Introduction and Literature Review | 121 |
| 3.1.1 | Complexity of Parametric Optimization | 124 |
| 3.2 | Parametric Optimization for Resource Allocation in R&D | 126 |
| 3.3 | MILP Optimality Range | 127 |
| 3.3.1 | Range of Infeasibility | 129 |
| 3.3.2 | Classification of Optimality Region Formulations | 130 |
| 3.4 | Multiparametric Cost Vector Case | 134 |
| 3.4.1 | Theoretical Properties | 135 |
| 3.4.2 | Intersection-Based Algorithm for a Single Parameter | 140 |
| 3.4.3 | Multiparametric Intersection-Based Algorithm | 145 |
| 3.4.4 | Multiparametric Optimality-Region Algorithm | 154 |
| 3.4.5 | Multiparametric Branch-and-Bound Algorithm | 159 |
| 3.5 | General Case with a Single Parameter | 165 |
| 3.5.1 | Assumptions and Theoretical Properties | 165 |
| 3.5.2 | Parametric Linear Program | 167 |
| 3.5.3 | Parametric Mixed-Integer Linear Program | 180 |
| 3.5.4 | Implementation | 187 |
| 3.5.5 | Numerical Results | 189 |
| 3.6 | Conclusions | 190 |
| 3.7 | Future Work | 191 |
| 3.7.1 | Algorithmic Improvement | 191 |

| | | |
|----------|---|------------|
| 3.7.2 | Extension to Nonlinear Cost Vector Case | 193 |
| 3.7.3 | Extension to General Nonlinear Case | 196 |
| 3.7.4 | Extension to General Multiparametric MILP | 198 |
| 4 | Bilevel Programming | 203 |
| 4.1 | Introduction and Literature Review | 203 |
| 4.2 | Definitions | 205 |
| 4.3 | Reformulations | 207 |
| 4.4 | Optimality Requirement | 209 |
| 4.5 | Consequences of Nonconvexity in the Inner Program | 212 |
| 4.5.1 | Branching on the \mathbf{y} Variables | 212 |
| 4.5.2 | Upper Bounding Procedure | 215 |
| 4.5.3 | Lower Bounding Procedure | 218 |
| 4.5.4 | Complication in KKT Approaches: Multiplier Bounds | 224 |
| 4.6 | Algorithmic Development | 225 |
| 4.6.1 | Assumptions | 226 |
| 4.6.2 | Lower Bounding Procedure | 229 |
| 4.6.3 | Upper Bounding Procedure | 235 |
| 4.6.4 | Algorithm Statement | 237 |
| 4.6.5 | Convergence Proof | 240 |
| 4.6.6 | Implementation and Numerical Results | 250 |
| 4.7 | Conclusions and Future Work | 261 |
| 5 | Conclusions and Future Work | 265 |
| A | Modeling Details | 267 |
| A.1 | Appendix: Symbols Used | 267 |
| A.2 | Appendix: Nomenclature | 267 |
| A.3 | Appendix: Physical Properties | 269 |
| A.4 | Calculation of Energy Densities | 271 |
| A.5 | Component Models | 272 |

| | | |
|----------|---|------------|
| A.5.1 | Splitter | 272 |
| A.5.2 | Reactor and Burners | 273 |
| A.5.3 | Mixer | 273 |
| A.5.4 | Membrane | 274 |
| A.5.5 | Compressor | 274 |
| A.5.6 | Fuel Cells | 275 |
| A.5.7 | Flash for Separation of Fuel Cell Effluents | 285 |
| A.5.8 | Oxygen Generators | 286 |
| A.5.9 | Water Breathing Systems | 287 |
| A.5.10 | Hydrogen Generators | 287 |
| A.5.11 | Implementation and Convergence Scheme | 289 |
| A.6 | Reduced Model for Heat Exchanger | 290 |
| A.6.1 | Tube Model | 290 |
| A.6.2 | Slab Model | 291 |
| A.6.3 | Defining Unit Interactions | 292 |
| B | Parametric Optimization Test Set | 295 |
| C | Bilevel Optimization Test Set | 315 |
| C.1 | Original Examples | 315 |
| C.2 | Examples from Gümüs and Floudas [133] | 341 |
| C.3 | Examples from Sahin and Ciric [246] | 348 |

List of Figures

| | | |
|------|---|----|
| 2-1 | Comparison of state-of-the-art batteries with theoretical energy density of fuels in a perfect fuel cell at ambient temperature, in which all the Gibbs free energy of reaction is used to produce power. | 29 |
| 2-2 | Overall methodology. | 35 |
| 2-3 | Examples of interactions with experimental efforts. | 38 |
| 2-4 | Set of alternatives considered. | 43 |
| 2-5 | Conceptual difference between coupled (left) and non-coupled (right) process components. | 51 |
| 2-6 | Influence of heat losses and scale on the energy density | 60 |
| 2-7 | Effect of recycling on the energy density for different values of the compression parameter K_P | 61 |
| 2-8 | Effect of SOFC efficiency on the energy density for different conversion values | 62 |
| 2-9 | Comparison of hydride based and hydrocarbon based processes in terms of volumetric and gravimetric fuel energy density. | 64 |
| 2-10 | Comparison of hydrogen peroxide and compressed oxygen in terms of the volumetric (left) and gravimetric (right) system energy density. | 66 |
| 2-11 | Volumetric and gravimetric system energy density of hydrocarbon partial oxidation in combination with a SOFC as a function of mission duration and power output. | 67 |
| 2-12 | Effect of fuel combinations and layout options on gravimetric fuel energy density of an ammonia-cracking based process. | 69 |
| 2-13 | Effect of water recycling in a DMFC. | 72 |

| | | |
|------|---|-----|
| 2-14 | Effect of cooling load in a DMFC. | 73 |
| 2-15 | Effect of water recycling in the reforming reaction of hydrocarbons. | 74 |
| 2-16 | Methane as a portable fuel? | 76 |
| 2-17 | Set of alternatives considered for the parametric optimization case study. | 78 |
| 2-18 | Optimal gravimetric fuel energy density as a function of achievable fuel cell efficiency. | 80 |
| 2-19 | Geometry and temperature profiles for explicit modeling of catalyst support (left) and lumped model (right). | 84 |
| 2-20 | Temperature profiles for explicit and average modeling of slabs. | 85 |
| 2-21 | Reactor geometry and temperature profiles obtained by CFD simulation corresponding to reactor with and without catalyst support. | 86 |
| 2-22 | Stack with a heating element in the top | 87 |
| 2-23 | Stack with a heating element in the middle | 87 |
| 2-24 | Two-dimensional model with heating element on top. | 88 |
| 2-25 | Three-dimensional model with heating element in the middle. | 89 |
| 2-26 | Concentration profile from FEMLAB at axial position 0.1 | 95 |
| 2-27 | Transient profile of molfraction at early time at the reactor middle for different grid sizes (without reaction). | 96 |
| 2-28 | Comparison of conversion at the outlet as a function of the reactor temperature for the different models. | 98 |
| 2-29 | Geometry of the heat exchanger (not to scale) | 99 |
| 2-30 | Velocity profile for an inlet velocity of 1 m/s. | 101 |
| 2-31 | Temperature profile for an inlet velocity of 1 m/s. | 101 |
| 2-32 | Temperature profile for an inlet velocity of 0.01 m/s. | 102 |
| 2-33 | Conceptual process flowsheet. | 105 |
| 2-34 | Ratio between heat losses and electrical power (left plot) and optimal design parameters (right plot) as a function of the operating temperature, for $PW = 1W$ | 113 |

| | | |
|------|---|-----|
| 2-35 | Fuel energy density (left plot) and fuel cell efficiency (right plot) as a function of the operating temperature, for literature exchange current density values. | 113 |
| 2-36 | Fuel energy density (left plot) and fuel cell efficiency (right plot) as a function of electrolyte thickness. | 114 |
| 3-1 | Graphical illustration of one-dimensional intersections-based algorithm for the parametric optimization in the cost vector case of (3.12). Solution I (magenta) corresponds to $\mathbf{y} = (1, 0, 0, 0)$, solution II (cyan) to $\mathbf{y} = (0, 0, 0, 1)$, solution III (green) to $\mathbf{y} = (0, 0, 1, 0)$, and solution IV (blue) to $\mathbf{y} = (0, 1, 0, 0)$ | 144 |
| 3-2 | Graphical illustration of Algorithm 3.2 for example (3.13)). Blue corresponds to $\mathbf{y} = (1, 0, 0)$, green to $\mathbf{y} = (0, 1, 0)$, and cyan to $\mathbf{y} = (0, 0, 1)$. The algorithm requires 8 calls for 3 solutions and 8 vertices. Vertices in the parameter space are marked with a square when the optimal solution is not verified, and with a circle when it has. | 154 |
| 3-3 | Graphical illustration of the Algorithm 3.3 for example (3.13)). Blue corresponds to $\mathbf{y} = (1, 0, 0)$, green to $\mathbf{y} = (0, 1, 0)$, and cyan to $\mathbf{y} = (0, 0, 1)$. The algorithm requires 3 optimality region formulations for 3 solutions and 8 vertices. Vertices in the parameter space are marked with a square when the optimal solution is not verified, and with a circle when it has. | 159 |
| 4-1 | Inner level objective function, its convex envelope and its α -BB underestimator for example (4.6) | 221 |
| A-1 | Explanation of the symbols used. | 267 |
| A-2 | Stream Numbering | 292 |
| A-3 | Tube Numbering for Stream j | 292 |
| B-1 | Small micropower case study. | 300 |
| B-2 | Larger micropower case study. | 301 |

| | | |
|------|---|-----|
| C-1 | Inner level objective function for Example C.4. | 318 |
| C-2 | Inner level objective function for Example C.5. | 319 |
| C-3 | Inner level objective function for Example C.10. | 322 |
| C-4 | Inner level objective function, its KKT points and its minima for Example C.11. | 324 |
| C-5 | Minima and suboptimal KKT points for the inner problem of Example C.12. | 325 |
| C-6 | Minima and suboptimal KKT points for the inner problem of Example C.13. | 326 |
| C-7 | Minima and suboptimal KKT points for the inner problem of Example C.14. | 327 |
| C-8 | Minima and suboptimal KKT points for the inner problem of Example C.15. | 329 |
| C-9 | Minima and suboptimal KKT points for the inner problem of Example C.16. | 330 |
| C-10 | Minima and suboptimal KKT points for the inner problem of Example C.17. | 331 |
| C-11 | Minima and suboptimal KKT points for the inner problem of Example C.18. | 332 |
| C-12 | Minima and suboptimal KKT points for the inner problem of Example C.19. | 333 |
| C-13 | Minima and suboptimal KKT points for the inner problem of Example C.20. | 335 |
| C-14 | Inner objective function, its KKT points and its minima for Example C.21. | 336 |
| C-15 | Minima and suboptimal KKT points for the inner problem of Example C.22. | 337 |
| C-16 | Feasible set in the x_1, x_2 space for Example C.25. | 341 |
| C-17 | Equivalent objective function of Example C.26. | 342 |
| C-18 | Inner program of Example C.32. | 350 |

C-19 Equivalent objective function for Example C.32. 350

List of Tables

| | | |
|------|---|-----|
| 2.1 | Summary of the results for the chemical equilibria | 54 |
| 2.2 | Ideal energy densities | 56 |
| 2.3 | Process parameters for the comparison of processes | 57 |
| 2.4 | Process comparison for atmospheric air | 58 |
| 2.5 | Process comparison for compressed oxygen | 59 |
| 2.6 | Process parameters for the comparison in Figure 2-9. | 65 |
| 2.7 | Process parameters for the ammonia cracking case study, Figure 2-12. | 70 |
| 2.8 | Process parameters for water reforming study in Figure 2-15. | 75 |
| 2.9 | Process parameters for parametric optimization case study in Figure 2-18. | 79 |
| 2.10 | Parameter values for the steady-state model | 110 |
| 2.11 | Optimal operation and design results for $PW = 1$ W. | 111 |
| 2.12 | Maximizing energy efficiency vs. maximizing energy density | 116 |
| 3.1 | Computational requirements in seconds for the parametric LP and MILP. No distinction is done for CPU times less than 0.01s. | 189 |
| 4.1 | Summary of problem properties | 257 |
| 4.2 | Numerical results without branching | 258 |
| 4.3 | Numerical results with regular branching | 259 |
| 4.4 | Numerical results with special branching on the \mathbf{x} variables | 260 |
| A.1 | Nomenclature | 268 |

Chapter 1

Introduction and Overview

1.1 Overview

There are essentially three parts to this thesis, namely (i) an integrated design methodology for portable power generation based on fuel cell systems, (ii) algorithms for parametric mixed-integer programming, and (iii) an algorithm for the co-operative formulation of inequality constrained bilevel programs with nonconvex functions in both the inner and outer programs. There are sufficient reasons to warrant research in each of these topics on isolation and therefore thorough introductions, including motivation and literature review in the respective fields, are given in the following three chapters. The main purpose of this chapter is an overview of the connections between the three parts of this thesis.

1.2 Product Design Methodology

The widespread use of portable electric and electronic devices increases the need for efficient autonomous man-portable power supplies (up to about 50 W). Currently, batteries are the predominant technology in most applications. However, batteries have a large environmental impact, high cost and relatively low gravimetric (Wh/kg)

and volumetric (Wh/l) energy density. State-of-the-art primary batteries reach up to 1300 Wh/l and 700 Wh/kg and rechargeable up to 400 Wh/l and 300 Wh/kg, and the upper limit on performance is now being reached [183]. A promising alternative is to use common fuels/chemicals such as hydrocarbons or alcohols and there is great military [74] and civilian [116] interest in developing battery alternatives based on these fuels and portable fuel cell systems. In recent years microchemical systems have received special attention [146] and significant advances have been made. Chemical units such as reactors, separators and fuel cells with feature sizes in the submillimeter range have been considered for a variety of applications. Microchemical systems have several advantages compared to macroscale processes: the increased heat and mass transfer rates at the microscale allow higher yields [166]; and the small hold-up along with the controlled conditions allow reaction pathways deemed too dangerous for conventional processes; the small quantities required and the possibility of parallelization have sparked interest in micro-total-analysis-systems (lab-on-a-chip) [165]. Currently, most of the microreactors are not standalone devices, but rather are used within a conventional laboratory. The replacement of batteries for electronic devices requires truly man-portable systems and therefore the use of microfabrication technologies is plausible since a minimal device size is desired. Most of the research in micropower and microreaction technology has focused on fabrication techniques or detailed modeling, whereas there are still few contributions regarding design methodologies for such systems and this gives the broad motivation for the development of the design methodology.

The methodology proposed is based on decomposition into three levels of modeling detail, namely system-level models for process synthesis, intermediate fidelity models for optimization of component sizes and operation, and detailed computational fluid dynamics models for geometric design and justification of modeling assumptions. In Chapter 2 an overview of this methodology is given, followed by detailed descriptions of the various parts.

1.3 Parametric Optimization

Many microdevices and components of the process alternatives considered are not yet fully developed and characterized and therefore at all three levels of modeling there are several degrees of freedom as well as uncertain parameters. These parameters characterize the state of the technology considered or the performance of some components. For instance, at the intermediate fidelity modeling level, an uncertain parameter is the upper limit of the operating temperature, imposed by material properties which are not sufficiently-well understood to be included in the modeling. At the system-level examples of uncertain parameters include the thermodynamic efficiencies of the fuel cells and achievable selectivities in purification membranes.

For simulation-based approaches, both in static and dynamic problems, the effect of uncertain parameters is often captured with sensitivity analysis and parameter variation studies. For optimization-based methods post-optimality sensitivity analysis gives local information about the influence of the parameters, i.e., only the effect of an infinitesimally small parameter variation is captured. Parametric programming provides the solution of an optimization problem, the data of which depend on one or many unknown real-valued parameters, for each possible value of the parameter(s) and therefore can give global information, i.e., the influence of the uncertain parameters over a whole range of values is furnished.

Suppose in general that a model of a system under development with many components is given and the uncertain parameters describe the performance of the various components. The values of the parameters not only influence the performance of the system, but also the optimal design. Parametric optimization quantifies the influence of these parameters on the system performance and optimal design. This can help determine whether it is worthwhile to pursue improving the performance of a given component. Such questions of technology significance and resource allocation at the system-level motivates the development of the parametric optimization algorithms, described in Chapter 3. In particular the interest in mixed-integer programs arises from the fact that different technologies are considered and the choice between

alternative technologies is represented with integer variables.

1.4 Bilevel Programming

Bilevel programs are hierarchical programs where an *outer* program is constrained by an embedded *inner* program. Bilevel programming is used in macroscale process systems engineering for design under phase equilibrium [68] as well as for flexibility and feasibility problems, see, e.g., [47]. These formulations are of potential interest for man-portable power devices, but an application of these techniques is outside the scope of this thesis.

The main motivation to develop an algorithm for bilevel programs within this thesis is the inherent and well-known relation of bilevel programming and parametric optimization. In principle bilevel programs could be solved via parametric programming, by solving the inner program for all possible values of the outer variables. Recently this has been proposed for special convex cases [232], but in general it is not an advisable procedure, since obtaining the parametric global optimum would be very computationally expensive and in a sense provide more information than what is actually needed for solution of the bilevel program. The algorithm described in Chapter 4 only considers parametric upper bounds in a neighborhood of candidate optimal solutions of the bilevel program.

Chapter 2

Product Design Methodology for Micropower Generation

2.1 Introduction

The widespread use of portable electric and electronic devices increases the need for efficient autonomous man-portable power supplies [160, 97]. Portability limits the mass of the power generation system to a few kg and the volume to a few liters, at most, and consequently to power supplies of up to 50 W. Currently, batteries are the predominant technology in most applications. However, batteries have a large environmental impact, high cost and relatively low gravimetric (Wh/kg) and volumetric (Wh/l) energy densities. State-of-the-art primary batteries reach up to 1300 Wh/l and 700 Wh/kg and rechargeable up to 400 Wh/l and 300 Wh/kg [183, 55] and the upper limit on performance is now being reached as most of the materials that are practical for use as active materials in batteries have already been investigated and the list of unexplored materials is being depleted [97, 183].

There is both military [58, 160, 74, 222] and civilian [116] interest in alternative power generation. Many alternatives are in theory possible, such as electrochemical conversion of fuels in fuel cells, thermophotovoltaic cells [77, 283, 212], a microturbine driving a generator [103] or even exploiting nuclear power, e.g, with thermoelectrical elements [184]. Also, there are several approaches for producing energy by harvesting

human-based mechanical work, e.g., [80, 243, 175, 8].

Microfabricated fuel cell-based systems have attracted much interest in recent years because common fuels and chemicals, such as hydrocarbons or alcohols have very high energy contents (Figure 2-1) and fuel cell systems have the capability of achieving high efficiency, have few or no moving parts, and run silently. In order to achieve portability the use of microfabricated devices, as opposed to conventional devices, is plausible. In recent years it has become possible to fabricate many new unit operations at the microscale, and this number rises rapidly. However, only careful integration of these components can lead to a design that is competitive with existing technologies. Direct miniaturization of conventional systems is either impossible with current technology or leads to low energy densities, large parasitic losses and large start-up times. While systematic process synthesis and design is a mature field at the macroscale, microsystems exhibit a unique set of new challenges for process systems engineering. For example, at the microscale heat losses to the environment are a critical design consideration. The portability requirement, as well as the fact that the devices need to work fully automatically without the intervention of operators, also gives rise to many design constraints and safety issues. New design tools and evaluation methodologies are needed to address the challenges of microchemical systems.

There are two main approaches for fuel cell systems, namely direct fuel cells running on stored hydrogen, methanol, formic acid, or medium sized hydrocarbons, as well as fuel processing for hydrogen or syngas generation and subsequent oxidation of these intermediates in a fuel cell. Micropower generation devices based on either approach are products that comprise a more or less complex chemical process. There is a plethora of possible processes and process combinations, as well as a wide variety of applications and consumers, ranging from cellular phones and laptops for home use to the power needs of the dismounted soldier, thus it is plausible that the optimal device configuration will depend on the product specifications characterizing particular applications. This necessitates a flexible methodology for the comparison of different technology alternatives that can facilitate product engineering of these devices.

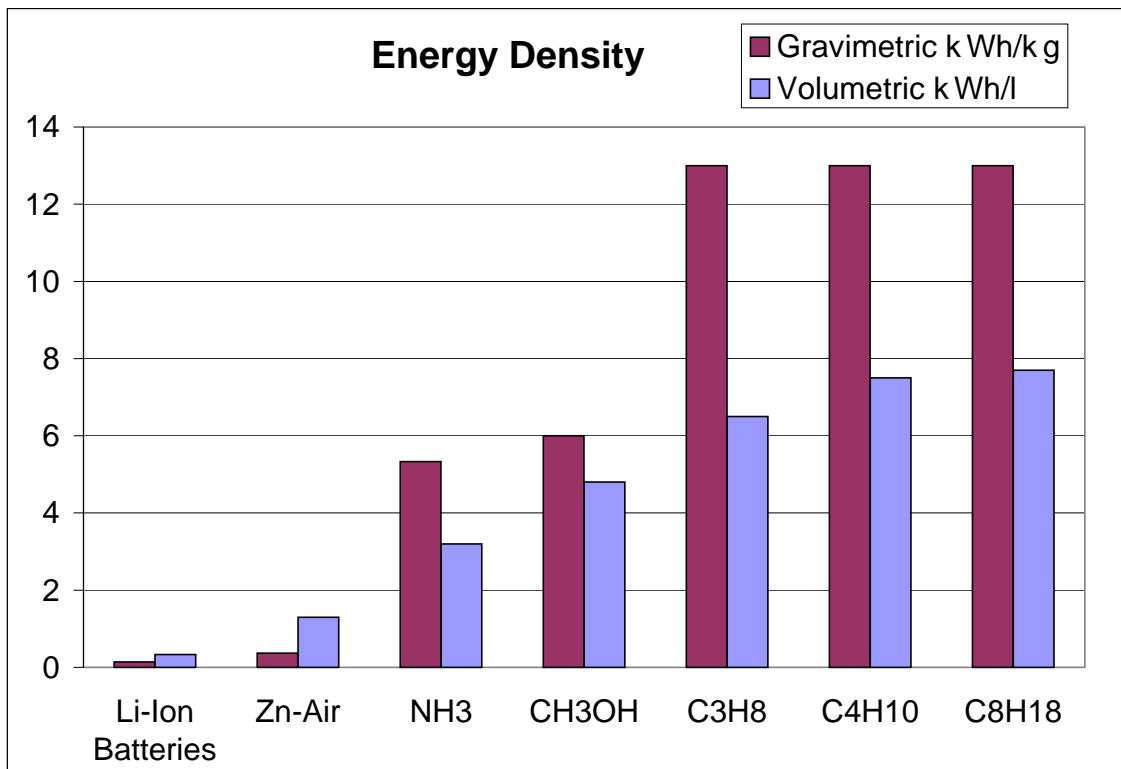


Figure 2-1: Comparison of state-of-the-art batteries with theoretical energy density of fuels in a perfect fuel cell at ambient temperature, in which all the Gibbs free energy of reaction is used to produce power.

The devices considered need to operate independently of external heat sources, despite, for example, the use of endothermic fuel processing reactions or high operating temperatures which lead to high heat loss fluxes. The simplest approach to provide the necessary heat is to use part of the fuel in a combustion or catalytic oxidation reaction, but a more promising approach is to use a fuel combination. The motivation is that one fuel can be used for reforming (hydrogen production) and another for combustion/heat generation. Using multiple fuels is of particular interest in the case that a low energy-density fuel (e.g., ammonia or methanol) is used for hydrogen production, especially when an endothermic reaction is used (e.g., cracking).

In larger scale power production, including the electric car, emphasis is placed on efficient utilization of the fuel. This is because the fuel cost is of the same or higher order of magnitude as the fabrication cost of the power production system. In portable power production the economical and ecological operating costs are insignificant com-

pared to the fabrication costs of the systems. Typically different man-portable power generation systems are compared using the energy density of the system as a metric. The gravimetric energy density, or specific energy [Wh/kg], is expressed as the electrical energy produced per unit mass of system [183] and the (volumetric) energy density [Wh/l] is defined as the electrical energy produced per unit volume of the system. Depending on the application, either of the densities is of greater importance. It is essential to define the system appropriately including the power generation devices as well as the fuel containers. The objective of maximal energy density is in general not equivalent to the objective of maximal efficiency. The simplest example illustrating this, is the comparison between different fuels; choosing a fuel with high energy density can lead to a higher system energy density despite a lower efficiency. For instance a 35% efficient butane system has a higher energy density than a 70% efficient ammonia system (see Figure 2-1). A similar behavior is seen for systems with a combination of different fuels/chemicals, such as the ammonia-butane example in Section 2.9, where energy density and energy efficiency bear different weights on each species. The extreme case of species combination is the addition of water in steam reforming reactions, which does not affect the energy efficiency but greatly reduces the energy density. For systems involving only one stored species, the argument is a little more elaborate. The fuel energy density and efficiency are proportional, but the system energy density is not; if heat losses are not limiting, it is conceivable that a complex device with high residence time will lead to higher efficiency than a simple device with low residence time, but at the cost of additional weight and volume. Also, system efficiency is not necessarily equivalent to component efficiency. For instance, operating a fuel cell near its open-circuit voltage minimizes the irreversibilities and therefore some metrics of efficiency, but also results in very low power density and a small system efficiency.

In addition to high energy densities an adequate portable power production process must be insensitive to transportation, and ideally work under changing orientation (upside-down) as well as in a variety of ambient conditions, including low and high temperatures. Especially for military and space applications extreme ambient

conditions are possible, such as immersed in water or vacuum. Since most power consuming devices are not operated constantly and have rapidly changing power demands, the dynamics and automated operation of portable power production are very important. The processes must operate fully autonomously, automatically and without any safety concern, such as the use or generation of toxic or dangerous materials. It is paramount in computing energy density to have a process that operates independently of external heat sources, despite, for example, the use of endothermic fuel processing reactions. For most applications the life cycle price is a serious consideration, especially for devices with relatively high power consumption, such as portable computers. The life cycle price includes manufacturing and refueling or recharging and eventual disposal/recycling of devices. Because of the widespread use of portable power production its environmental impact is substantial. In contrast to the macroscale, the impact is not associated with the power production per se, but rather with the materials used in devices and the fabrication and recycle/disposal processes. From a consumer point of view, the process must have a relatively simple way of recharging, refueling or replacing.

2.2 Scope of Methodology

There are several, often conflicting uses of the prefix micro. Traditionally, microreactor referred to laboratory-scale tubular reactors for catalyst testing [165]. With the advent of microfabrication technology a plausible use of the prefix micro is to refer to systems fabricated by these methods [165, 100]. Another strict definition is to only use the term micro for systems with a largest dimension of less than one millimeter. A more loose use of the term micro is to characterize microstructured systems, i.e., systems with some characteristic length in the micrometer range. The devices considered in this thesis have characteristic dimensions ranging from the submicron level for membrane thickness up to a few millimeters for the fuel cell length (inner dimension), while the overall system size including packaging is restricted by the size of the existing technologies, i.e., to centimeters at the most. Similarly, there is some

ambiguity in the terms micropower and portable power, which are sometimes used for residential distributed power generation and the electric car respectively. In this thesis we consider systems that are suitable for man-portable applications with power outputs in the order of 0.1-50W. The term optimization is used with various meanings in the literature. In this thesis it is used for methods based on mathematical programming, i.e., systems of equations with some degrees of freedom and one or more objectives, solved using computer implemented algorithms.

In this chapter an integrated design methodology for portable power generation based on fuel cell systems is proposed. The necessity for such product design is warranted due to the plethora of possible processes and process combinations, as well as the wide variety of applications and consumers, ranging from cellular phones and laptops for home use to the power needs of the dismounted soldier. The strong interconnection of design and operation (steady-state and dynamic) and the complexity of the systems lead to various counter-intuitive effects and therefore make a systematic design methodology employing mathematical models, simulation and optimization, as opposed to empirical design based on trial-and-error, necessary. Micropower generation devices can be considered chemical products, because they affect chemical change [202]. Unlike traditional chemical products, their characteristics do not depend on the molecular structure or microstructure but rather on the performance of the underlying chemical and electrochemical unit operations. Most of the methodologies proposed for product design, e.g., [202, 280, 79, 279], include a step identifying the customer's needs before inventing and analyzing alternative products that can fulfill these needs, and we briefly cover this in Section 2.5. The focus of this thesis is on the development of a methodology for generation of ideas and selection and optimization of the most promising alternatives. The final step in product design [202] is to analyze the manufacturing alternatives to make the desired product, which, in the case of micropower generation devices, calls upon MEMS fabrication technology. Manufacturing of the devices is outside of the scope of this thesis but the alternatives considered are limited based on manufacturing considerations. Material and structural considerations [259, 265] are also out of the scope of this thesis, and are only

included implicitly, e.g., in the bounds for operating temperatures.

Under the assumption that rapid start-up operation is possible, the average performance mainly depends on the steady-state performance of the processes; nevertheless, the transient behavior is extremely important and needs to be addressed in the future. It is likely that certain processes exhibiting poor transient behavior must be excluded. Similarly, most electronic devices have a power demand varying over time. The case studies presented in this thesis consider only the steady-state case with a constant power demand. Moreover, although the models are tailored to microfabricated fuel cell systems the aforementioned methodology can be applied to generic products that involve physico-chemical processes.

2.3 Literature Review

While systematic process synthesis and design is a mature field at the macroscale, e.g., [92, 47, 226], there is a wide scope for research at the microscale. Prior to this thesis, efforts for methodological microreactor process design, only amounted to simple principles, such as Just In Time (JIT), Zero Hold-up, inherent safety, modularity, and Keep It Simple Stupid (KISS) [242, 245, 148]. The scalability of micropower processes is an issue for the optimal design, since scale-up based on replication is not necessarily optimal [43, 155].

There is a significant number of publications on detailed modeling of specific microchemical components. The research groups of Professors Klavs F. Jensen and Martin A. Schmidt at MIT have developed models for a variety of microchemical systems, e.g., [233, 155, 187, 18, 27, 120]. The research group of Professor Dionisios G. Vlachos at Delaware focuses on detailed modeling of mostly combustion-based reactors, e.g., [214, 215, 88, 213], flow patterns, e.g., [90, 89] and development of kinetic models, e.g., [87, 194]. The research group of Professor Mayuresh V. Kothare at Lehigh University considers mostly control issues in microchemical systems [21, 20, 50, 49]. The research group of Professor Steinar Hauan at CMU is studying design issues in mostly electrophoretic separation systems, e.g., [227]. In [126] there is an attempt

to approximate the concentration profile in a class of parallel wall microreactors, while in [186] control and understanding by online monitoring is proposed. Regarding the layout of microdevices in mesoscale plants, there are several contributions [242, 245, 140, 185, 173, 146, 147, 148]. Few authors implement mathematical programming, as in [276, 275, 72, 73], where reactor optimization is performed or in [268], where the optimal temperature trajectory for a PFR with a given reaction is found.

The area of man-portable power generation is extremely active and in [196] we provide a collection of well over hundred contributions, mainly in journal articles. Holladay et al. recently performed a literature review on hydrogen production [151]; another review article is by Maynard and Meyers [191]. There are several academic programs exploring microfabricated fuel cell systems, including MIT [1, 176, 31, 145], UIUC [2, 240, 237, 285], IMTEK in Germany [142], Batelle [219, 150], Bell laboratories [193], Lawrence Livermore Laboratories [205], ETHZ Zürich [236] and Caltech [252]. Also several companies such as Motorola, Toshiba, Casio, Fujitsu, NEC and Sanyo have research projects with the aim of developing miniature fuel cells [3, 189, 264, 248, 168], focusing mostly on the direct methanol fuel cell (DMFC). The vast majority of the publications deals with fabrication issues. There are a few contributions on basic scaling considerations [57, 97, 30, 60], and some contributions on detailed modeling [141, 59].

2.4 Methodology Overview

While there are recent advances in multi-scale methods that aim to couple automatically modeling at different scales, e.g., [52], with current computational possibilities and available algorithms it is impossible to solve for the optimal design and operation in one step because the devices considered involve complex geometries, multiple scales, time-dependence and parametric uncertainty. Therefore, our methodology is based on decomposition into three levels of modeling detail, namely system-level models for process synthesis, intermediate fidelity models for optimization of sizes and op-

eration, and detailed computational fluid dynamics models for geometric design (see also Figure 2-2).

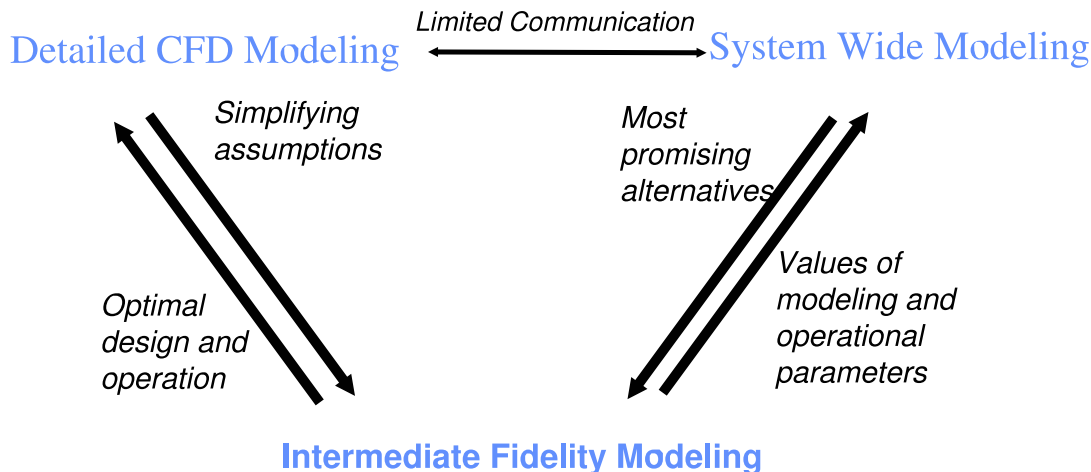


Figure 2-2: Overall methodology.

Process synthesis and layout considerations are performed with the use of algebraic models that are general enough to be independent of technological details, such as the catalysts used or the reactor configuration. Since the models are general and relatively simple, devices and reaction pathways at an early stage of development can be modeled. Through the use of simulation and parametric mixed-integer optimization the most promising process structures along with idealized layouts are selected among thousands of alternatives [201, 200]. We consider a variety of fuels including hydrogen, ammonia, various hydrocarbons and alcohols, and fuel cells including solid oxide fuel cell (SOFC), polymer electrolyte membrane (PEM), single chamber solid oxide fuel cell, direct methanol fuel cell (DMFC) and proton conducting fuel cell based on ceramic technology (PCFC). The optimal process structure depends on technological advances and product specifications. The system-level analysis provides limits of performance and can be used to determine at an early stage if a proposed device is worth pursuing; as an example the use of methane, which has been proposed in the literature, is shown to be marginally competitive with existing battery technologies, because of the storage requirements.

At the intermediate fidelity level we use distributed models, which allow optimization of unit sizes and operation (steady-state and transient) for a given process structure without the need to specify a detailed geometry. The resulting models involve partial differential-algebraic equations and the mathematical programming formulations employed include global and local dynamic optimization as well as stochastic programming [63, 64, 38, 62, 284]. The models used are rigorous and based on validated kinetic models. This level of modeling detail is particularly useful for technologies with demonstrated proof-of-principle.

Finally, the use of detailed two- and three-dimensional computational fluid dynamics allows geometrical improvements as well as the derivation and validation of modeling assumptions that are employed in the system-level and intermediate fidelity models. The development of these models requires specification of the geometry and therefore benefits from collaboration with fabrication efforts. Since the convergence of such models is time consuming and not robust, it is only possible to consider small variations in the geometry and this is done based on simulations as opposed to embedded in mathematical programming formulations. One of the major findings from CFD models is that for a class of devices the temperature in the active regions (reactor, etc.) is essentially spatially uniform; this is also supported by scaling analysis and preliminary experimental results.

Our methodology is formulated with the goal of harvesting and adapting the knowledge basis from macroscale process synthesis, design and operation. A one-to-one transfer is not possible because of different objectives, relevant physical phenomena, and limitations in fabrication. For instance process-synthesis at the macroscale is usually performed in stages, e.g., [47, 92], by first specifying the input-output structure of the process, then the recycle structure, then the separation system, and finally the heat recovery network. The physical layout is performed in the late stages of process design and is primarily driven by safety considerations. This hierarchical decomposition is possible because different units can operate essentially independently from each other, a fact that has led to the unit-operations paradigm. At the microscale a different design paradigm is necessary, that of closely interconnected components

of an integrated process. It is therefore necessary to consider heat integration and layout in the early stages of the process design simultaneously with the input-output structure.

The decomposition into three levels of modeling detail is done with respect to the different considerations at each scale and the coupling between the three levels is made by engineering judgment. The chosen decomposition allows interactions with experimental efforts from collaborations or literature, see Figure 2-3 for examples. At the system-level the set of alternatives considered is based on fabrication limits, and system-level considerations can be used to determine on which processes the fabrication effort should focus, as described in Section 2.6.5. Catalysis and reaction engineering efforts can provide lumped reactions to be used in the system-level models, while sensitivity considerations at the system-level can suggest the reactions on which catalysis effort should focus. For detailed modeling an initial geometry can be provided by reactor engineering efforts, and computational fluid dynamic analysis can suggest improvements on this geometry. At the intermediate fidelity level, kinetic models and limits of operating conditions are required and provided by reaction engineering and material characterization efforts. On the other hand, intermediate fidelity models provide optimal sizing of components and operating conditions.

2.5 Product Specifications

There are several metrics for the assessment of portable power generation devices, and depending on the application they can be formulated as design objectives or constraints. For most applications, design objectives include minimizing weight and/or volume of the power generation device plus fuel. An interesting differentiation is between rigid and flexible volume; for certain applications a flexible shape is desirable; for other applications a collapsible fuel container could be useful, so that the volume is reduced with time. The life time (measured in hours) of operation before the device needs replacement, recharging or refueling can be either a design objective or constraint. Economical and environmental cost are important criteria which are

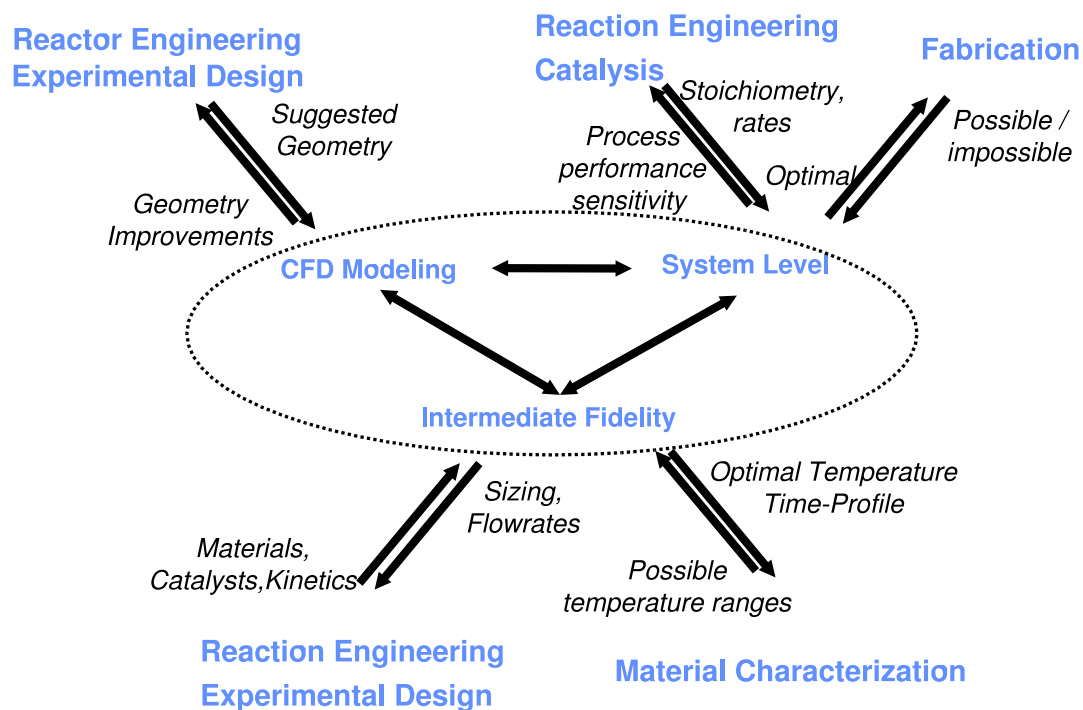


Figure 2-3: Examples of interactions with experimental efforts.

dominated by the materials used in devices and the fabrication and recycle/disposal of processes, rather than the fuel utilized, and therefore are not the topic of this thesis. Design constraints include reliability, safety and flexibility to ambient conditions. Reliability should in general be at least as high as that of the devices one wants to power. From a consumer's point of view, the power generation device must have a relatively simple way of recharging, refueling or replacing. Power generation is associated with heat generation, inversely proportional to the overall system efficiency, e.g., [193]; inefficient processes might be considered uncomfortable for portable applications because of the high heat generation, e.g., a cellular phone getting hot, or yield an undesired heat signature in the battle field. For rechargeable and refuelable devices an important metric is the maximal number of operating cycles, as well as the performance degradation with increasing number of cycles.

There are a large number of devices, with different characteristics, that currently require man-portable power production. Cellular phones currently use Li-ion rechargeable batteries with a mass of about 100g, a cost of around \$20, standby op-

eration of many hours and runtime of at most a few hours. Digital camcorders have a power demand of a few Watts and typically use rechargeable batteries with a cost of \$20-\$50. Laptop computer batteries are rechargeable, typically Li-Ion, have a mass of at most a few kg and typically have a capacity of under 100Wh/kg, resulting in a few hours of power supply at around 5 – 10W; they cost around \$100-\$200 and have a lifetime of approximately 300 charge/discharge cycles. Flashlights and toys operate with different types of batteries, either rechargeable or primary, with a power supply of 1-10W; the battery weight and operating time vary significantly with an operating cost on the order of \$5/h. On the limit of portability are electrical vehicles for the elderly and handicapped, which typically use lead-acid rechargeable batteries with a mass of several kg and a mission duration of many hours. It is to be expected that in the near future new power consuming devices, with possibly drastically different specifications on the power demand, will come to the market. An example is so called exoskeletons (also dubbed power pants, power elbows, etc.): robotic suits with the promise of multiplying the force of soldiers or rescue workers or even allowing motion to disabled people [156, 157, 112]. These devices will probably be characterized by a very low power demand during stand-by operation, for monitoring purposes, and a spike in the power demand, reaching tens or hundreds of Watts during actual operation. Other power consuming devices that are likely to become interesting applications for man-portable power generation include portable medical devices and robots.

Not only the power consuming device, but also the customer, influence the specifications on the power generation device, and since potential customers range from children to a dismounted soldier there is a great variety of needs. Power generation devices for children need to be inherently at least as safe as current batteries and non-toxic, even when operated differently than specified; the price is very important, while the performance and reliability are not crucial. Businessmen, who want to travel with their electronic devices, need power generation devices that can be carried and operated on airplanes, and refueling in different countries must be possible; performance and reliability are more important than price for this potential customer.

Mountaineers need power generation devices that can operate under extreme conditions for long mission durations; reliability is extremely important and performance largely outweighs price considerations. The dismounted soldier can be trained for safety and is already exposed to dangerous materials, and therefore safety requirements are less important than in civilian applications; performance and reliability are the main criteria and cost considerations are almost negligible. Design constraints for the dismounted soldier may include operation without noise generation or a thermal signature, while operation under extreme conditions is possible.

There is the perception that high-temperature devices are not acceptable for a consumer-product, because of the alleged heat dissipation and the risk associated with catastrophic failures. While high-temperature devices have a challenging thermal management [27] and start-up considerations are very important [65], the real consideration from the consumer's point of view is the overall heat dissipation, which is associated with the thermal efficiency and not the operating temperature. Similarly the real concern is not the possibility of a catastrophic failure of the device, but rather the energy content of the stored fuels in case of failure of the storage cartridge, which is essentially independent of the operating temperature. Depending on the fuel used, a catastrophic failure of the cartridge may lead to a release of toxic components or an explosion.

2.6 System-Level Analysis

Acknowledgments. *The modeling effort was built upon the work by Dr. Ignasi Palou-Rivera in the period from summer 2001 to spring 2002, who considered methanol cracking, ammonia cracking and propane partial oxidation in PEM and SOFC, with complete conversions and without the consideration of heat losses.*

Furthermore, Michael M. Hencke significantly participated in the modeling effort. Under the author's guidance he implemented layout options, PCFC, DMFC, single chamber fuel cell as well as some heat integration options.

Finally, Professor Klavs F. Jensen, Dr. Aleksander Franz and the MIT MURI team

in general were instrumental in the development of the alternatives considered.

Most power consuming devices are not operated constantly and have rapidly changing power demands, and therefore the dynamics and operation of power generation devices are very important. Similar to the electric vehicle application [278], a fast start-up procedure, at most on the order of minutes, is required. Assuming that the devices will be able to respond to power demands rapidly, the average performance will most likely be dominated by the steady-state behavior of the devices. The comparison of alternatives at the system-level is therefore performed considering steady-state processes; the calculation of system energy densities is described in Appendix A.

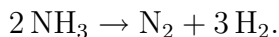
The choice between alternatives at the system-level is based on the notion of a superstructure from macroscale process design. Superstructure is a construct that contains all the alternatives to be considered in the selection of an optimal process structure [47]. An actual process design is a subset of the units and connections in the superstructure. While in the macroscale there are few limitations for process synthesis, in the microscale only relatively simple processes are possible [242, 245]. The set of alternatives considered here was formulated with the constraint that the realization of the processes is either currently possible, has been proposed in the literature or is foreseeable in the short term future (next years). As a consequence of the inherent requirement for process simplicity and the limitations in fabrication we chose to manually synthesize the set of alternatives considered, as opposed to using an automatic method such as in [177]. Unlike macroscale process synthesis, the complexity of man-portable power generation arises from the large choices of fuels, fuel reforming reactions and fuel cells and the early stage of component development, rather than from an elaborate combination of mixing, reaction and separation steps. In the past alternative and/or complementary approaches to the process superstructure have been proposed for macroscale process synthesis based on attainable regions [152, 122, 107], phenomena-based process synthesis [220] and the state-space approach [32]. Application of these ideas is outside the scope of this thesis and the superstructure approach is used here as the most natural choice.

2.6.1 Alternatives Considered

In order to describe the alternatives, we use a conceptual flowsheet superstructure (Figure 2-4), with the symbols explained in Figure A-1 of the appendix. We want to emphasize that the superstructure is only conceptual, and several of the “unit operations” can actually be physically combined. Our models account for thermal integration of the processes, as described in detail in Section 2.6.2. While kinetic data are available for special catalysts, e.g., [87, 159], that allow for detailed modeling, our intention is to have a general modeling framework that can cover various geometries and reactor types (PFR, CSTR, packed bed, etc.) and be independent of the specific catalysts used. Our models are therefore based on user specified efficiency parameters in the various units such as conversion, electrochemical efficiency, separation efficiency, etc. Once these parameters, as well as the operating conditions have been specified, the performance of the system is calculated. The physical properties and the equations used to model the “unit operations” are detailed in the Appendix A.

A basic requirement for a fuel is that it is compressible at relatively low pressures, so that it occupies a small volume. Ideally the fuels used should be nontoxic and inherently safe, but this requirement can be relaxed by allowing fuels that pose health and safety concerns similar to chemicals used in common consumer applications. From the vast choice of fuels and chemicals hydrides, ammonia, methanol, ethanol, heptane, and propane/butane mixtures are considered in this study. For comparison purposes hydrogen and methane stored as compressed gases are also included.

Ammonia can be thermally cracked at high temperatures producing hydrogen and nitrogen



Ammonia is extremely toxic and corrosive and therefore could only be used in special applications, such as remote sensors, but it has the advantage that it does not contain any carbon and its thermal cracking does not produce carbon monoxide, which has deleterious effects on proton exchange membrane (PEM) fuel cell performance. Therefore, it is often regarded as a fuel alternative, e.g., [192], and cracking has been demonstrated successfully at the microscale [26, 118].

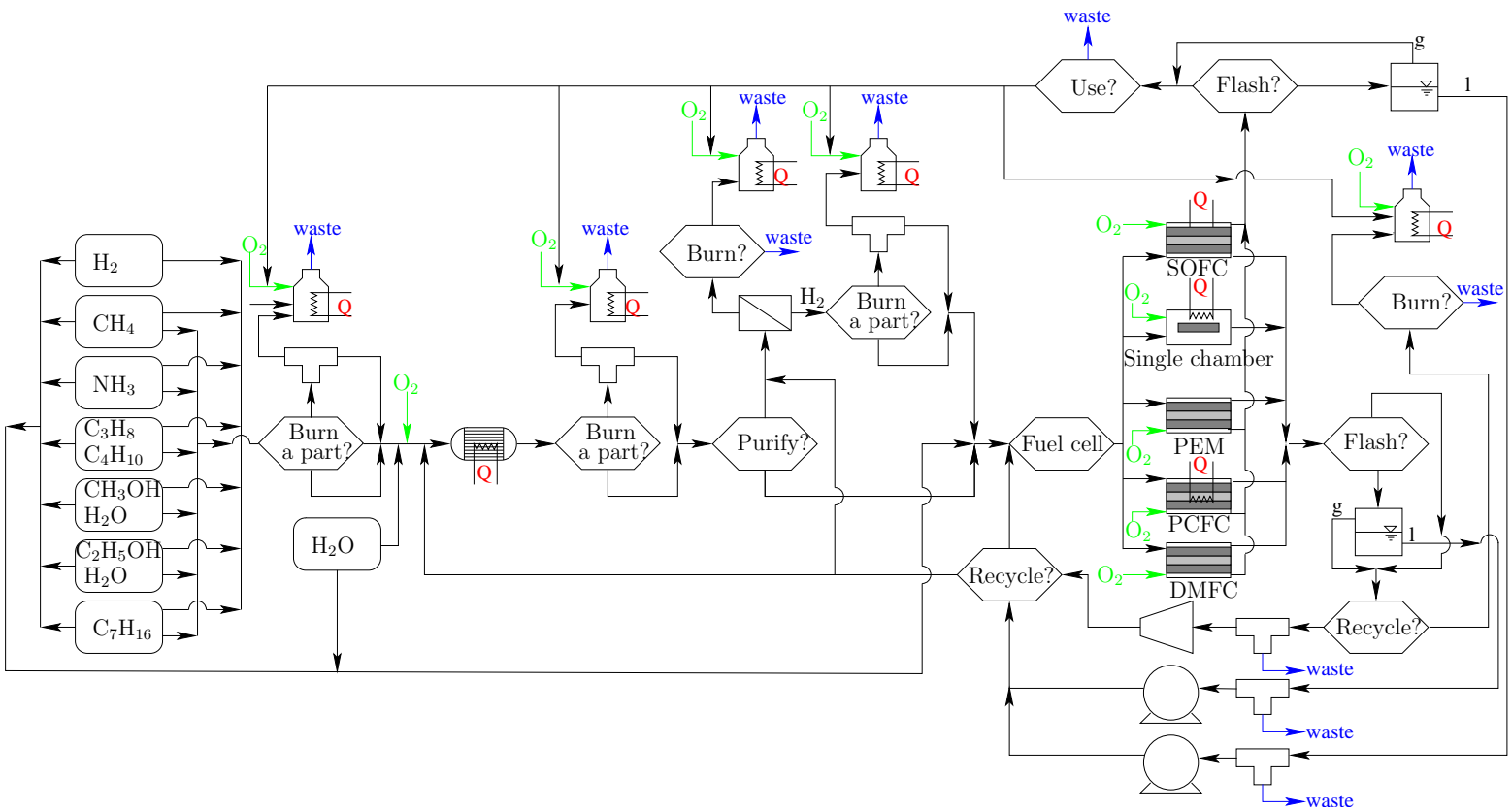
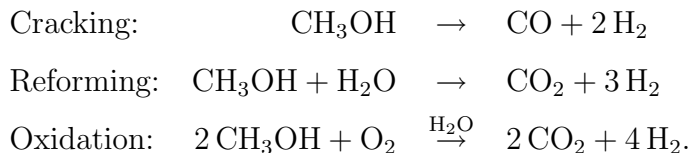


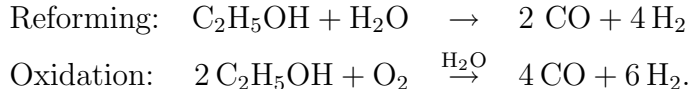
Figure 2-4: Set of alternatives considered.

Methanol is liquid at ambient conditions, but has a high vapor pressure, so that it can be easily vaporized. Methanol is flammable and over-exposure can have detrimental effects on health, but the risk associated with it is comparable to chemicals in common use, such as isopropyl-alcohol which is used as rubbing alcohol. There are three idealized ways of processing methanol for syngas generation:

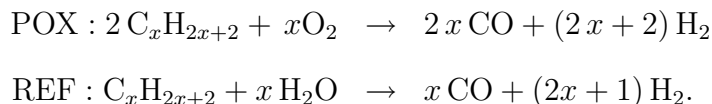


Methanol can be either stored as a pure component or in solution with water. Methanol can also be used directly in the direct methanol fuel cell (DMFC).

Ethanol has been proposed as a fuel for hydrogen production [83]. Ethanol has a more complicated chemistry than methanol, but has the advantage that it is less toxic. We consider two idealized reactions



Several representative hydrocarbons are included. Methane is included because it has been proposed in the literature; at ambient temperature it is supercritical and thus cannot be compressed adequately. Propane/butane are gaseous at ambient conditions, but can be liquefied at moderate pressures (< 10 bar) and are used as representatives of the hydrocarbon class. Propane/butane are flammable and present a health hazard by displacing the oxygen in the atmosphere, but are commonly used in applications such as lighters and as a heat source for outdoor activities. Heptane is liquid and was included as a representative of heavier hydrocarbons, in particular as a model fuel for JP-8. Hydrocarbons can either be steam reformed (REF) or partially oxidized (POX):



Recently advances towards partial oxidation in microstructured reactors have been made [174]. Also for some hydrocarbons reaction kinetics have been proposed [67]

Compressed hydrogen as well as a relatively broad class of hydrogen generators, described in detail in Appendix A, are included. Hydrogen does not need to be processed and can be readily oxidized in all fuel cell types considered here.

Flow pressure losses are not considered and all processes are assumed to operate at atmospheric pressure. As a consequence liquid and gaseous water and methanol have to be considered, but all other components are gaseous. Butane/propane and ammonia would most likely be stored as compressed liquids under a moderate pressure ($\leq 10\text{bar}$), in order to minimize the storage volume and to provide the necessary pressure gradient for the flow. In a detailed model the vaporization unit needs to be included, but because the heat of vaporization could be provided by heat transfer from the environment, the overall energy balance is not affected significantly by neglecting this unit.

Regarding the oxygen supply there are four possibilities, which for simplicity are not included in Figure 2-4. One possibility is to use atmospheric air, in which case a pressure rise has to be achieved by some mechanism, such as a microblower, which will be associated with an electric power loss. Another possibility is to use compressed air, which has the advantage that sufficient pressure will be available, but also means that the nitrogen and oxygen mass must be accounted for in the energy density calculations. Compressed oxygen could also be used, but in addition to the fact that the oxygen mass must be accounted for in the energy density calculations, there are some safety considerations associated with the use of compressed oxygen. The advantage of pure oxygen is the reduction of flowrates as well as the fact that no heating of the nitrogen is necessary. Finally, we also consider oxygen generators, which can offer a significant increase in volumetric energy density as compared to compressed gases. The modeling of these options is described in Appendix A.

For most applications, water needs to be either recycled or provided by a cartridge. For underwater applications an interesting alternative is to extract water from the ambient. In that case a water extraction device is needed and some energetic penalty

will be incurred. The details of these calculations are found in Appendix A.

The first design choice is to choose the fuel that will be used for power production and whether to perform fuel processing in a reactor, or to directly feed the fuel to a fuel cell. Based on the process design heuristic for simplicity [245] the postulated superstructure contains only one reactor. The next design choice is whether this fuel or another fuel will be fed into a burner for heat generation. The heat produced from burners serves to compensate for stream preheating, heat losses, endothermic reactions or even heating of the system at startup.

Depending on the fuel processing reaction a secondary feed of water or oxygen to the reactor is necessary. If desired, part of the reactor products can be split and burned to supply heat, in which case a stream split is necessary. A recycle of the reactor effluents is not included because recycling could only achieve backmixing. Any potential benefit of backmixing is unlikely to compensate for the power loss required to recycle. Recycling also seems unnecessary, since high conversions and many reactor flowpatterns have been demonstrated experimentally in the microscale for most of the considered reactions.

Certain components, such as carbon monoxide have deleterious effects on some fuel cells, e.g., PEM, and it may therefore be necessary to perform a gas purification. We assume that the purification will lead to two streams, one of essentially pure hydrogen along with a waste stream. We consider a partial loss of the hydrogen in the waste stream, but we neglect any energetic penalty for the purification and the effect of a sweep stream, which may be necessary for the operation [114]. The purification could either be sequential to the reactor, or the reactor and the membrane could be combined into one unit, allowing for higher selectivity of the reactions towards hydrogen [254, 114]. The separation waste can be either discarded or burned. If desired, the purification product (H_2) can be split, and a part can be fed into a burner.

We consider a variety of fuel cells, namely either a Solid Oxide Fuel Cell (SOFC) with the option of internal reforming, a Proton Ceramic Fuel Cell (PCFC), a hydrogen operated Polymer Electrolyte Membrane Fuel Cell (PEM), a Single Chamber fuel cell

operating with hydrogen and carbon monoxide, or finally a Direct Methanol Fuel Cell (DMFC). A SOFC has the benefit of fuel flexibility, but it is operated at high temperature which leads to large heat losses and problematic start-up. PEMs are run at low temperatures but cannot tolerate impurities, and water management is an issue. Single Chamber fuel cells are potentially easier to fabricate [96], but have the drawback that they are operated with premixed gases which potentially can lead to explosions and require catalysts with high selectivity. A PCFC is a relatively new concept [75], which has the potential of fuel flexibility while operating at slightly lower temperatures than SOFCs. A DMFC is a PEM based fuel cell in which a dilute methanol solution in water is reformed at a relatively low temperature, around 350K; major technical challenges include methanol crossover and water management. The reader is referred to the literature for extensive discussions about the technology differences in the fuel cells, e.g., [9, 258, 136]; details concerning the fuel cell modeling are found in Appendix A.

The conversion in the fuel cells (also denoted “fuel utilization”) is not complete, and the unreacted part of the fuel can either be burned or recycled. The basic recycling option is to split the fuel cell effluent into a recycle and a purge stream. The recycle stream can be mixed with the reactor inlet, the membrane inlet or the fuel cell inlet. A more promising recycling option would be recycling after separation, e.g., separate the hydrogen of the fuel cell effluent and recycle it to the fuel cell, or separate the steam/water and use it for reforming reactions and to prevent coking. These options are very appealing from the point of view of minimizing the mass, but separation might be very difficult to implement in the general case. We allow for the option of separating the liquid and gaseous components of the anode and cathode effluents in a flash at a given temperature, most likely near-ambient, and recycling a fraction of the liquids (mainly water and methanol) to the reactor or the fuel cell anode. Depending on the implementation of the recycle stream, a pressure increase mechanism may be necessary, e.g., a microfabricated pump, and we consider an energetic penalty in terms of a compression power. The feasibility of recycling is controversial, because of the lack of efficient units for pressure increase; nevertheless

the option of recycling has been included in this study for the sake of generality. The remaining liquid components constitute a purge stream. The gaseous components can be recycled to the reactor, membrane, or fuel cell as in [201].

The cathode effluent stream of the fuel cell can be reused to provide oxygen for a burner because it is plausible that the fuel cells will be operated at a relatively large oxygen excess. Reusing excess oxygen is most advantageous in volume-critical applications where the oxygen cartridge may occupy a large fraction of the total system volume. In addition, the temperature of the cathode effluent stream is higher than the ambient, so this reduces the energetic requirement of preheating the oxygen feed to the burner. However, in circumstances where the fuel cell discard temperature is substantially lower than the operating temperature of the burner (i.e., for a PEM or DMFC), preheating is still necessary. The cathode effluent also contains nitrogen, and in some cases, e.g., a PEM, also steam, and heating of these components to the burner operating temperature may outweigh the advantage of using preheated oxygen.

2.6.2 Integrated Layout and Thermal Management

The graphical representation of the superstructure (Figure 2-4) does not contain information about the physical layout. A very promising approach for thermal management is to couple two or more units thermally in a near-isothermal stack [25]. In this manner direct heat transfer between heat sinks and heat sources is possible, as well as heat recovery of the effluent streams; thermally coupling two units also reduces the surface area and as a consequence the heat losses. Combining units is thus a layout consideration that influences the process performance. As a consequence, the problems of flowsheet design, physical layout and heat integration need to be solved simultaneously.

Heat losses are considered with a lumped model. Based on the calculated volumetric flow \dot{V} and a specified residence time τ the necessary volume is calculated as $V = \dot{V} \tau$, as well as an equivalent surface $A = 6V^{2/3}$ (assuming a fixed aspect ratio of the devices). The heat losses are then calculated using an overall heat transfer

coefficient U_{loss} (dependent on the insulation) and an overall emissivity (including the view factor and accounting for the presence of radiation shields) ϵ , as

$$Q_{loss} = A \left(U_{loss} (T_{op} - T_{amb}) + \epsilon \sigma_{SB} (T_{op}^4 - T_{amb}^4) \right).$$

Heat recovery is difficult to be realized at this scale, but there are efforts towards this end [28, 212], by allowing for a heat exchange between the inlet and outlet gases. In our models this approach is reflected by the use of a discard temperature T_{out} from the main units (reactor, burners and fuel cell), which can be lower than the operating temperature T_{op} .

In the following description, we will refer to individual process components (i.e., fuel cell, reactor, burner, etc.) as “units”, although they are not independent in the sense of the unit operations design paradigm. Specification of the process layout requires an indication of the relative location and connectivity of every unit present in the selected flowsheet of the superstructure, e.g., is the fuel cell in thermal contact with the reactor? We propose an idealization of the layout considerations, allowing only for two extremes, implemented using logical decisions for the connectivity of each pair of units; this simplification is done for the same reasons that we used generic models for the reactor and the fuel cells. One extreme is that the units are thermally connected, so that they share the surface that results in heat losses, and one energy balance is sufficient. Within these “stacks”, flow streams proceed directly between units, so that heat losses between units are neglected. Due to the small length scales associated with these microdevices, convection and conduction are rapid in these stacks, thus necessitating that all units within a given stack operate with small temperature differences between them. Although our models allow for unit operating temperatures to be defined separately, this temperature constraint must be considered when deciding to combine two units. It is unlikely to find a way to use the heat excess of units operating at low temperatures, e.g., PEM or DMFC, and we therefore do not allow the possibility of stacking these units with any other units. The combination of units may also be limited by fabrication possibilities that are not

the subject of this thesis.

The other extreme that we consider is that the units are separated, so that each one has separate heat losses to the ambient and significant heat losses occur when mass or heat are transported between the units. In the case of a stream flowing from unit i to unit j , the temperature inlet to unit j is calculated as

$$T_j = T_i - \chi_{temp} (T_i - T_{amb}),$$

where the energetic penalty (“temperature loss factor”) $\chi_{temp} \in [0, 1]$ is a given parameter. In addition, heat exchange is inefficient, if not impossible, if the units are not integrated into stacks. We do allow for the option of heat exchange from burners that are not thermally coupled to the heat sink, but we impose an energetic penalty. The heat input to unit j is taken as a fraction of the heat output from unit i

$$Q_{j,in} = (1 - \chi_{heat}) Q_{i,out},$$

and the difference between $Q_{i,out}$ and $Q_{j,in}$ is assumed to be lost to the ambient. We do not account for details of the heat exchange; possible realizations of heat exchange are radiation or conduction through a rod between the units, or preheating of inlet gases to one unit by the heat excess of the other.

In Figure 2-5 the two extreme cases of layout are illustrated for a process, in which butane is partially oxidized and the syngas produced is fed into a SOFC, while the cathode effluents are used to oxidize the unburned syngas from the anode for heat generation. In one extreme case the SOFC, reactor and burner are assumed to be thermally coupled, while in the other extreme case, all three process components are separate, with remote heat exchange between the burner and the SOFC. It should be noted that Figure 2-5 is conceptual and not an actual design.

While layout options including many stacks are conceivable, we allow a maximum of two theoretical stacks, each of which can have up to five individual units. This restriction is motivated by the fact that the superstructure has up to two main units, namely the reactor and a fuel cell and by this restriction an explosion in the number

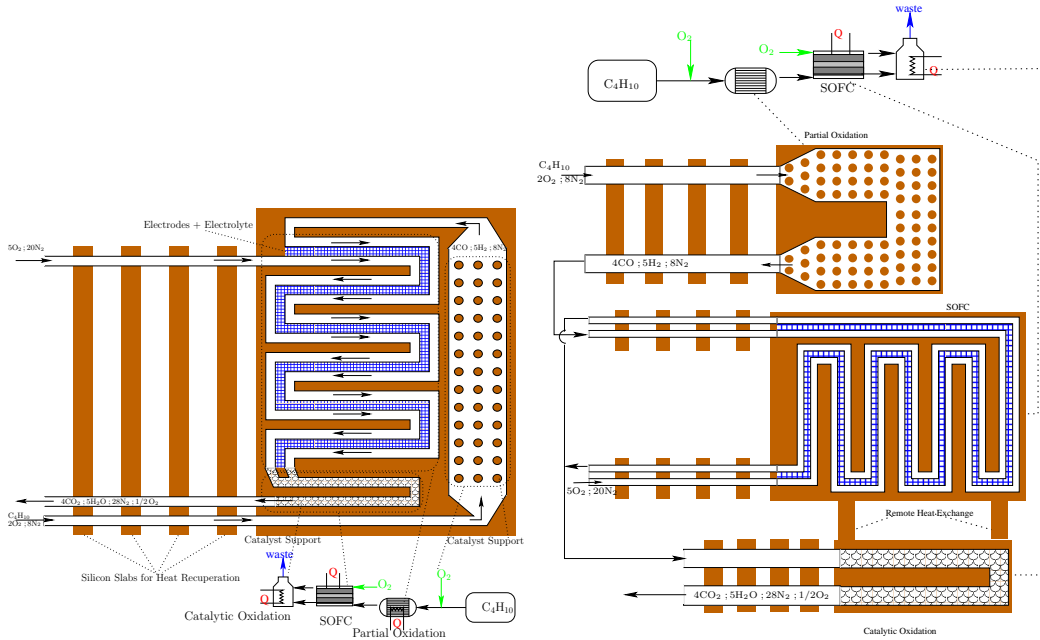


Figure 2-5: Conceptual difference between coupled (left) and non-coupled (right) process components.

of logical choices is avoided. The five units potentially available for each stack are the fuel cell effluent burner, the membrane waste burner, the fuel cell, and either a fuel burner, a reactor products burner, or a membrane products burner. We assume only one instance of the former four units and therefore they can be present in only one stack; we do allow for the possibility of splitting the fuel, reactor products, and membrane products streams, however, so that the latter units may be present in both stacks.

In the case that multiple burners are present in a stack, it is possible to combine them into a single unit, by mixing the inlet streams and producing a single waste stream. This may be easier from a fabrication stand point, or may result in a higher average conversion. According to the constraints on the number of units and stacks, there can be at most three burners in a single stack. In this case there are three possible ways to premix the burner streams: combine all three streams (one burner), combine two streams and leave the third stream separate (two burners), or leave all three streams separate (three burners). Our models allow for all three possibilities.

Each stack has to be autothermal (heat load on each stack negative or zero).

Since we assume perfect heat exchange within the stack and heat exchange from non-coupled units subject to an energetic penalty, $\chi_{heat} \in [0, 1]$, the energy balance around a single stack with components i , inlet streams j , and outlet streams k is given by

$$\sum_i \sum_k N_{i,k} H_{i,k}(T_k) = \sum_i \sum_j N_{i,j} H_{i,j}(T_j) + Q_p - Q_{loss} + PW_{FC}.$$

We require that the sum of the heat duty Q_p of the stack and the effective heat exchange $(1 - \chi_{heat}) Q_{distant}$ from distant units, is less than or equal to zero.

$$Q_p + (1 - \chi_{heat}) Q_{distant} \leq 0.$$

In general heat removal can be problematic [193, 191], but we assume that for the high-temperature processes considered it will be possible by reducing the insulation, introducing a heat sink element, or increasing the oxygen excess. We currently do not directly account for the, presumably small, decrease in the system performance by implementing these options.

For low-temperature fuel cells, such as hydrogen operated PEM and DMFC, cooling needs to be accounted for. Passive cooling is included in the heat losses to the ambient, by the heat loss coefficient U_{loss} , but this may not be sufficient. We therefore allow two possibilities for cooling, namely air-based cooling and water-based cooling. Air cooling corresponds to an increase of air excess, or a fan and cooling is achieved by heating up air from ambient temperature to a temperature close to the fuel cell temperature. The air requirement is added to the air needed for the reactions. Water-cooling models the effect of controlling the humidity in the fuel cells; in the case of heat excess the temperature increases and water is evaporated at a higher rate; control of the humidity requires the addition of water.

In a similar vein to the combination of units, we also consider the option of combining fuel cartridges. This may be more difficult to fabricate, but has the advantage of minimizing the cartridge volume. We allow for the option of combining all fuel cartridges, combining the cartridges that go to burners, or keeping all the cartridges

separate. In the case that compressed oxygen or an oxygen generator is used, we only consider the option of a single oxygen cartridge.

2.6.3 Chemical Equilibrium Considerations

In macroscale systems the notion of equilibrium reactors is often employed for reactions that are equilibrium-limited. In this section we examine whether or not the reactions considered are equilibrium limited using two approaches. One approach is the restricted equilibrium approach, where only one reaction is considered, based on the assumption that a catalyst exists which speeds up this one reaction to such an extent that components not participating in this reaction can be treated as inert. The other approach is to assume that all components equilibrate with each other. In both cases the chemical equilibrium is formulated as a convex-nonlinear minimization problem with linear constraints [257]

$$\min_{n_i} \sum_{i=1}^{16} n_i \left(G_i^\circ(T) + RT \ln \left(\frac{P}{P_{ref}} \right) + RT \ln \left(\frac{n_i}{\sum_{j=1}^{16} n_j} \right) \right),$$

where $G_i^\circ(T)$ is the Gibbs free energy of the pure component i at the reference pressure P_{ref} . Atom balances are enforced as constraints:

$$\sum_{i=1}^{16} n_i \beta_{i,X} = \sum_{i=1}^{16} n_{i0} \beta_{i,X} \quad X = N, O, H, C,$$

where $\beta_{i,X}$ is the number of atoms of element X in component i and n_{i0} is the initial number of moles of component i . In addition, all mole numbers must be nonnegative ($n_i \geq 0$). We consider a total of 16 components: NH_3 , N_2 , N_2O , NO , NO_2 , O_2 , H_2 , H_2O , CO , CO_2 , CH_3OH , CH_4 , C_2H_6 , C_2H_4 , C_3H_8 and C_4H_{10} . We used GAMS [56] and confirmed the results with three available solvers SNOPT, CONOPT and BARON.

Table 2.1 contains the summary of the equilibrium yields for the reactions considered. For the initial mole numbers stoichiometric mixtures with atmospheric air is used for all cases. In addition for the methanol oxidation the effect of water presence

is also studied, with a mixture of 2 moles methanol, 1 mole oxygen and 4 moles water. The yield is defined as the amount of hydrogen at equilibrium divided by the amount achieved with the idealized reaction.

Table 2.1: Summary of the results for the chemical equilibria

| Reaction | Restricted Equilibrium | | Unrestricted Equilibrium | |
|---|------------------------|--------|--------------------------|--------|
| | T in K | Yield | T in K | Yield |
| C ₃ H ₈ /C ₄ H ₁₀ partial oxidation | 1100 | > 0.99 | 1100 | 0.7 |
| | | | 1600 | > 0.99 |
| C ₃ H ₈ /C ₄ H ₁₀ reforming | 1100 | > 0.99 | 1100 | 0.6 |
| | | | 1500 | > 0.99 |
| CH ₃ OH cracking | 600 | > 0.99 | 600 | 0.01 |
| | | | 1600 | 0.98 |
| CH ₃ OH reforming | 500 | 0.93 | 1300 | 0.73 |
| CH ₃ OH oxidation | 500 | 0.95 | 1200 | 0.65 |
| CH ₃ OH oxidation with H ₂ O | 400 | > 0.99 | 1000 | 0.8 |
| NH ₃ cracking | 900 | > 0.99 | 900 | > 0.99 |

The study of chemical equilibria shows that the reactions considered are not significantly equilibrium limited, provided catalysts that suppress the formation of undesired components can be developed. As a consequence the inclusion of equilibrium based calculations in the reactor and fuel cell modeling would not significantly improve the quality of the models and was therefore not pursued. The chemical equilibria also show that at temperatures $\leq 1100K$ and at moderate air excess $\leq 20\%$ the formation of nitrogen oxides is insignificant from the perspective of mass and energy balances. As a consequence it is legitimate not to include those components in our calculations.

2.6.4 Simulation-Based Case Studies

The described superstructure is represented as a steady-state simulation model using our in-house software packages ABACUSS II [273, 272] and DAEPACK [269, 271]. The mass and energy balances are formulated within the process simulator ABACUSS II, while the physical property calculations are performed in Fortran external procedures, which are called by the simulator, using automatic differentiation as described in [273]. The implementation is available in the form of a web-interface [198],

which allows for facile use by remote users, who are unfamiliar with the modeling language and the details of the models. Upon request and subject to approval this web-interface can be made available for academic purposes.

The process simulation requires a fully determined system of equations and therefore all design choices have to be made prior to the simulation. There are also a number of parameters that need to be specified, including the operating conditions (operating T_{op} and discard temperatures T_{out} , residence times τ , air excess Φ , power output PW , ratio of reactants, recycle ratio α) as well as the modeling parameters (conversions ζ , overall heat loss coefficients U_{loss} and emissivities ϵ in the reactor, the fuel cell and the burners, membrane efficiency η_M , fuel cell efficiency η_{FC} , power load for the compressor K_P).

Many microdevices and components of the proposed processes are not yet fully developed. Therefore some parameters, such as the range of feasible operating temperatures in the different units, are not accurately known. Sensitivity analysis and parameter variation show the effect of changing those parameters on the best process structure and energy density. For the steady state the sensitivities correspond to partial derivatives of the model variables with respect to these parameters.

The number of processes considered is so large that an exhaustive comparison and study of the parameter dependence is not possible within the scope of this thesis. Instead, we choose to compare a small set of alternatives in terms of different metrics as well as to study the influence of key parameters (different for each alternative). These studies show the capabilities of the methodology developed and present insights into the product design of portable power generation devices. It should be noted that the numerical results depend on the values of the operating and modeling parameter values used. The strength of our methodology is that the formulated models retain their validity for the whole range of parameters and can be used as a tool for the evaluation and comparison of the different technologies.

Maximal Densities

Since in the considered processes the fuels are chemically converted to hydrogen or syngas the maximal achievable energy density can be estimated assuming atmospheric air and calculating the production of hydrogen and carbon monoxide per kg of the fuel or fuel/water. In Table 2.2 a power production of 64 Wh/(mol H₂) and 71 Wh/(mol CO) is assumed and the heat of reaction $\Delta_r H$ is reported for $T = 298$. The maximal densities differ from the theoretical values for direct fuel cells (Figure 2-1), since in the processes considered the chemical potential of the fuel processing does not contribute to the power production.

Table 2.2: Ideal energy densities

| Reaction | Power from H ₂ [Wh/kg] | Power from CO [Wh/kg] | Total Power [Wh/kg] | $\Delta_r H$ [Wh/kg] |
|--|--------------------------------------|--------------------------|------------------------|-------------------------|
| $C_3H_8 + 1.5 O_2 \rightarrow 3 CO + 4 H_2$ | 5630 | 4770 | 10400 | -1400 |
| $C_4H_{10} + 2O_2 \rightarrow 4 CO + 5 H_2$ | 5460 | 4930 | 10400 | -1500 |
| $C_3H_8 + 3 H_2O \rightarrow 3 CO + 7 H_2$ | 4480 | 2170 | 6640 | 1180 |
| $C_4H_{10} + 4H_2O \rightarrow 4 CO + 9 H_2$ | 4380 | 2200 | 6580 | 1390 |
| $CH_3OH \rightarrow CO + 2 H_2$ | 3960 | 2230 | 6190 | 790 |
| $CH_3OH + H_2O \rightarrow CO_2 + 3 H_2$ | 3800 | 0 | 3800 | 275 |
| $2 CH_3OH + O_2 \rightarrow 2 CO_2 + 4 H_2$ | 3960 | 0 | 3960 | -1670 |
| $2 NH_3 \rightarrow 3 H_2 + N_2$ | 5590 | 0 | 5590 | 745 |

Comparison of Processes

In this base case a small subset of the possible process configurations is considered with conservative estimates for the operating parameters. For the partial oxidation (HC-POX) and reformation (HC-REF) of hydrocarbons an equimolar mixture of propane/butane is used with no excess of air or water in the reactor. Pure methanol is used as a fuel for all methanol processes, while the reactor inlet is pure methanol for cracking (CH₃OH-CR), equimolar mixture of water and methanol for reformation (CH₃OH-REF) and oxidation (CH₃OH-OX), with no oxygen excess. Pure ammonia is used for the ammonia cracking (NH₃-CR). The feasibility of recycling is controversial and therefore is not included in the base case. For the PEM-based processes purification of the fuel cell inlet stream is performed, with the exception of the ammonia

cracking. No membrane is used for the SOFC-based processes. The values used for the operating conditions and the model parameters are summarized in Table 2.3. For each process different reactor (R) and burner (B) temperatures were used. Different heat integration options were studied and only the best energy density is reported. The reported heat losses Q_{loss} include the heat produced in the PEM, while the heat excess Q_{ex} includes only the heat available at high temperatures. The heat excess is nonzero for processes where the reaction exothermicity is greater than the heat losses and no burners are needed, or for processes where the burning of waste produces more heat than needed. For simplicity the cooling load is ignored and an overall heat balance is calculated, assuming arrangement in a single stack.

Table 2.3: Process parameters for the comparison of processes

| | |
|--------------------------------|------------------------------|
| Ambient temperature | $T_{amb} = 298\text{K}$ |
| Power output | $PW = 1\text{W}$ |
| Conversion in reactor | $\zeta = 0.9$ |
| SOFC temperature | $T_{op} = 950\text{K}$ |
| Residence time in reactor | $\tau = 1\text{ms}$ |
| Discard temperature from SOFC | $T_{out} = 750\text{K}$ |
| Conversion in burners | $\zeta = 0.95$ |
| PEM temperature | $T_{op} = 410\text{K}$ |
| Residence time in burners | $\tau = 1\text{ms}$ |
| Discard temperature from PEM | $T_{out} = 410\text{K}$ |
| Air excess in burners | $\Phi = 1.2$ |
| Conversion in fuel cell | $\zeta = 0.8$ |
| Overall heat loss coefficient | $U = 3\text{W/m}^2/\text{K}$ |
| Residence time in fuel cell | $\tau = 20\text{ms}$ |
| Emissivity (incl. view factor) | $\epsilon = 0.2$ |
| Efficiency of fuel cell | $\eta_{FC} = 0.7$ |
| Air excess in fuel cell | $\Phi = 1.2$ |
| Membrane efficiency | $\eta_M = 0.8$ |

The results for atmospheric air with an energetic penalty for the air flow of $K_P = 10\text{J/mol/K}$ are summarized in Table 2.4. While these results depend on the numerical values of the parameters chosen, and only relatively rough estimates are available, it becomes obvious that the processes considered can lead to much higher energy densities than state-of-the-art batteries. The most promising option is par-

tial oxidation of propane/butane in combination with an SOFC. This is due to the higher maximal achievable density of this process as well as to the fact that this is an exothermic reaction, and the heat excess of the reaction can partially compensate for the heat load. The heat losses are of the same order of magnitude as the power generation.

Table 2.4: Process comparison for atmospheric air

| Fuel | FC | $T_{R,op}$ | $T_{R,out}$ | $T_{B,op}$ | $T_{B,out}$ | [Wh/kg] | [Wh/l] | Q_{loss} | Q_{ex} |
|------------------------|------|------------|-------------|------------|-------------|---------|--------|------------|----------|
| HC-POX | PEM | 1100 | 800 | 1200 | 900 | 1890 | 1010 | 1.0 | 0.1 |
| HC-POX | SOFC | 1100 | 1100 | 1200 | 900 | 3070 | 1621 | 1.7 | 0 |
| HC-REF | PEM | 1100 | 800 | 1200 | 900 | 1490 | 1070 | 1.2 | 0.2 |
| HC-REF | SOFC | 1100 | 1100 | 1200 | 900 | 1740 | 1140 | 1.7 | 0 |
| CH ₃ OH-CR | PEM | 600 | 500 | 700 | 700 | 1330 | 1070 | 0.6 | 1.2 |
| CH ₃ OH-CR | SOFC | 600 | 600 | 1000 | 700 | 1820 | 1460 | 1.1 | 0 |
| CH ₃ OH-REF | PEM | 500 | 500 | 700 | 600 | 1350 | 1160 | 0.5 | 0.0 |
| CH ₃ OH-REF | SOFC | 500 | 500 | 1000 | 700 | 1160 | 980 | 1.2 | 0 |
| CH ₃ OH-OX | PEM | 400 | 400 | — | — | 890 | 770 | 0.5 | 0.5 |
| CH ₃ OH-OX | SOFC | 400 | 400 | 1000 | 700 | 930 | 800 | 1.5 | 0 |
| NH ₃ -CR | PEM | 900 | 700 | 1200 | 900 | 1620 | 990 | 1.0 | 0 |

The calculations are repeated for the case of compressed oxygen, available at 100 bar, instead of atmospheric air, and the resulting energy densities are much lower (Table 2.5), because the oxygen mass and volume must also be accounted for. In particular the volumetric energy density is an order of magnitude lower, because of the significant volume that the oxygen needs. It should be noted that the gravimetric energy density is overestimated because the mass of the empty cartridges would be significant for this case. As expected processes requiring great amounts of oxygen, such as partial oxidation of hydrocarbons are most influenced, and the differences between the processes are smaller than in the case of atmospheric air.

Heat Losses and Scaling

In order to study the influence of scale and heat losses, the power output as well as the overall heat transfer coefficient and the emissivity were varied for the case of partial oxidation of propane/butane with the SOFC assuming arrangement in a single stack..

Table 2.5: Process comparison for compressed oxygen

| Fuel | FC | $T_{R,op}$ | $T_{R,out}$ | $T_{B,op}$ | $T_{B,out}$ | [Wh/kg] | [Wh/l] | Q_{loss} | Q_{ex} |
|--------------------------|------|------------|-------------|------------|-------------|---------|--------|------------|----------|
| HC-POX | PEM | 1100 | 800 | — | — | 730 | 130 | 0.8 | 0.2 |
| HC-POX | SOFC | 1100 | 1100 | — | — | 990 | 150 | 0.5 | 0.35 |
| HC-REF | PEM | 1100 | 800 | 1200 | 900 | 640 | 120 | 0.9 | 0.4 |
| HC-REF | SOFC | 1100 | 1100 | 1200 | 900 | 840 | 145 | 0.7 | 0 |
| CH ₃ OH – CR | PEM | 600 | 500 | 700 | 700 | 590 | 115 | 0.6 | 1.2 |
| CH ₃ OH – CR | SOFC | 600 | 600 | 1000 | 700 | 910 | 170 | 0.5 | 0.0 |
| CH ₃ OH – REF | PEM | 500 | 500 | 700 | 600 | 745 | 170 | .6 | 0.0 |
| CH ₃ OH – REF | SOFC | 500 | 500 | 1000 | 700 | 740 | 150 | 0.6 | 0.0 |
| CH ₃ OH – OX | PEM | 400 | 400 | — | — | 565 | 144 | 0.5 | 0.4 |
| CH ₃ OH – OX | SOFC | 400 | 400 | — | — | 580 | 130 | 0.7 | 0.1 |
| NH ₃ – CR | PEM | 900 | 700 | 1200 | 900 | 865 | 170 | 0.8 | 0.0 |

Varying the residence time would have a similar effect with a different dependence function.

The results are shown in Figure 2-6. If no heat losses are taken into account the process is invariant to scale for the considered level of model detail. Small scale (<1W) processes are influenced by heat losses much more than larger scale, since for higher power production the available heat excess partially compensates for the heat losses. For relative low heat losses ($\epsilon = 0$, $U = 25\text{W/m}^2/\text{K}$ and $\epsilon = 0.3$, $U = 5\text{W/m}^2/\text{K}$) there is a power output (6W and 9W respectively), above which the heat excess overcomes the heat losses and the energy density for the case of zero heat losses is matched. Because of the high operating temperature the influence of radiation is more important than conduction/convection. For endothermic processes (not shown) the influence of heat losses is even more dramatic.

Basic Recycling

The effect of recycling and the dependence of the compressor parameter K_P are shown in Figure 2-7 for the case of partial oxidation of propane/butane with the PEM, ignoring the cooling load. Low values of the compressor parameter correspond to a low pressure ratio, or to high compression efficiency. For low values of the compressor parameter ($K_P < 300\text{J/mol/K}$) recycling of the unburned hydrogen slightly improves the process efficiency, while for high values of the compressor the energy

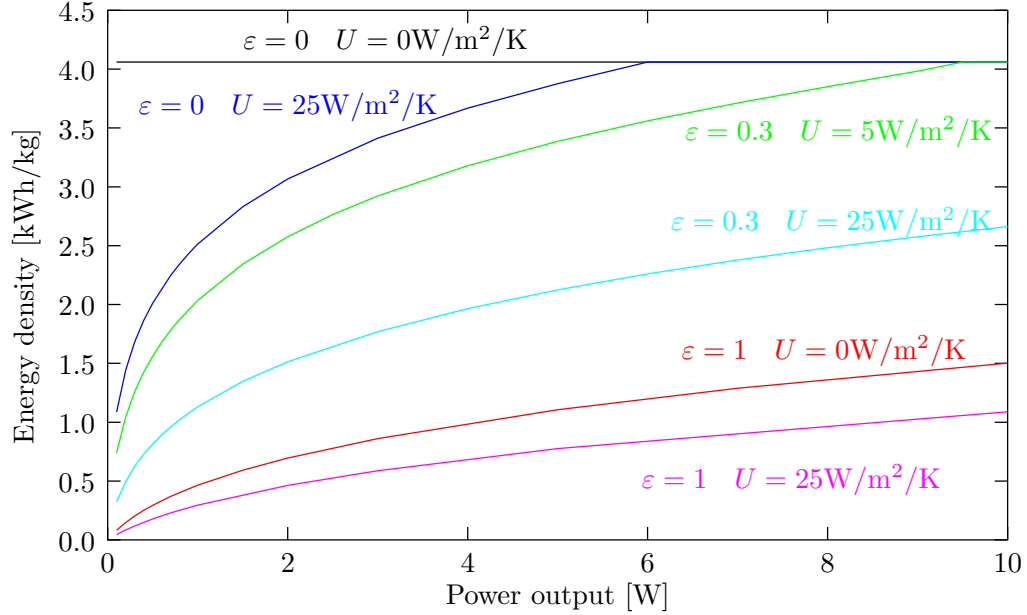


Figure 2-6: Influence of heat losses and scale on the energy density

needed for compression overcomes the benefit of recycling. Above a recycle ratio of approximately 0.35, burning the fuel cell effluents does not produce enough heat to compensate for the heat losses, and part of the fuel products needs to be burned, which leads to a jump in the energy density.

Conversion and Efficiency

The effect of the SOFC efficiency η_{SOFC} and the conversion ζ_{FC} are studied for the case of partial oxidation of propane/butane with the SOFC assuming arrangement in a single stack. In Figure 2-8 the energy density as well as the flowsheet corresponding to the best heat integration option are shown. For total conversion $\zeta_{SOFC} = 1$ there is a kink at $\eta_{SOFC} = 0.52$, because at this efficiency the heat excess in the fuel cell is not enough to compensate for the heat losses, and part of the fuel needs to be burned. For the intermediate conversions ($\zeta_{SOFC} = 0.6$ and $\zeta_{SOFC} = 0.8$), there are three kinks corresponding to the point where the heat excess of the fuel cell is not enough ($\eta = 0.33$ and $\eta = 0.43$), to the point where burning the fuel cell effluents leads to higher energy density than burning part of the fuel ($\eta = 0.40$ and $\eta = 0.45$) and the point where burning the fuel cell effluents does not provide enough heat

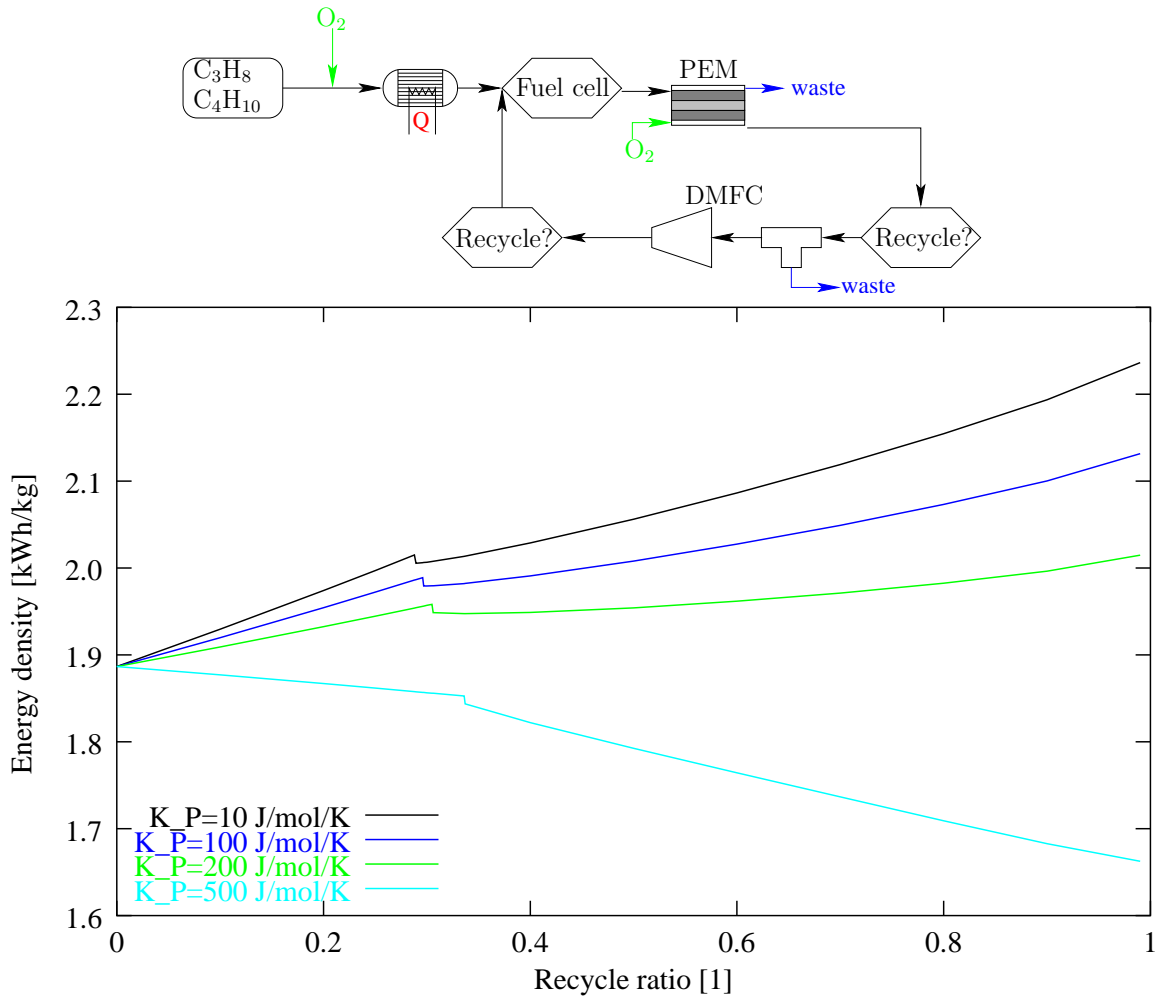


Figure 2-7: Effect of recycling on the energy density for different values of the compression parameter K_P

($\eta = 0.53$ and $\eta = 0.64$ respectively) and two burners are needed. For the low conversion $\zeta_{SOFC} = 0.3$ there is a kink at $\eta_{SOFC} = 0.48$ corresponding to the point where burning the fuel cell effluents leads to higher energy density than burning part of the fuel.

Influence of Product Specifications on Hydrides and Hydrocarbons

In Figure 2-9 we compare one SOFC-based and two PEM-based processes as a function of relevant technological parameters using two different metrics, namely volumetric and gravimetric energy density. For simplicity, we base the densities only on the amount of fuel needed, which dominates over the device size for long mission

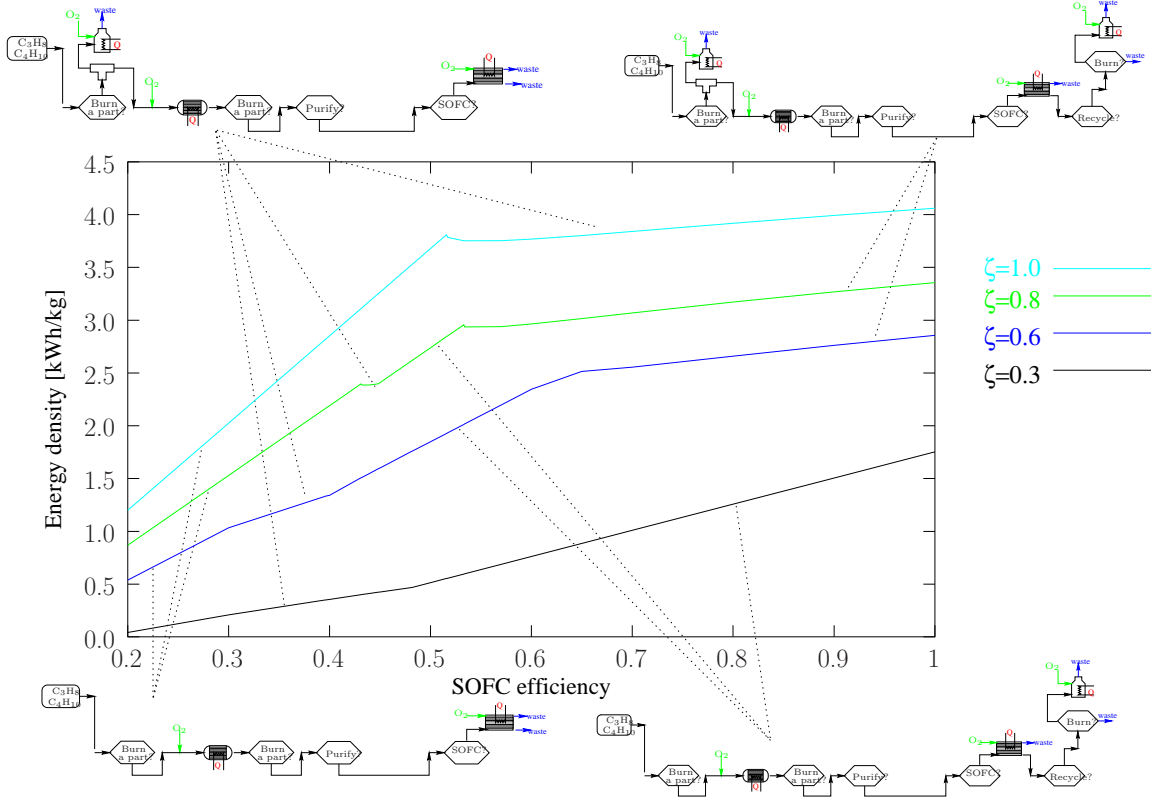


Figure 2-8: Effect of SOFC efficiency on the energy density for different conversion values

durations; these are referred to as gravimetric and volumetric *fuel* energy density, respectively. In the first process, denoted “Hydride”, hydrogen is released by a hydride and fed directly into a PEM. For the gravimetric energy density we use a hydride density of $0.2\text{g H}_2/\text{cm}^3$ and vary the weight percentage of hydrogen, while for the volumetric energy density we use a value of 8 weight % and vary the hydride density. In the second process, denoted “POX-PEM”, propane/butane are partially oxidized, and the hydrogen is separated with a varying recovery η_M and then fed into a PEM. Finally, in the SOFC based process, denoted “POX-SOFC”, propane/butane are partially oxidized and fed into a SOFC and the fuel cell effluents are used in a combustion chamber to provide sufficient heat for the high temperatures; the reactor, burner, and SOFC are assumed to be in thermal contact; carbon monoxide and hydrocarbons are assumed to be consumed in reforming reactions in the SOFC. For the PEM based processes there is always heat excess and the decision of oxidizing the fuel-rich streams

depends on the product requirements; they should be oxidized in an environment with strict requirements on toxic or dangerous emissions, and should not be oxidized when heat removal or heat signature are of concern.

Table 2.6 summarizes the parameter values used. The main observation is that the optimal design depends on the relative importance of objectives, i.e., maximal volumetric or gravimetric energy density, as well as advances in technology, i.e., membrane efficiency and hydride density. It is therefore plausible that different applications will require a different power generation device and that the optimal design can change with technological improvements. Regarding the actual comparison of hydrides and hydrocarbons, the general behavior is that highly concentrated hydrides can compete with hydrocarbon based processes in terms of volumetric energy density, whereas in terms of the gravimetric energy density they cannot. We want to again point out that the exact numbers depend on the parameter values used. Hydrogen recovery is very important, but even with complete recovery, the loss of the chemical potential of carbon monoxide makes PEM based processes less attractive than those utilizing a SOFC.

While the exact device size depends on fabrication and packaging technologies, it is to be expected that the processes which use hydrocarbon processing will result in a larger device than hydride based fuel cells due to the presence of a reactor. Therefore for small mission durations and/or power requirements the hydride based processes might be the optimal choice. On the other hand for large mission durations and power requirements, the fuel size dominates over the device size and one would expect that hydrocarbons will be the fuel of choice.

Effect of Oxygen Generators

In the basic comparison of processes we analyzed the effect of using compressed oxygen for cases where atmospheric air is not available, e.g., for underwater operations such as recreational diving. The volume of the compressed oxygen is very large and consequently the volumetric energy density is low. Here we compare the process of the previous design based on partial oxidation of propane/butane in combination with

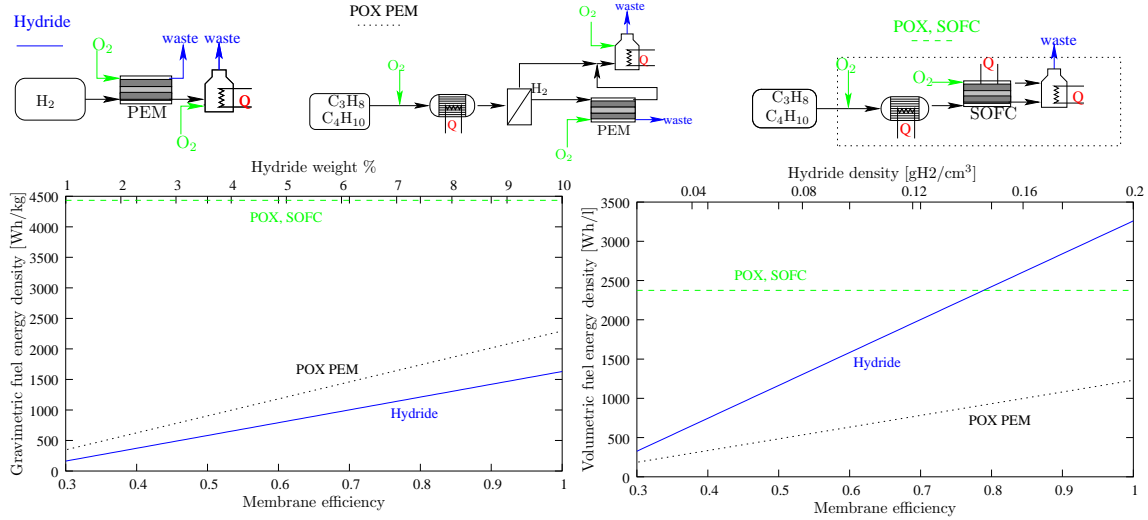


Figure 2-9: Comparison of hydride based and hydrocarbon based processes in terms of volumetric and gravimetric fuel energy density.

a SOFC for a mission duration $\tau_{mission} = 30$ h using either an oxygen generator or compressed oxygen as the source of oxygen. For the case of compressed oxygen we assume a plastic fuel storage material with a maximal allowable tensile stress of 100 MPa and a density of 1.5 kg/l and no minimum thickness requirement and we vary the storage pressure in the range 10-1000 bar. For the case of hydrogen peroxide we use again a cartridge density of 1.5 kg/l and vary the hydrogen peroxide weight fraction for two cartridge thicknesses, namely 1mm and 10mm. The properties for the storage materials were based on the range given in [223] for epoxic and polyesteric materials. As metrics for the comparison of the processes we use the volumetric and gravimetric *system* energy densities, where the system includes the power generation device and the stored fuel, as described in Appendix A. Either metric can be more important depending on the application; e.g., for a mountaineer minimizing weight is more important than minimizing volume, while for a diver minimizing the mass is not as crucial, because of buoyancy effects. The startup time was assumed to be $\tau_{startup} = 60$ s with an auxiliary battery with a gravimetric energy density of 200Wh/kg and a volumetric energy density of 200Wh/l; the volume factor for the device is assumed to be 10 and the device density 1kg/l; the influence of these parameters on the system volume and mass is insignificant, since the volume and mass are dominated

Table 2.6: Process parameters for the comparison in Figure 2-9.

| | |
|--|------------------------------|
| Ambient temperature | $T_{amb} = 298\text{K}$ |
| Power output | $PW = 1\text{W}$ |
| Reactor temperature | $T_{op} = 1000\text{K}$ |
| Reactor outlet temperature | $T_{out} = 500$ |
| Conversion in reactor | $\zeta = 0.9$ |
| SOFC temperature | $T_{op} = 1000\text{K}$ |
| Residence time in reactor | $\tau = 1\text{ms}$ |
| Discard temperature from SOFC | $T_{out} = 500\text{K}$ |
| Conversion in burners | $\zeta = 0.95$ |
| PEM temperature | $T_{op} = 350\text{K}$ |
| Residence time in burners | $\tau = 1\text{ms}$ |
| Discard temperature from PEM | $T_{out} = 350\text{K}$ |
| Air excess in burners | $\Phi = 1.2$ |
| Conversion in fuel cell | $\zeta = 0.8$ |
| Overall heat loss coefficient | $U = 3\text{W/m}^2/\text{K}$ |
| Residence time in fuel cell | $\tau = 20\text{ms}$ |
| Emissivity (incl. view factor) | $\epsilon = 0.2$ |
| Efficiency of fuel cell | $\eta_{FC} = 0.7$ |
| Air excess in fuel cell | $\Phi = 1.2$ |
| Compression parameter for the air feed | $K_C = 10\text{J/mol/K}$ |
| Burner temperature | $T_{op} = 1000\text{K}$ |
| Discard temperature from burner | $T_{out} = 500\text{K}$ |
| Water factor in fuel cell | $\Psi = 1$ |
| Propane molfraction in feed | 0.5 |
| No air excess in reactor | $\Phi = 1.0$ |

by the oxygen storage device.

In terms of the volumetric system energy density (Figure 2-10 left), relatively dilute oxygen generators can reach the performance of highly compressed oxygen. In terms of the gravimetric system energy density (Figure 2-10 right), only very concentrated hydrogen peroxide can out-perform the compressed gas. The gravimetric system energy density in the case of compressed oxygen is decreasing with pressure, because the required cartridge wall thickness increases superlinearly with the storage pressure, while the cartridge volume decreases linearly. The volumetric system energy density increases with storage pressure because of compression, but when the pressure approaches the allowable tensile stress, the wall cartridge thickness increases so

much that the cartridge volume becomes significant. In terms of the optimal product there is a tradeoff between the safety requirement and the objective of minimizing weight and/or volume; safe operation requires thick cartridges and a dilute oxygen generator or moderate storage pressures, while energy density maximization requires a highly concentrated oxygen generator or high storage pressures using thin-walled cartridges. Depending on the relative importance of the objectives, either the oxygen generators, or compressed oxygen may be the optimal choice. It can also be seen that the comparison with state-of-the-art batteries regarding the energy density is not very favorable and only highly efficient processes will be able to compete with batteries regarding this metric.

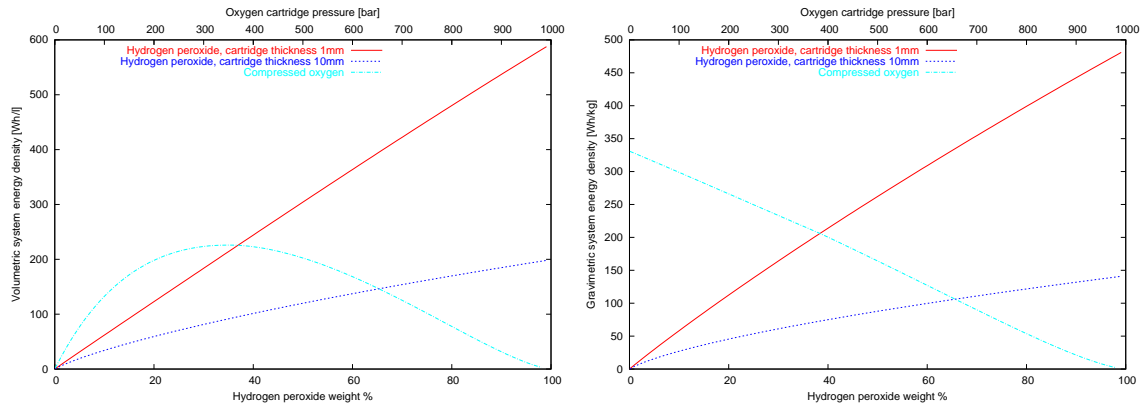


Figure 2-10: Comparison of hydrogen peroxide and compressed oxygen in terms of the volumetric (left) and gravimetric (right) system energy density.

Effect of Scale on Process Performance

The scalability of micropower generation devices is particularly interesting since there are two major scales. One is the nominal power output, which is mainly associated with the device size, and the other is the time between refueling (mission duration), which is associated with the fuel cartridge size. In this case study we present the influence of these two scales on achievable system performance using the volumetric and gravimetric energy densities as metrics. We use the same parameters as in Table 2.6, a volume factor of 10 and a device density of 1kg/l. The fuel cartridge thickness was taken to be 1mm and the density 1.5kg/l. The startup time was assumed to

be $\tau_{startup} = 60\text{s}$ with an auxiliary battery with energy densities of 200Wh/kg and 200Wh/l . As metrics for the scalability we use the volumetric and gravimetric *system* energy densities, where the system includes the power generation device and the stored fuel, as described in Appendix A. Figure 2-11 shows the achievable energy density in $\text{Wh}/(\text{l system})$ and $\text{Wh}/(\text{kg system})$.

For low power outputs the heat losses dominate over the exothermicity of the fuel processing and burning of the fuel cell effluents as well as part of the fuel is needed (Design I). Since the heat generation scales linearly with power output while heat losses scale sublinearly (with a power of $2/3$) the achievable energy density increases significantly with the power output. At a power output of about 0.6W a kink is observed, because for higher power output the heat generation from burning the fuel cell effluents is sufficient (Design II). Above approximately 1.6W the process is exothermic enough, so that the fuel cell effluents need not be oxidized (Design III). The system energy density increases with mission duration and approaches the energy density with respect to the fuel volume/mass because the device size becomes negligible. This case study demonstrates that the influence of scale on process performance is significant; since different processes scale in general differently, the optimal design is also likely to be influenced by the scale.

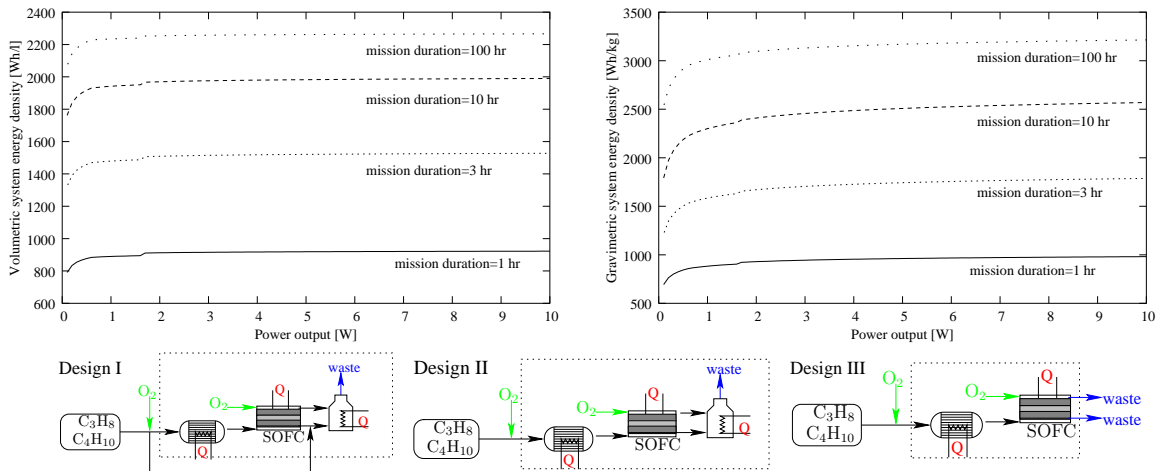


Figure 2-11: Volumetric and gravimetric system energy density of hydrocarbon partial oxidation in combination with a SOFC as a function of mission duration and power output.

Effect of Fuel Combinations and Layout

In this case study we investigate the effect of using a second fuel for heat generation as well as how different layouts can yield significantly different system performances. A process that has been proposed, e.g., [192], is ammonia decomposition to nitrogen and hydrogen and subsequent oxidation of the hydrogen in a PEM fuel cell. A major drawback of this process is that ammonia is corrosive and extremely toxic. From a technological point of view this process has the benefit that the ammonia does not contain carbon, and thus poisoning of the PEM can be avoided without the need for a separation following the fuel processing. However, the process has many drawbacks, including high operating temperatures for the fuel processing reactor and an endothermic fuel processing reaction, so that burning the fuel cell effluents may not provide sufficient heat [201]. Performance improvements can be achieved by the use of a second high energy fuel, e.g., hydrocarbons, for heat generation. We consider two extreme cases of layout, namely that either the two burners are separate and we have remote heat exchange or that the two streams to be burned are combined in a burner which is in thermal contact with the reactor. In [87] it was demonstrated that conversion for the ammonia cracking reaction is essentially complete for residence times in the order of ms and a fuel processing temperature of 650°C. Here we assume complete conversion of ammonia in the reactor and vary the residence time in the reactor in the range 0-100ms. Table 2.7 summarizes the parameter values used and Figure 2-12 shows the results in terms of the fuel energy density.

Even for low residence times combining units into a stack has a significant impact, and thermal integration seems necessary. This case study illustrates that flowsheet design and thermal management, including combination of heat sources and heat sinks, need to be considered simultaneously. The use of ammonia oxidation for heat generation in separate units becomes essentially impossible for high residence times because of the resulting increase in heat losses. The choice of a single fuel or a fuel combination is not obvious; from a perspective of maximizing the energy density fuel combination is very advantageous, but it bears the logistic difficulties of carrying two

fuels. This tradeoff implies that for different applications a different design will be used.

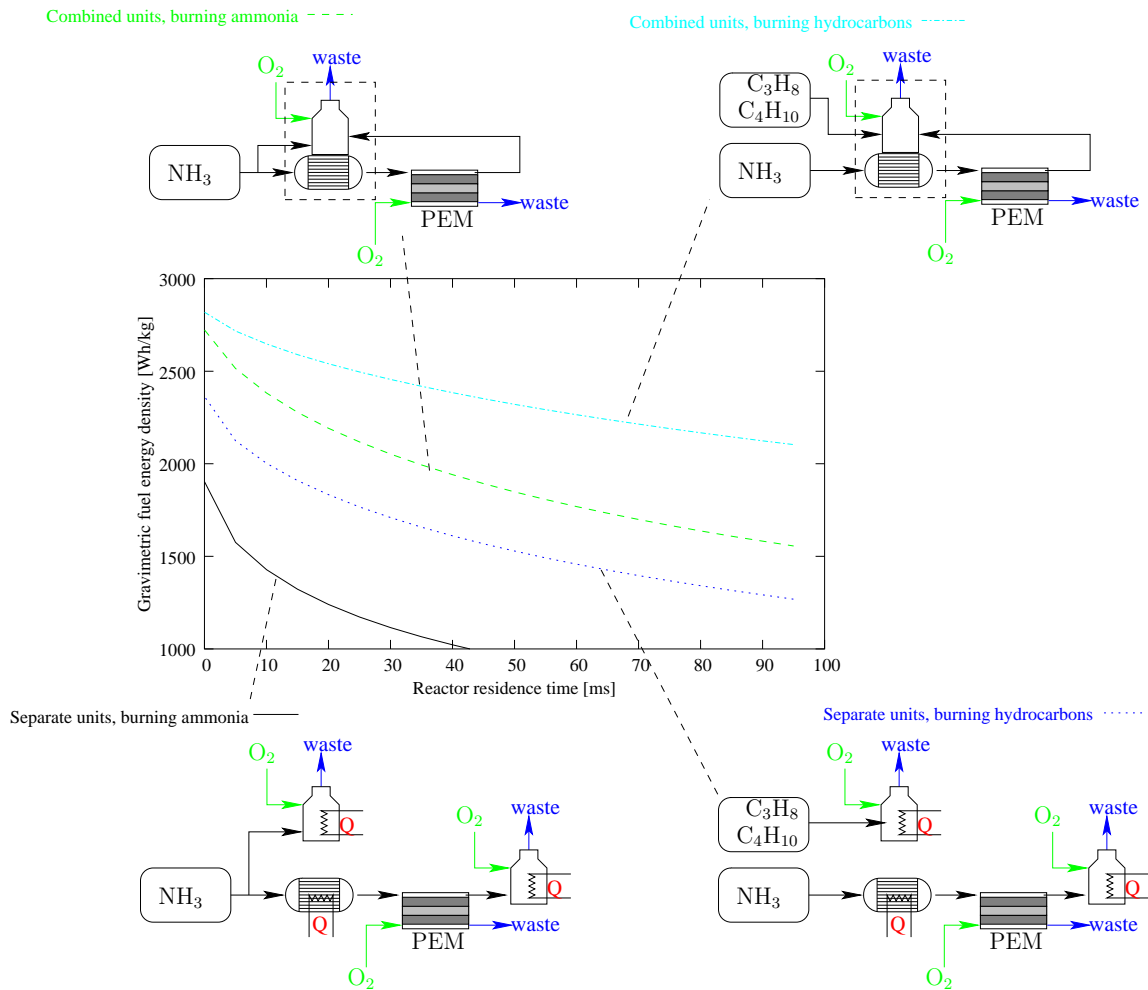


Figure 2-12: Effect of fuel combinations and layout options on gravimetric fuel energy density of an ammonia-cracking based process.

Water Management in DMFC

It is well known that water management is a key consideration in direct methanol fuel cells, e.g., [136]. Here we investigate the effect of internal (through diffusion from the cathode to the anode) and external (through flash separation of the fuel cell effluents) recycling of water. We use an operating temperature of $T = 350\text{K}$, a conversion of methanol equal to $\zeta = 0.6$, a diffusion factor for hydrogen $k_{\text{H}_2}^{\text{trns}} = 1$, a methanol transport coefficient $k_{\text{CH}_3\text{OH}}^{\text{trns}} = 1$ and a water factor $\Psi = 2$, as defined in

Table 2.7: Process parameters for the ammonia cracking case study, Figure 2-12.

| | |
|--|------------------------------|
| Ambient temperature | $T_{amb} = 298\text{K}$ |
| Power output | $PW = 1\text{W}$ |
| Reactor temperature | $T_{op} = 923\text{K}$ |
| Discard temperature from reactor | $T_{out} = 623\text{K}$ |
| Conversion in burners | $\zeta = 0.95$ |
| PEM temperature | $T_{op} = 350\text{K}$ |
| Residence time in burners | $\tau = 1\text{ms}$ |
| Discard temperature from PEM | $T_{out} = 350\text{K}$ |
| Air excess in burners | $\Phi = 1.2$ |
| Conversion in fuel cell | $\zeta = 0.8$ |
| Overall heat loss coefficient | $U = 3\text{W/m}^2/\text{K}$ |
| Residence time in fuel cell | $\tau = 20\text{ms}$ |
| Emissivity (incl. view factor) | $\epsilon = 0.2$ |
| Efficiency of fuel cell | $\eta_{FC} = 0.7$ |
| Air excess in fuel cell | $\Phi = 1.2$ |
| Compression parameter for the air feed | $K_C = 10\text{J/mol/K}$ |
| Burner temperature | $T_{op} = 1000\text{K}$ |
| Discard temperature from burner | $T_{out} = 500\text{K}$ |
| Temperature loss factor | $\chi_{temp} = 0.6$ |
| Heat loss factor | $\chi_{heat} = 0.6$ |

the description of the DMFC in Appendix A. The flash for the fuel cell effluents is assumed to operate at the ambient temperature $T_{amb} = 298\text{K}$. As a figure of merit we use gravimetric *fuel* energy density, because we currently do not have accurate estimates for the sizing of mechanisms for external recycling, e.g., flash and pumps; the energy density is plotted as a function of the overall transport coefficient for water $k_{H_2O}^{trns} = -0.7 - 3$. For simplicity we ignore the cooling load, which we consider separately in the next case study.

We consider four possible product designs. In the simplest case (Design I) no separation or recycling is performed, but rather the fuel cell effluents are directly wasted; in this case the energy density decreases strongly with increasing water transport coefficient, and the DMFC can operate efficiently only in the presence of internal recycling $k_{H_2O}^{trns} < 0$. Design II assumes that half of the liquid part of the fuel cell effluents can be recycled to the anode with a negligible energy penalty; again the energy density greatly depends on the water transport coefficient, but there is a wider range where

the fuel cell can operate efficiently; for $k_{\text{H}_2\text{O}}^{\text{trns}} < 0$, the internal recycling provides sufficient water to the anode. Finally, for Designs III, IV we assume that effective external recycling can be achieved (90% of the liquid fuel cell effluents); in Design III without an energetic penalty and in Design IV with the extremely large energetic penalty of $K_P = 1 \times 10^5 \text{J/l}$; in both cases the fuel cell performance is efficient for a wide range of values for the water transport coefficient; the effect of the energetic penalty is relatively small, due to the small volume of the liquid streams. For different values of the water excess factor Ψ , corresponding to a different composition of the fuel cell feed, the same qualitative behavior is observed with a shift in the achievable energy density and the range of values for the water transport coefficient. It should also be noted that while external recycling, if at all possible, can improve the process performance, there is a tradeoff of increase in the device size and complexity. Whether recycling should be pursued depends on the product specifications and objectives, e.g., the power output and mission duration and the requirements on process performance, as well as on advances in the technology, e.g., the water transport coefficient.

Cooling load in DMFC

Since direct methanol fuel cells operate at a low temperature, removal of the excess heat can be problematic. The same basic parameters as in the water management case study are used here. A zero value for the water transport coefficient is assumed $k_{\text{H}_2\text{O}}^{\text{trns}} = 0$ and no recycling is considered. A residence time of 200ms is assumed. In Figure 2-14 the achievable energy density is plotted as a function of the overall heat transfer coefficient. Significant performance losses are observed unless the passive heat loss coefficient is sufficiently large. As expected, cooling by supplying fresh water is not advisable.

Water Management in Water Reforming Reactions

Steam reforming is widely used in stationary applications, e.g., [106]; this is an interesting alternative because water/steam are relatively inexpensive, and in the case of fuel cells operating at high temperature the heat excess from the fuel cell can be

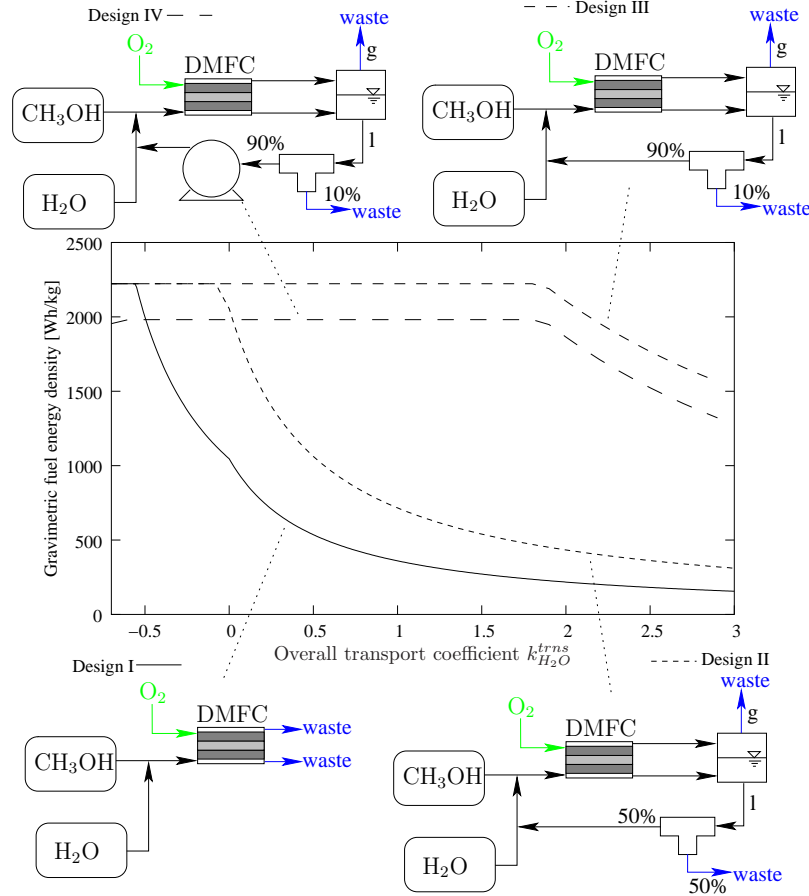


Figure 2-13: Effect of water recycling in a DMFC.

used for the fuel processing reaction. In portable applications, on the other hand, carrying the water strongly decreases the energy density, and therefore steam reforming reactions are not necessarily the optimal fuel processing path. Water separation and recycling is a task that may be impossible to implement at the microscale, but it has the promise of significant improvements in performance because it can minimize the size of the water cartridge. On the other hand, reforming reactions are operated at relatively high temperatures, and therefore recycling of water is associated with a large energetic penalty for vaporization and heating; also recycling an excess of water dilutes the fuel and increases the required device volume, resulting in increased heat losses. The tradeoff between these considerations leads to an optimum recycling ratio for a given requirement for water in the reactor feed.

Here we consider a process based on the combination of a hydrocarbon reforming

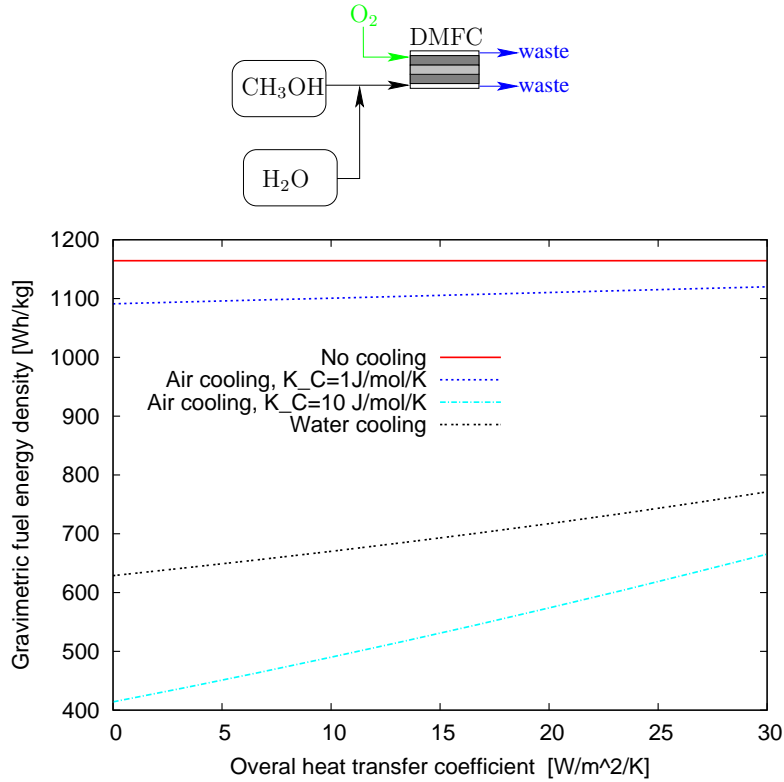


Figure 2-14: Effect of cooling load in a DMFC.

reaction with a SOFC. The water requirement in the reactor feed is specified according to an excess factor Ψ , relative to the complete reforming reaction:

$$N_{\text{H}_2\text{O},in} \geq \Psi (3 N_{\text{C}_3\text{H}_8,in} + 4 N_{\text{C}_4\text{H}_{10},in}).$$

The gaseous components after the flash separation are combusted, and the energy balance is closed by burning hydrocarbons. The reactor, fuel cell and burner are assumed to be in thermal contact, and carbon monoxide and hydrocarbons are assumed to be consumed in reforming reactions in the SOFC. The option of a water cartridge or water breathing system (assuming ambient water) is given. Table 2.8 summarizes the parameter values used.

Figure 2-15 shows the effect of recycling for different stoichiometric compositions of the reactor inlet as a function of the recycling ratio. Depending on value of the water factor Ψ , the optimum is observed at a recycling ratio of about 0.5 – 0.8. In a similar vein to the case of recycling in a DMFC there is a tradeoff between an increase

in the device size and complexity and the improvement of product performance. A water breathing system, even for the large value of the penalty used, can lead to a significant improvement in process performance, again at the expense of additional complexity.

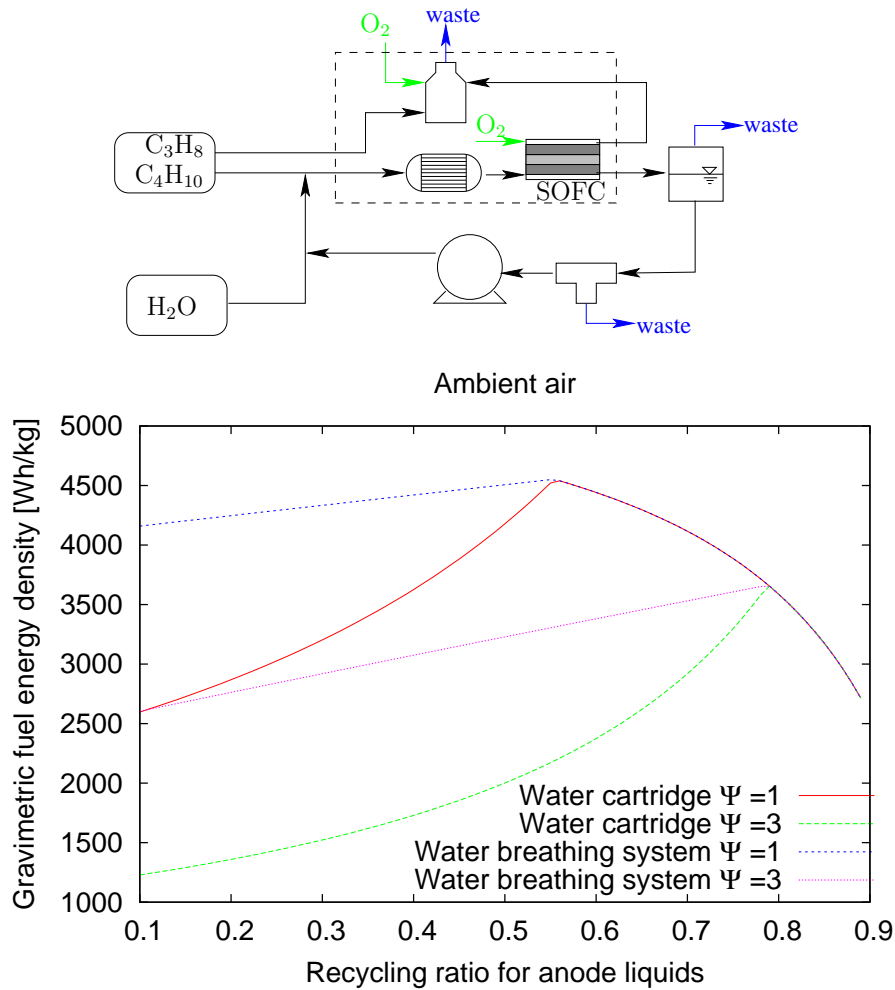


Figure 2-15: Effect of water recycling in the reforming reaction of hydrocarbons.

Methane as a Portable Fuel?

At the macroscale steam reforming of natural gas is the predominant technology for hydrogen production. Methane as a feedstock for hydrogen production has the advantage that no carbon bonds need to be broken and the ratio of hydrogen-to-carbon atoms is maximal among the hydrocarbons. Moreover, compared with methanol and

Table 2.8: Process parameters for water reforming study in Figure 2-15.

| | |
|--|--|
| Ambient temperature | $T_{amb} = 298\text{K}$ |
| Power output | $PW = 1\text{W}$ |
| Reactor temperature | $T_{op} = 1000\text{K}$ |
| Reactor outlet temperature | $T_{out} = 1000\text{K}$ |
| Conversion in reactor | $\zeta = 0.9$ |
| SOFC temperature | $T_{op} = 1000\text{K}$ |
| Residence time in reactor | $\tau = 10\text{ms}$ |
| Discard temperature from SOFC | $T_{out} = 500\text{K}$ |
| Conversion in burners | $\zeta = 0.95$ |
| Residence time in burners | $\tau = 1\text{ms}$ |
| Air excess in burners | $\Phi = 1.2$ |
| Conversion in fuel cell | $\zeta = 0.8$ |
| Overall heat loss coefficient | $U = 3\text{W}/\text{m}^2/\text{K}$ |
| Residence time in fuel cell | $\tau = 20\text{ms}$ |
| Emissivity (incl. view factor) | $\epsilon = 0.2$ |
| Efficiency of fuel cell | $\eta_{FC} = 0.7$ |
| Air excess in fuel cell | $\Phi = 1.2$ |
| Compression parameter for air feed | $K_C = 10\text{J}/\text{mol}/\text{K}$ |
| Burner temperature | $T_{op} = 1000\text{K}$ |
| Discard temperature from burner | $T_{op} = 500\text{K}$ |
| Water factor in fuel cell | $\Psi = 1$ |
| Pump parameter | $K_P = 100\text{J}/\text{l}$ |
| Propane molfraction in feed | 0.5 |
| Energetic penalty for water breathing system | 1 |

formic acid, it has the advantage that it is not oxidized. Based on its simple chemistry it is used in reactor development, e.g., [167]. As a feedstock for man-portable applications it has the major drawback that it is supercritical at ambient temperatures and the most likely storage is as a compressed gas, which leads to low energy densities. In this case study we examine power generation of $PW = 10\text{W}$ in a direct SOFC operating at $T = 1000\text{K}$ under two conditions. In the ideal case we consider complete conversion, no irreversibilities ($\eta_{SOFC} = 1$) and external heat supply; this gives an optimistic upper bound. As a more realistic performance, a conversion of 70% at an efficiency $\eta_{SOFC} = 70\%$ is assumed, closing the energy balance by combustion. In either case we assume storage in plastic container with a maximal stress $\sigma_{max} = 100\text{MPa}$ and a density of $\rho = 1.5\text{kg}/\text{m}^3$. In Figure 2-15 we plot the achievable

volumetric and gravimetric energy density as a function of the storage pressure. The gravimetric energy density is monotonically decreasing with the storage pressure, due to increased cartridge thickness. For low pressures the volumetric energy density increases with increasing storage pressure, due to compression of the gas. For pressures approaching the maximal tensile stress the required cartridge thickness increases disproportionately to the gas compression and the volumetric energy density decreases with increasing pressure. For realistic fuel cell performances very high storage pressures are needed to meet the performance of state-of-the-art rechargeable batteries. Even in the ideal case, battery performance can only be overcome at very high storage pressures.

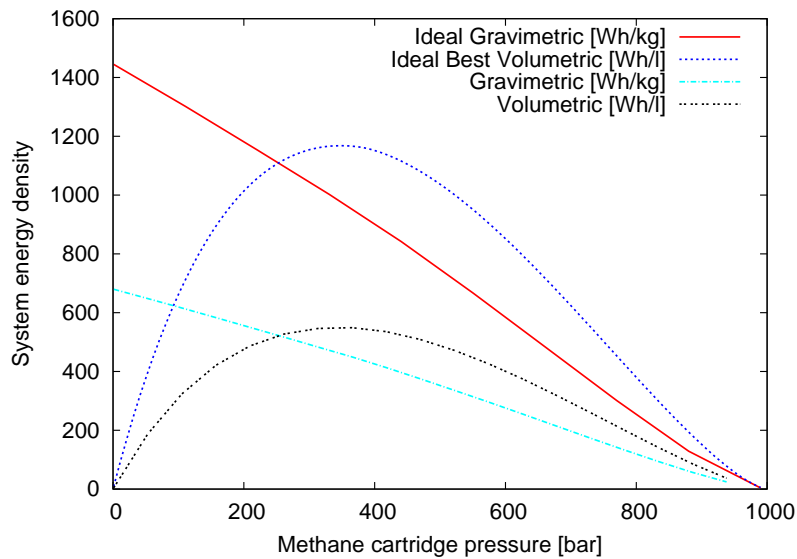


Figure 2-16: Methane as a portable fuel?

2.6.5 Parametric Optimization-Based Case Study

The case studies presented in the previous section were done for fixed process alternatives, i.e., for fixed choices of fuel and fuel reforming and fuel cell type and only in some cases, such as the effect of fuel cell efficiency, the optimal heat integration option was obtained by manually considering the alternatives. The effect of parameter variation on process performance was obtained approximately by simulating for

a finite number of parameter values; the corresponding parameter grid was chosen either a priori or manually by inspection of the sensitivity of the key results to the unknown parameter. Automatic methods to cover the entire parameter space rigorously are in general desirable. More importantly, in many cases it is more interesting to also observe how the parameter value affects the optimal process configuration. As described in Section 3.2, the most appropriate tool for such a parameter study is parametric optimization. Due to the presence of discrete design choices, such as the choice of fuel, and continuous decision variables, such as the fuel flowrates, a mixed-integer formulation is necessary. Moreover there are several sources of nonlinearity resulting in a mixed-integer *nonlinear* parametric program. With a few simplifications a mixed-integer *linear* formulation is possible and the algorithms developed in Section 3.5 can be used.

Since the operating pressures and temperatures are considered as parameters a common source of nonlinearity is eliminated [47]. The full set of process alternatives contains mixers and stream splitters of unknown molfraction and no method is known to represent the mass and species balances with only linear constraints. Species and component balances for a set of alternatives not considering these options can be written linearly if the molar flowrates are chosen as the variables. Note that the hydrogen separation has outlets of known composition and can be written linearly, compare also [17]. The surface area for heat losses is nonlinear in the volume, which is assumed proportional to the molar flowrates. The calculation of the heat losses can be approximated by linearizing around an approximate volume. Finally elaborate calculations relevant to the system energy density, such as the calculation of cartridge volume introduce nonlinearity, so here only the fuel energy density will be considered, assuming an infinite mission duration.

Figure 2-17 shows the set of alternatives considered. To use the parametric optimization algorithms, the models are implemented in C, which is tedious and error prone; therefore a relatively small set of alternatives is considered. Ammonia, methanol and propane/butane are taken as the fuel choices, with the reforming options described in [201] and the options of PEM and SOFC. The option of hydrogen

purification is allowed for the SOFC and is mandated for the PEM (except for the case of ammonia fuel). Autothermal operation is ensured by burning the fuel cell effluents and/or a fuel, allowing for fuel combinations. For simplicity the cooling requirement for the PEM is not considered and the simple energy balances presented in [201] are used. Due to these simplifications, the results should be considered qualitative rather than quantitative. For the reformulation of the bilinear terms between integers and continuous variables the big-M method is used [123].

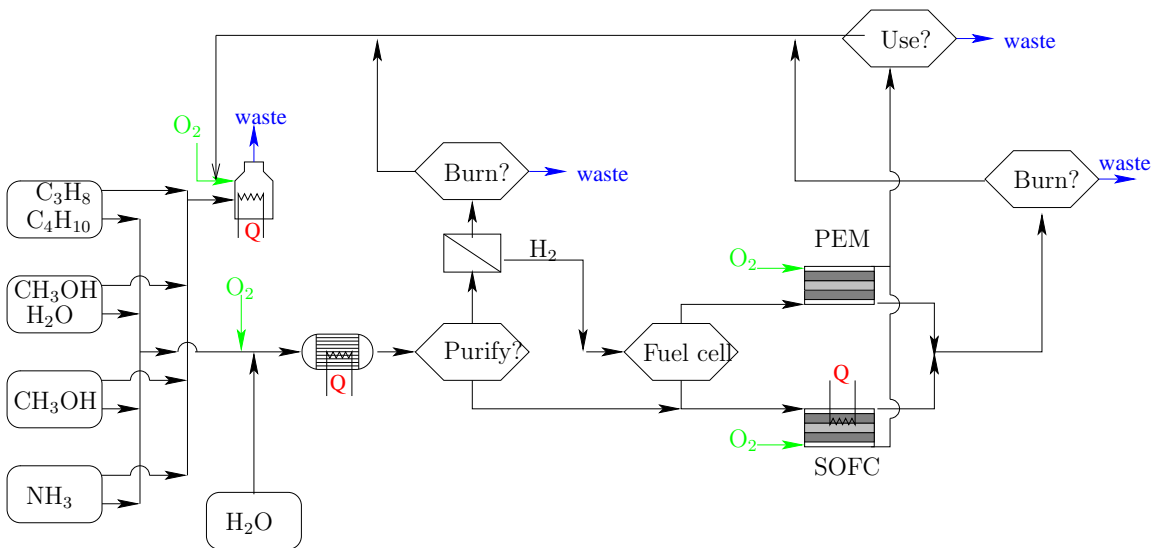


Figure 2-17: Set of alternatives considered for the parametric optimization case study.

From the plethora of possible case studies, here the effect of SOFC efficiency on optimal design and gravimetric energy density is shown. Figure 2-18 shows the gravimetric fuel energy density as a function of the achievable SOFC efficiency for the parameter values indicated in Table 2.9. Note that the same operating conditions (temperature, residence time, conversion) are used for all reactions, which is oversimplifying. Example B.11 in the Appendix B contains the equations corresponding to the case study.

For low SOFC efficiencies, below approximately 35%, or half the PEM efficiency, the optimal process configuration is to use ammonia decomposition, followed by a hydrogen separation and electrochemical conversion of the hydrogen in a PEM. The energy balance is closed by burning the fuel cell effluents and the membrane waste;

Table 2.9: Process parameters for parametric optimization case study in Figure 2-18.

| | |
|------------------------------------|------------------------------|
| Ambient temperature | $T_{amb} = 298\text{K}$ |
| Power output | $PW = 1\text{W}$ |
| Reactor temperature | $T_{op} = 900\text{K}$ |
| Reactor outlet temperature | $T_{out} = 700\text{K}$ |
| Conversion in reactor | $\zeta = 0.9$ |
| SOFC temperature | $T_{op} = 900\text{K}$ |
| PEM temperature | $T_{op} = 410\text{K}$ |
| Residence time in reactor | $\tau = 1\text{ms}$ |
| Discard temperature from SOFC | $T_{out} = 700\text{K}$ |
| Conversion in burners | $\zeta = 0.95$ |
| Residence time in burners | $\tau = 1\text{ms}$ |
| Air excess | $\Phi = 1.2$ |
| Conversion in fuel cells | $\zeta = 0.8$ |
| Overall heat loss coefficient | $U = 3\text{W/m}^2/\text{K}$ |
| Residence time in fuel cell | $\tau = 20\text{ms}$ |
| Emissivity (incl. view factor) | $\epsilon = 0.2$ |
| Efficiency of PEM | $\eta_{FC} = 0.7$ |
| Air excess in fuel cell | $\Phi = 1.2$ |
| Compression parameter for air feed | $K_C = 10\text{J/mol/K}$ |
| Burner temperature | $T_{op} = 900\text{K}$ |
| Discard temperature from burner | $T_{op} = 700\text{K}$ |
| Water factor in fuel cell | $\Psi = 1$ |
| Water excess in reactor | $\Psi = 1$ |
| Pump parameter | $K_P = 100\text{J/l}$ |
| Membrane efficiency | $\eta_{Pd} = 0.8$ |
| Propane molfraction in feed | 0.5 |

note that this configuration is better than directly feeding the reactor products into the PEM, because the gas separation is possible at a high temperature and heat recovery is better. Between an SOFC efficiency of 35% and 67%, the optimal process is partial oxidation of propane/butane and electrochemical conversion of the generated syngas in an SOFC. The reactor and fuel cell exotherm is sufficient to cover for the heat losses. For SOFC efficiencies above 67% the fuel cell effluents need to be combusted to close the energy balance; recall that higher efficiency is associated with less heat generation.

For the low efficiency window, the change in SOFC efficiency is not reflected in the achievable energy density. On the other hand, for the mid efficiency window, the

process performance is highly sensitive to improvements in the SOFC performance, while for the high efficiency window, the sensitivity is lower. Technological improvements of components below a threshold component performance do not affect system performance because other technologies are better. Perfection of component performance is not warranted either, because the effect on process performance is minimal, showing an effect of diminishing returns. Parametric optimization is a valuable tool in identifying these regions and therefore in allocating (research) resources.

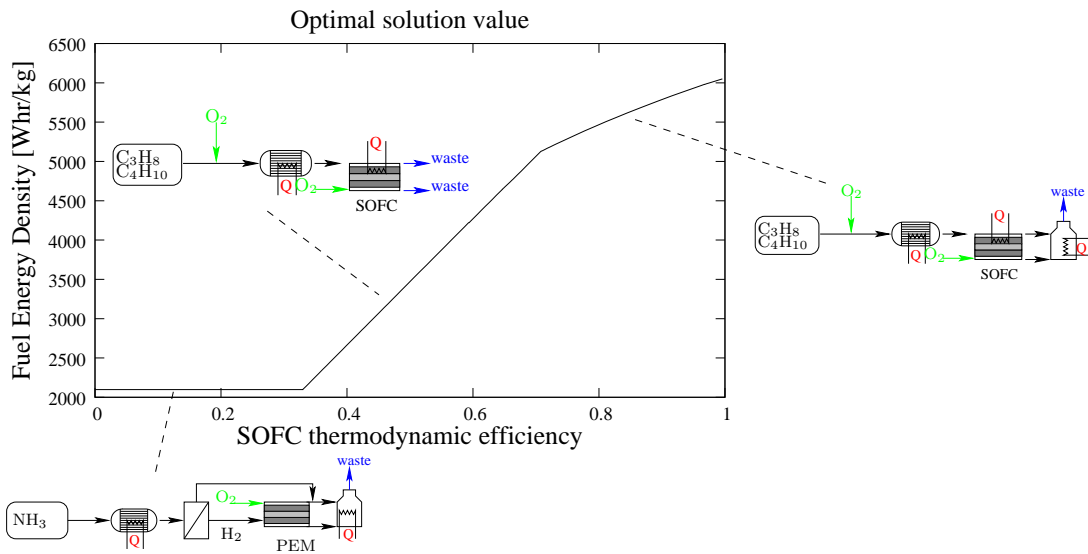


Figure 2-18: Optimal gravimetric fuel energy density as a function of achievable fuel cell efficiency.

2.7 Detailed Modeling for Justification of Modeling Assumptions

The intermediate fidelity modeling case studies [63, 64, 38, 62] use four main modeling simplifications, namely (i) the temperature is spatially uniform in the steady-state case and the transient case, (ii) radial effects can be neglected for the mass- and species-balance and one-dimensional spatial models can be written, (iii) diffusion in axial direction can be neglected for the mass- and species-balance, and (iv) in the transient case pseudo-steady-state species balances can be used. In this section we

provide a justification for these modeling assumptions. We largely follow the notation by Deen [82], as well as some of the scaling techniques presented there. We assume that we can look sequentially at the assumptions; we first consider the assumption regarding the uniform temperature for a transient problem without gas flow and steady-state problems with gas flow; we then discuss the mass and species balances used, assuming a uniform temperature. Compare also Ajmera et al. [18] who show CSTR performance of their reactor and Deshmukh et al. [87] and later Ni et al. [210] who identify CSTR or PFR for some microreactors.

2.7.1 Uniform Temperature at Steady-State

Most of our models assume a stack with a uniform temperature, e.g., reactor and fuel cell share the same temperature. The motivation for this assumption were our simulations of the reactors of Arana et al. [24, 27] and photographs by Arana et al. suggesting an approximately uniform temperature for reactors with catalyst support under proper operation. A simple justification for the uniform temperature assumption is that heat transfer at the microscale is fast and the microfabricated devices of interest are based on silicon, which is a relatively good heat conductor. Note that uniform temperature cannot be assumed for all microreactors; for drastically different approaches, e.g., homogeneous combustion [214, 215, 213], this assumption may not be appropriate, or only valid in subregions. Note also that in the system-level and intermediate fidelity models a lumped model is used for the fluidic connections which do not have uniform temperature.

Because thermal conductivity in silicon is much higher than in the gas phase, catalyst support structures such as posts or slabs result in significantly increased heat transfer; for instance for an equal volume ratio of gas to solid the volume-averaged heat conductivity is approximately half of silicon. Explicitly modeling the support structures is very expensive and as an approximation we often use volume-averaged values. Note that the small length between catalyst support in the radial direction causes approximate thermal equilibrium between the gas and solid (locally).

Convection versus Conduction

The ratio of convection and conduction is given by

$$\frac{k_{av}^{th}}{c_{p,g} \rho_g u L} \approx \frac{10\text{W/m/K}}{10^3\text{J/kg/K} \times 0.5\text{kg/m}^3 \times 1\text{m/s} \times 10^{-3}\text{m}} \approx 20,$$

where we have used conservative estimates; conduction (in the silicon-structures) is dominant over convection (through the gas phase).

Estimation of Maximal Temperature Difference

The maximal temperature within a reactor is essentially determined by the ratio of heat transfer within the reactor and heat losses to the ambient. Because of the high temperatures, heat losses to the ambient are dominated by radiation and the heat transfer per unit area can be approximated as $\epsilon \sigma_{SB} T_{if}^4$, where ϵ is the product of emissivity and view factor, $\sigma_{SB} = 5.67 \times 10^{-8}\text{W/m}^2/\text{K}^4$ the Stefan-Boltzmann constant and T_{if} the temperature at the interface. Heat transfer inside the reactor is characterized by the (average) heat conductivity k_{av}^{th} divided by the characteristic length L . Therefore the maximal temperature difference can be estimated as

$$\Delta T = \frac{\epsilon \sigma_{SB} T_{if}^4 L}{k_{av}^{th}} \approx \frac{0.1 \times 5.67 \times 10^{-8}\text{W/m}^2/\text{K}^4 \times 1300^4\text{K}^4 \times 5 \times 10^{-3}\text{m}}{10\text{W/m/K}} \approx 10\text{K},$$

where we have used conservative estimates. This justifies the assumption of uniform temperature.

Three-dimensional Duct-Reactor Simulation

Here we examine the effect of averaging the heat conductivity and the assumption of uniform temperature using a duct-reactor, of width and height $500\mu\text{m}$ and length 2.5mm , pictured in Figure 2-19. Exploring the symmetry we only model a quarter of the geometry. Our model also includes a 1mm long inlet and outlet to the reactor; we assume that the reactor contains 4 silicon slabs as catalyst support which cover $\frac{2}{5}$ of the width. We assume an inlet velocity of 1m/s . For the simulations we used the

finite element package FEMLAB version 3 [4], and the Navier-Stokes equations with variable density and conduction-convection equations for the energy balance. The chemistry was not modeled, but rather a heat generation term was used. When the slabs are modeled explicitly the heat generation term is introduced as a surface term; for the volume averaged model a volume heat generation is used. Heat losses to the ambient were considered as boundary conditions.

Figure 2-19 shows that using volume-averaged heat conductivity qualitatively and quantitatively captures the effect of increased heat transfer of the catalyst support structure and an explicit model of the slabs is not required. Modeling the slabs explicitly increases the modeling and computational requirements significantly and makes convergence much more demanding. Also the temperature within the reactor portion is essentially uniform. Note that for the temperature profiles, a stretching of the axis is performed.

For the volume-averaged model we explore three cases for the heat generation, namely constant, linear and exponential dependence on the axial coordinate, always with the same overall heat generation. In Figure 2-20 the temperature is plotted along the axial coordinate for these three cases as well as the case with explicit modeling of the slabs. The temperature in the reactor portion is essentially uniform, while in the inlet and outlet (where no silicon structure is present) there is a temperature gradient; also the differences between the different heat generation terms are relatively small.

Two-dimensional CFD Reactor Simulation with Volume-Averaged Heat Conductivity

Here we briefly discuss the results of computational fluid dynamic analysis of a reactor by Arana et al. The reactor geometry along with the obtained temperature profiles are shown in Figure 2-21. Note that the reactor is similar to the ones described in [24, 27] but has a different gas flow pattern; here the two portions of the reactor (for endothermic and exothermic reaction respectively) are concentric, as opposed to parallel as in the design described in the references. For the simulations we used the finite element package FEMLAB, and the Navier-Stokes equations with variable

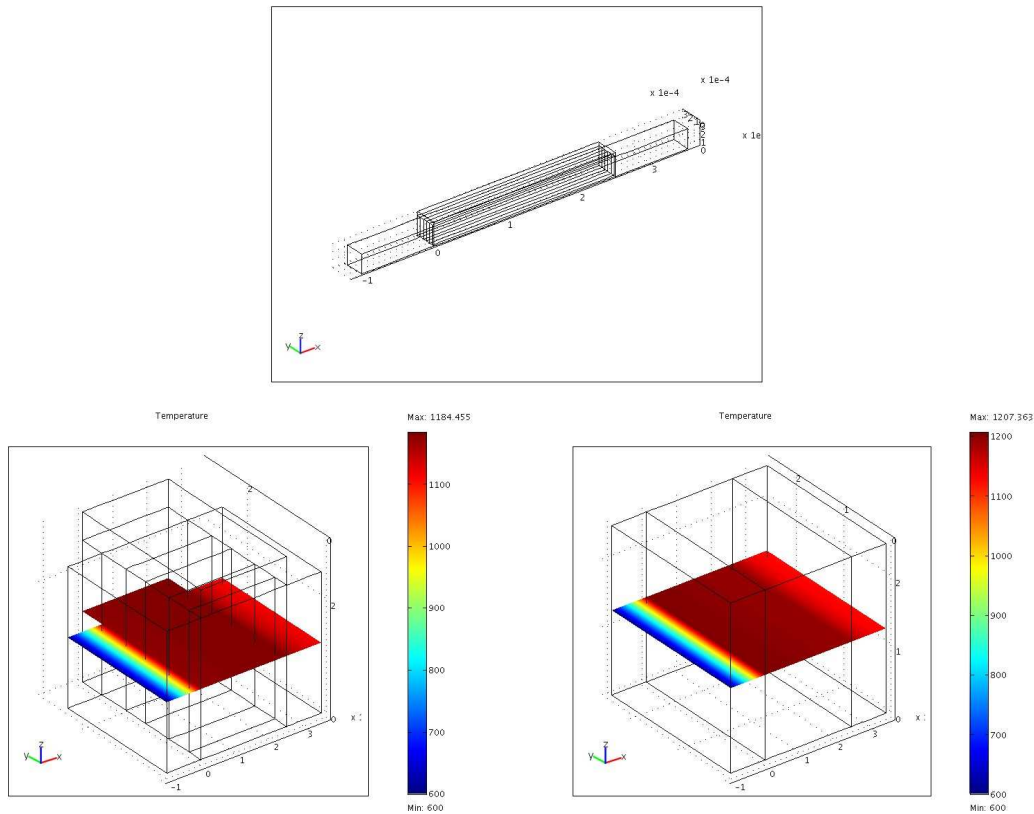


Figure 2-19: Geometry and temperature profiles for explicit modeling of catalyst support (left) and lumped model (right).

density and conduction-convection for the energy balance. The chemistry was not modeled, but rather a heat generation and heat consumption term was introduced, based on complete conversion. Also the catalyst support was not explicitly modeled, but rather a volume-averaged heat conductivity was used. One sees in Figure 2-21 that without catalyst support ($k^{th} = k_g^{th}$) there is a significant temperature gradient, while with catalyst support ($k^{th} = k_{av}^{th}$) the temperature difference within the reactor is relatively small. Note that the temperature profiles are plotted in a stretched geometry.

2.7.2 Uniform Temperature in the Transient Case

In this subsection we examine the assumption of uniform temperature in the transient case. We consider two geometries:

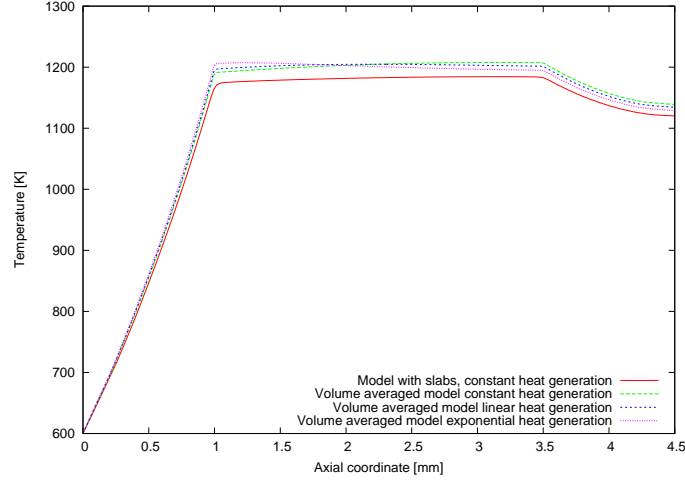


Figure 2-20: Temperature profiles for explicit and average modeling of slabs.

1. A cubic stack of length L with a heat load \dot{Q} applied on one side/face with all other sides/faces adiabatic, see Figure 2-22.
2. A cubic stack of length L , in which a small heating element is placed in the middle, Figure 2-23.

For the former geometry we develop a two-dimensional and a three-dimensional model in FEMLAB and for the latter a three-dimensional model in FEMLAB. Note that the geometries considered are extreme and give a conservative estimate. In realistic reactor designs the heat generation (and consumption) should be distributed in space, resulting in characteristic lengths for heat conduction significantly smaller than the reactor dimensions. The magnitude of the heat load is chosen to heat the stack within a time τ from T_i to T_f ; its numerical value is approximately 1W. We ignore flow through the system and take average physical properties. For both geometries we perform an analysis based on [82].

Scaling Analysis

Neglecting the convective flow and assuming constant physical properties the energy balance reads

$$\rho c_p \frac{\partial T}{\partial t} = k \nabla^2 T \quad (2.1)$$

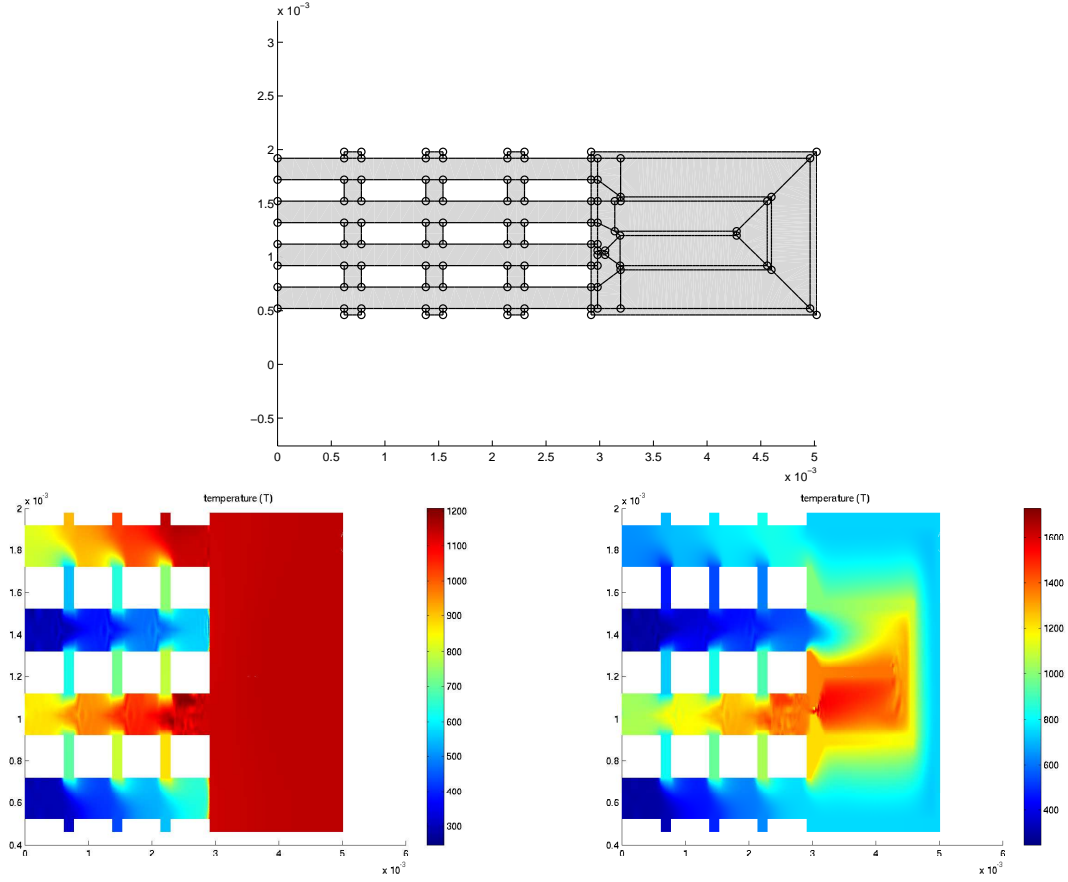


Figure 2-21: Reactor geometry and temperature profiles obtained by CFD simulation corresponding to reactor with and without catalyst support.

We make equation (2.1) nondimensional by using the following scaling:

- The temperature is scaled, so that the initial temperature corresponds to a value of 0 and the final temperature to a value of 1: $\tilde{T} = \frac{T-T_i}{T_f-T_i}$.
- The time is scaled, so that the initial time corresponds to a value of 0 and the final time to a value of 1 $\tilde{t} = \frac{t}{\tau}$.
- The space vector is scaled by dividing through the characteristic length L : $\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L}$.

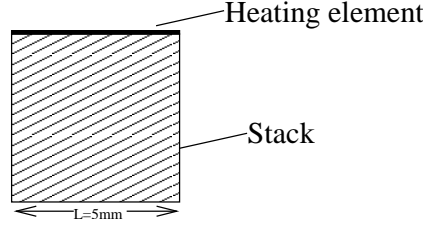


Figure 2-22: Stack with a heating element in the top

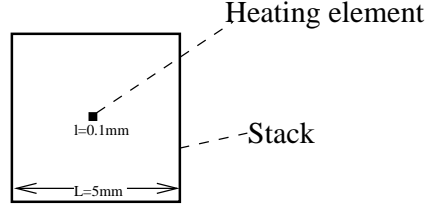


Figure 2-23: Stack with a heating element in the middle

We therefore obtain

$$\begin{aligned}
 \rho c_p \frac{T_f - T_i}{\tau} \frac{\partial \tilde{T}}{\partial \tilde{t}} &= k \frac{T_f - T_i}{L^2} \tilde{\nabla}^2 \tilde{T} \\
 \Rightarrow \frac{\partial \tilde{T}}{\partial \tilde{t}} &= \frac{k}{\rho c_p} \frac{\tau}{L^2} \tilde{\nabla}^2 \tilde{T} \\
 &= \frac{\alpha \tau}{L^2} \tilde{\nabla}^2 \tilde{T}.
 \end{aligned} \tag{2.2}$$

When the ratio $\frac{\alpha \tau}{L^2}$ is high, corresponding to a fast heat transport relative to the time scale of change, the temperature in the stack can be considered uniform in space (the term $\tilde{\nabla}^2 \tilde{T}$ needs to be very small, since it is multiplied by a very large number). A conservative scenario gives (physical properties for Silicon)

- $\rho \approx 2300 \text{kg/m}^3$,
- $c_p \approx \frac{20 \text{J/mol/K}}{0.028 \text{kg/mol}} \approx 700 \text{J/kg/K}$,
- $k \approx 145 - 32 \text{W/m/K}$ in the considered temperature range $T = 300 - 1000 \text{K}$,
- $\tau \approx 100 \text{s}$,
- $L \approx 1 \text{mm} - 5 \text{mm}$,

so that the ratio $\frac{\alpha \tau}{L^2}$ takes values in the order of $10 - 1000$. For characteristic lengths in the order of cm the ratio can be in the order of 1 and the assumptions would not be justified.

FEMLAB Simulation

The transient two- and three-dimensional simulations in FEMLAB validate the previous analysis; the geometries appear of nearly uniform temperature. The parameter values used were:

- $L = 5\text{mm}$,
- $k = 70\text{W/m/K}$, corresponding to an average between air and Silicon,
- $\rho = 1100\text{kg/m}^3$, corresponding to an average between air and Silicon,
- $c_p = 700\text{J/kg/K}$.

As boundary conditions we include heat losses to the ambient and a uniform temperature of 300K is taken as the initial condition.

Figure 2-24 shows the temperature profile for two points in time for the case that a heating element is on the top. The plotted temperature interval is approximately 1K .

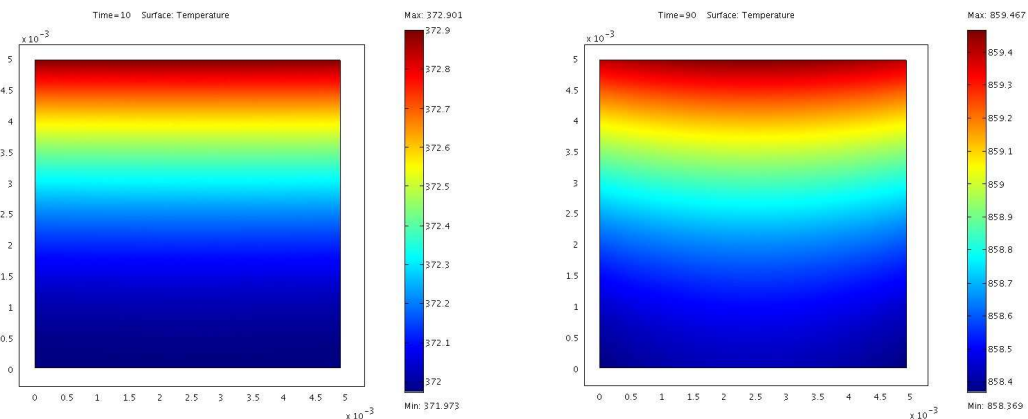


Figure 2-24: Two-dimensional model with heating element on top.

Similar results are seen for the second geometry. Only around the small box there is a small region with temperature gradient; depending on the physical properties

used (Si at 1000K, Si at 300K, volumetric average of Si and air) there is a 15-50K difference in this small region. The bulk of the stack has uniform temperature (less than 5K difference). In Figure 2-25 we plot the temperature for two regions at a simulation time of 70s. Note that the two regions have a different scale; for the region around the heating box we use a temperature range of 15K, whereas for the outside region an interval of 0.1K.

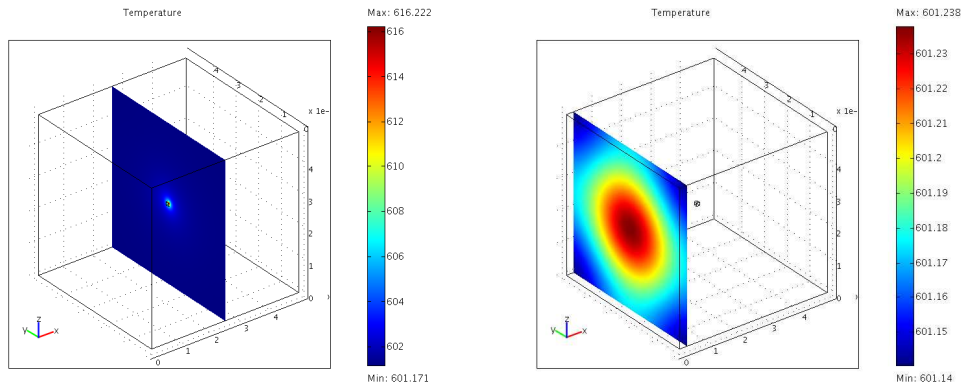


Figure 2-25: Three-dimensional model with heating element in the middle.

2.7.3 One-Dimensional Species Balance

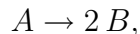
Assuming a uniform temperature we now focus on the following simplifications:

1. Neglecting radial effects (one-dimensional mass and species balances with an average velocity given from mass balance).
2. Neglecting axial diffusion.
3. Pseudo-steady-state species balances.

The first two assumptions essentially give a PFR. The motivation for these assumptions is that preliminary two- and three-dimensional steady-state simulations with FEMLAB (performed in May 2003) had shown a PFR type behavior. A simple justification is that although the flow is laminar, the species diffusion is sufficiently fast to assure a constant profile in radial direction, but slow enough, so that axial diffusion

can be neglected. The pseudo-steady state balance are motivated from the fact that due to the residence time one can neglect the hold-up in the reactor.

For simplicity we will assume first order kinetics and uniform temperature with the model reaction



with molecular masses $M_A = 10 \text{ g/mol}$ and $M_B = M_A/2$. Motivated by the dimensions of the reactor designs by Arana et al. [24, 27] a duct of height 0.2mm and length 5mm is assumed. Based on an inlet speed of 1m/s this results in a nominal residence time of 5ms. We assume first order kinetics in the component A with a reaction rate, following the Arrhenius law with $k_0 = 3 \times 10^4/\text{s}$ and $E_A = 2 \times 10^4\text{J/mol}$.

Pseudo-Steady-State Mass Balance

The continuity equation is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0. \quad (2.3)$$

We make equation (2.3) nondimensional by using the following scaling:

- The time is scaled as $\tilde{t} = \frac{t}{\tau}$ so that the initial time corresponds to a value of 0 and the final time to a value of 1.
- The density is scaled, so that for ambient temperature ($T_0 = 300\text{K}$) and pure component B , i.e., a composition $\mathbf{y}_0 = (0, 1)$, the density corresponds to 1: $\tilde{\rho}(T, \mathbf{y}) = \frac{\rho(T, \mathbf{y})}{\rho(T_0, \mathbf{y}_0)}$. The scaled density can vary from 2 (ambient temperature, pure A), to 1/3 (elevated temperature, pure B).
- The space vector is scaled by dividing through the characteristic length L : $\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L}$.
- The velocity is scaled by dividing through the nominal u_{in} (at ambient temperature): $\tilde{\mathbf{v}} = \frac{\mathbf{v}}{u_{in}}$.

and we obtain

$$\frac{L}{u_{in} \tau} \frac{\partial \tilde{\rho}}{\partial \tilde{t}} + \tilde{\nabla} \cdot (\tilde{\rho} \tilde{\mathbf{v}}) = 0. \quad (2.4)$$

Since we have $L \approx 5 \times 10^{-3}\text{m}$ (taking in x direction), $\tau \approx 100\text{s}$, $u_{in} \approx 1\text{m/s}$, we obtain $\frac{L}{u_{in} \tau} \ll 1$ and we can assume pseudo-steady state, or neglect the hold-up in the reactor. This can be also explained by the fact that the residence time is much smaller than the time scale of change. Note also that the reference value for the density does not appear in equation (2.4). Note that if there is a drastic change in the modeling assumptions, e.g., heating time very fast, or some other fast transient, this analysis needs to be revisited. Also note that for material considerations the short transient period might be important, e.g., by the introduction of shockwaves if the inlet flow is increased as a sharp step function.

Pseudo-Steady-State Species Balance

Omitting the diffusion term $\nabla \mathbf{J}$, which is studied separately, and assuming first order Arrhenius kinetics the species balance equation reads:

$$\frac{\partial C}{\partial t} + C \nabla \cdot \mathbf{v} + \mathbf{v} \cdot \nabla C = C k_0 \exp\left(-\frac{E_A}{RT}\right). \quad (2.5)$$

Equation (2.5) is linear in the concentration, so no reference point is needed for C and the scaling gives:

$$\frac{L}{\tau u_{in}} \frac{\partial \tilde{C}}{\partial \tilde{t}} + \tilde{C} \tilde{\nabla} \cdot \tilde{\mathbf{v}} + \tilde{\mathbf{v}} \cdot \tilde{\nabla} \tilde{C} = \frac{\tilde{C} k_0}{\tau u_{in}} \exp\left(-\frac{E_A}{RT}\right). \quad (2.6)$$

For different order kinetics the reaction term $\frac{\tilde{C} k_0}{\tau u_{in}}$ would contain a reference concentration.

For the numerical values considered, we obtain $\frac{L}{u_{in} \tau} \ll 1$ and we can assume pseudo-steady state. In general we cannot neglect the term $\tilde{C} \tilde{\nabla} \cdot \tilde{\mathbf{v}}$; also since gas expands with the production of species B , the velocity in the term $\tilde{\mathbf{v}} \cdot \tilde{\nabla} \tilde{C}$ is a function of the reactor coordinate. As a result the residence time is lower than the nominal residence time, calculated based on inlet density.

One-Dimensional Convective Flow

In the intermediate fidelity models convective flow is assumed and axial diffusion is neglected. Furthermore a one-dimensional spatial distribution is used, assuming a uniform profile in the radial direction, due to fast radial diffusion. The validity of these assumptions depends on three dimensionless groups:

1. The ratio of height to length, here in the order of 1/100. The influence of this ratio is not trivial, but a small ratio suggests validity of the assumptions.
2. The ratio of convection to diffusion in the axial direction:

$$\text{Pe} = \frac{u L}{D} \approx \frac{1\text{m/s} \times 5 \times 10^{-3}\text{m}}{2 \times 10^{-5}\text{m}^2/\text{s}} \approx 250.$$

Since this is very large axial diffusion can be neglected.

3. The Damkoehler number, expressed as the ratio of reaction to diffusion:

$$\text{Da} = \frac{k d^2}{D} = \frac{d^2}{1/k D} \approx \frac{(50 \times 10^{-6}\text{m})^2}{1 \times 10^{-3}\text{s} \times 2 \times 10^{-5}\text{m}^2/\text{s}} \approx 0.1.$$

So the Damkoehler number is quite small and concentration gradients in radial direction are small. Note that we used $d = 50\mu\text{m}$, as opposed to the tube height, because we assume the presence of catalyst support and the diffusion needs to occur between those supports.

An equivalent analysis can be performed based on the influence of four characteristic times:

1. Reaction time $\frac{1}{k} \approx 1\text{ms}$.
2. Diffusion time in radial direction $\frac{d^2}{D} \approx \frac{(50 \times 10^{-6}\text{m})^2}{2 \times 10^{-5}\text{m}^2/\text{s}} \approx 0.1\text{ms}$.
3. Diffusion time in axial direction $\frac{L^2}{D} \approx \frac{(5 \times 10^{-3}\text{m})^2}{2 \times 10^{-5}\text{m}^2/\text{s}} \approx 1\text{s}$.
4. Residence time in reactor $\frac{L}{v} \approx 5\text{ms}$.

Simulations

For the following simulations it should be noted that we did not model surface reactions. In order to do so, one would need to specify the catalyst support structure.

FEMLAB two-dimensional transient simulation

We formulate a two-dimensional transient problem with a given temperature profile in time $T = 300 + 10t$ for a time period $t = 0 - 70$ s and two modes. Note that formulating a one-dimensional problem in FEMLAB version 3 is not possible for the transient problem with the standard tools.

1. Navier-Stokes equation with varying density in the continuity equation (denoted “Non-isothermal Navier Stokes” in FEMLAB)

- The density is calculated as a function of temperature and concentration $\rho = \frac{P_{amb}}{RT}(y M_A + (1 - y) M_B)$.
- A constant viscosity of $\eta = 2.2 \times 10^{-5}$ Pa s is assumed. Since the pressure drops are not of interest, the viscosity is of little importance. The flow development is quite fast, and for developed flow the profile does not depend on the viscosity.
- The molar fraction is calculated as a function of the temperature and concentration $\frac{C RT}{P_{amb}}$.
- Boundary condition at inlet: specified velocity $u = u_{in} \frac{T}{T_{amb}}$.
- Boundary condition at outlet: convective flow.
- Boundary condition at wall: no slip condition.
- Initial condition $P = P_{amb}, u = 0, v = 0$.

2. Conservative binary convection/diffusion

- A constant diffusion coefficient of $D = 2 \times 10^{-5}$ m²/s.
- Boundary condition at inlet: specified concentration $c = P_{amb}/(RT)$, corresponding to pure A.

- Boundary condition at outlet: convective flow.
- Boundary condition at wall: zero flux.
- Initial condition $C = P_{amb}/(RT)$, corresponding to pure A. This is not necessarily the most plausible assumption, but other initial conditions caused convergence problems. Moreover the initial conditions only affect the first ms.
- First order Arrhenius kinetics $C k_0 \exp\left(-\frac{E_A}{RT}\right)$.

There are some convergence problems at the first integration step, especially when the non-conservative equation is used instead. This might suggest a high index problem and which raises some doubts about the results. For the comparison of the results the conversion at the outlet is calculated as

$$\zeta = 1 - \frac{\frac{\int_{y=0}^d u(x=L,y) c(x=L,y) dy}{\int_{y=0}^d u(x=L,y) dy}}{\frac{\int_{y=0}^d u(x=0,y) c(x=0,y) dy}{\int_{y=0}^d u(x=0,y) dy}}. \quad (2.7)$$

FEMLAB two-dimensional steady-state simulation

In order to isolate the effect of neglecting the transient term we formulate a two-dimensional quasi-steady-state model, where we use the Navier-Stokes equation with varying density and the conservative convection/diffusion equation, similarly to the transient model. The two-dimensional model also allows to check the effect of using a one-dimensional model, by examining the profile in the radial direction. Figure 2-26 shows the molfraction profile across the radial direction at various temperatures for an axial position close to the reactor inlet. The profile is not constant in the cross-section, so the PFR assumption is only approximately valid.

FEMLAB one-dimensional steady state-simulation

By assuming that diffusion in the radial direction is sufficiently fast, one can average over the radial direction. We neglect the transient term (quasi-steady-state assumption), assume Fick diffusion and formulate a one-dimensional model in FEMLAB

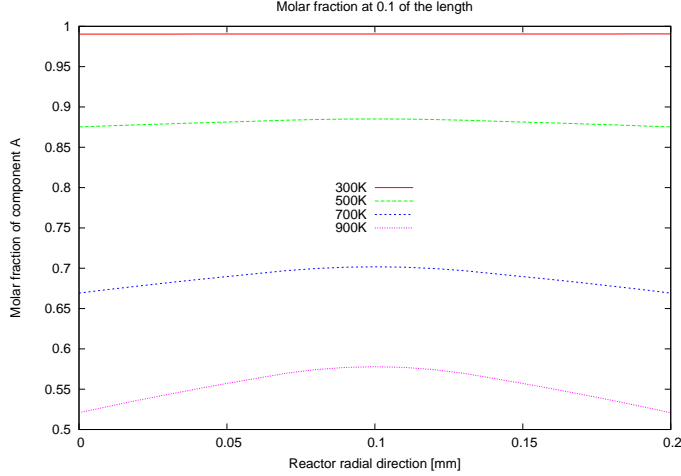


Figure 2-26: Concentration profile from FEMLAB at axial position 0.1

using the diffusion-convection equation

$$C \frac{\partial u}{\partial x} + u \frac{\partial C}{\partial x} = C k_0 \exp\left(-\frac{E_A}{RT(t)}\right) - D \frac{\partial^2 C}{\partial x^2}, \quad (2.8)$$

where the velocity u can be written as a function of the concentration C . FEMLAB provides two options for the diffusion equation; in the so called non-conservative formulation the term $C \frac{\partial u}{\partial x}$ is omitted; in the conservative formulation the complete equation (2.8) is used. Neglecting the term $C \frac{\partial u}{\partial x}$ leads to significant error because of the gas dilatation with reaction. We therefore used the conservative mode with a varying density.

ABACUSS 1d transient simulation

We formulate a one-dimensional spatially-discretized transient model in the process simulator ABACUSS [273, 272]. The mass and species balances are formulated as

$$\begin{aligned} \frac{\partial u}{\partial x} &= \frac{1}{T} \frac{\partial T}{\partial t} + \frac{r}{\tilde{\rho}} \\ \frac{\partial y_A}{\partial t} &= -u \frac{\partial y_A}{\partial x} + \frac{r}{\tilde{\rho}} (-1 - y_A) \\ \frac{\partial y_B}{\partial t} &= -u \frac{\partial y_B}{\partial x} + \frac{r}{\tilde{\rho}} (2 - y_B) \\ \tilde{\rho} &= \frac{P}{RT}, \end{aligned} \quad (2.9)$$

where $\tilde{\rho}$ is the molar density and r the reaction rate. The temperature profile is given as an input variable $T = T_{amb} + 10t$, and its time derivative $\frac{\partial T}{\partial t}$ needs to be inserted directly, otherwise a high index problem is created. We use backward finite differences, e.g., $\frac{\partial z}{\partial x} = \frac{z_i - z_{i-1}}{\delta x}$, with a fixed stepsize δx in the axial direction. As initial conditions for the molfractions we assume $y_A(x \neq 0, t = 0) = 0$, $y_B(x, t = 0) = 1$; this initial condition corresponds to complete conversion before the startup and also allows to observe the effect of the short transient. As boundary conditions we assume $y_A(x = 0, t) = 1$ and a velocity $u = u_{in} T/T_{amb}$.

To analyze the effects of discretization the reaction rate is set to zero and increasingly finer meshes are tested. Figure 2-27 shows the molar fraction at half the reactor length as a function of time for the first milliseconds using 100, 1,000 and 10,000 discretization points. Since diffusion is neglected, the correct profile is a step function from 0 to 1 at time 2.5ms. As expected, coarse discretization introduces significant numerical diffusion. As a conservative measure we therefore use 10,000 grid points for the results presented in the comparison. All changes occur within ms, corresponding to the residence time, and then the influence of the initial conditions is eliminated.

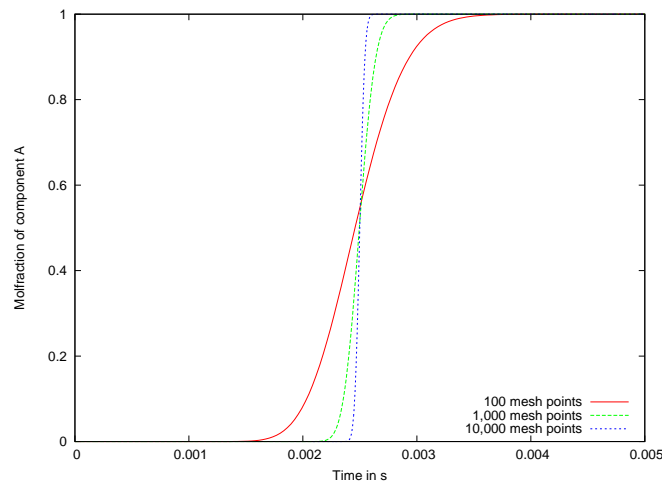


Figure 2-27: Transient profile of molfraction at early time at the reactor middle for different grid sizes (without reaction).

Comparison of the results

In Figure 2-28 we compare the conversion at the outlet as a function of temperature

calculated by the various models, ordered with decreasing number of ignored terms. Based on our numerical experiments we observe the following

- The effect of neglecting the transient terms is indeed very small. The comparison of transient and steady-state two-dimensional models shows a very small difference while the results from transient and steady-state one dimensional models are so close to each other that one can hardly distinguish between them.
- The effect of axial diffusion is also small, as can be seen by the comparison of the one-dimensional ABACUSS and the one-dimensional FEMLAB model as well as by varying the diffusion coefficient in the FEMLAB model (not shown here).
- The comparison of two- and one-dimensional models shows that the effect of averaging the concentration ($2d \rightarrow 1d$) is in the order of a few % (comparison of 2d and 1d models), which is acceptable. Note that even with the existence of a precise kinetic mechanism we can not be sure about the reaction rate, since the catalyst load may not be accurately known.

2.8 Computational Fluid Dynamics for Geometry Improvement

Acknowledgments. *Michael M. Hencke performed, under the author's guidance, the CFD simulations presented in this subsection and implemented and tested the reduced model in ABACUSS.*

Modeling and simulation benefited significantly from interactions with Ole M. Nielsen.

To overcome some limitations of the suspended tube reactor concept by Arana et al. [27], Nielsen et al. [212, 211] designed a suspended micro reactor ($S\mu$ REIII) with a three-dimensional heat exchange portion. This heat exchanger consists of several Silicon slabs connected by parallel SiN tubes with counter-current flow. Here a three-dimensional model of four slabs in the heat exchanger portion is developed and

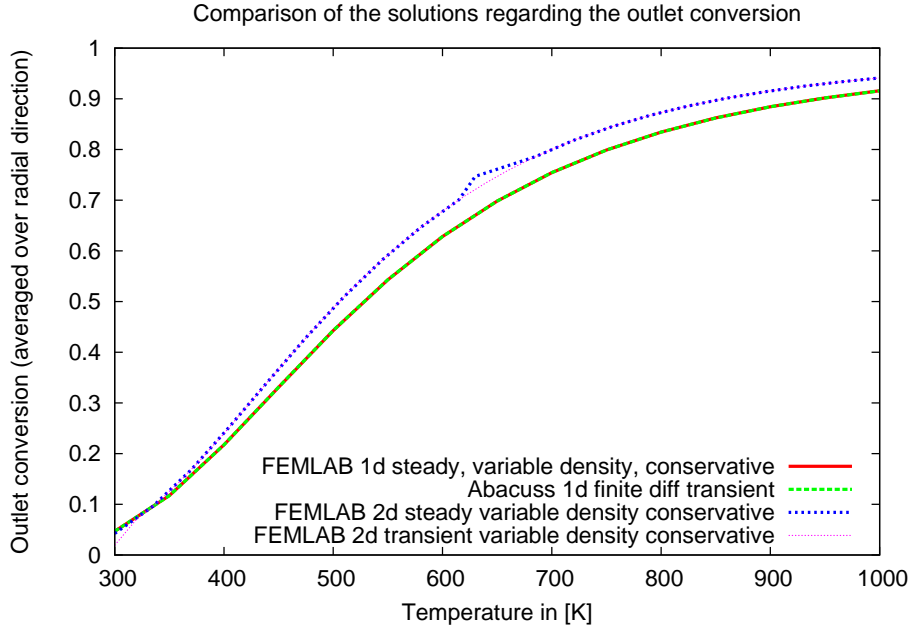


Figure 2-28: Comparison of conversion at the outlet as a function of the reactor temperature for the different models.

simulated in FEMLAB version 3 [4]. The temperature profile allows the development of a reduced model, implemented using our in-house software packages ABACUSS II [273, 272] and DAEPACK [269, 271]. This equation-oriented model includes heat exchange via radiation, conduction along the tubes, and conduction along connecting wires which could not be included in the FEMLAB model. The reduced model is flexible and computational robust and acts as a design tool. Here we focus on the development of the models and how they can be used for design. Other results including the calculation of pressure loss, different boundary conditions and parametric studies of the reduced model are described in Hencke et al. [144, 143].

The geometry of two heat exchanger slabs is shown in Figure 2-29. The vertical sections of the tubes are cylindrical and the tubes then expand slightly into a square duct within the slabs. An exact representation would lead to an explosion of required finite elements, and therefore in the model the tube is assumed rectangular of width $300\mu\text{m}$. The slabs are taken as $1500\mu\text{m}$ in width, $625\mu\text{m}$ in height, separated from each other by a vertical distance of $625\mu\text{m}$. In the three-dimensional models, the overhang of the slabs on all sides is taken as $150\mu\text{m}$, and the center-to-center distance

between parallel tubes is taken as $450\mu\text{m}$. For the equation-oriented simulation, the reacting zone is assumed to be $5\text{mm} \times 5\text{mm} \times 1.875\text{mm}$.

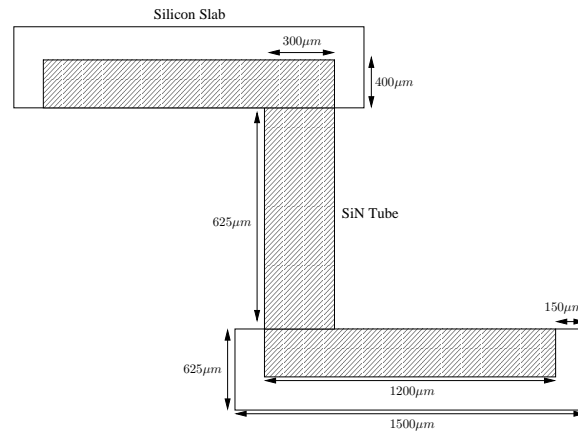


Figure 2-29: Geometry of the heat exchanger (not to scale)

2.8.1 CFD Model

In order to keep the computational requirements to a minimum, two counter-current streams and four slabs are modeled. One flow is from the inlet of the heat exchanger to the reactor, and the other is from the reactor effluent to the outlet of the heat exchanger. The following boundary conditions are used:

- No-slip at all solid walls.
- Inlet velocity = 1 m/s.
- Inlet temperature = 300 K.
- Reactor outlet temperature = 1000 K.
- Reactor outlet velocity = inlet velocity \times outlet temperature / inlet temperature (from continuity and ideal gas equation).
- Outlet pressures = 10^5 Pa.
- Convective flux for outlets.

- Conductive and radiative heat losses to the ambient of the top faces of the top slabs and the bottom faces of the bottom slabs

$$q = -U_{loss} (T - T_{amb}) - \epsilon \sigma_{SB} (T^4 - T_{amb}^4),$$

with $U_{loss} = 3\text{W/m}^2/\text{K}$ and $\epsilon = 0.2$ [201].

- Since FEMLAB version 3 cannot account for heat transfer between boundaries, all other boundaries are approximated as adiabatic.

The wall thickness is very small (few micrometers) and an explicit model is not possible computationally in the three-dimensional case. Hsing et al. [155] have proposed to model thin walls as a two-dimensional domain, but the coupling of two- and three-dimensional regions leads to convergence instabilities and was not pursued. To account for the tube walls, a volume-averaged thermal conductivity for the flow domain is used

$$k_{av}^{th} = k_{air}^{th} + 4 \cdot \frac{2\mu\text{m}}{300\mu\text{m}} \cdot k_{SiN}^{th},$$

where the factor $4 \cdot \frac{2\mu\text{m}}{300\mu\text{m}}$ arises from the geometry of four $2\mu\text{m}$ SiN walls surrounding air flow through a duct of width $300\mu\text{m}$. This approximation is valid under the assumption that the cross-section of the tubes is approximately isothermal.

The default nonlinear solver (UMFPACK) is used and the default mesh generation algorithm. The resulting model has 69,113 unknowns, is nonlinear and the solver at default values fails to converge without good initial guesses. As a convergence scheme we solve the system in three steps. In the first step we only solve for the flow profile at constant density. Using this result as an initial guess, we solve for both the flow and temperature flow profiles at constant density. At the final step we also include the variable density. Each of these three steps takes approximately 30-60 minutes on a AMD 1.2GHz, for a total of approximately 2-2.5 hours for the entire problem. Figure 2-30 shows the velocity profile; note that the effect of increasing density with decreasing temperature is correctly captured. The most interesting result of this model is the fact that the slabs as well as the outlet streams of the slabs appear to

be isothermal as shown in Figure 2-31.

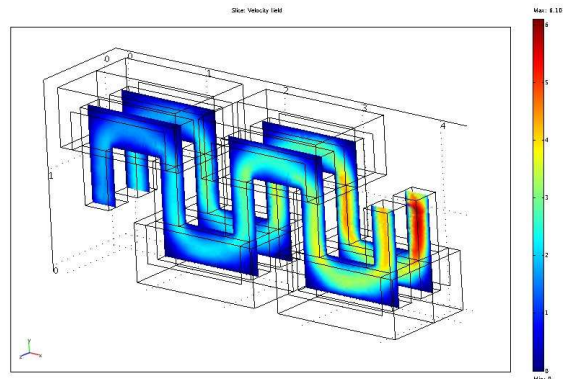


Figure 2-30: Velocity profile for an inlet velocity of 1 m/s.

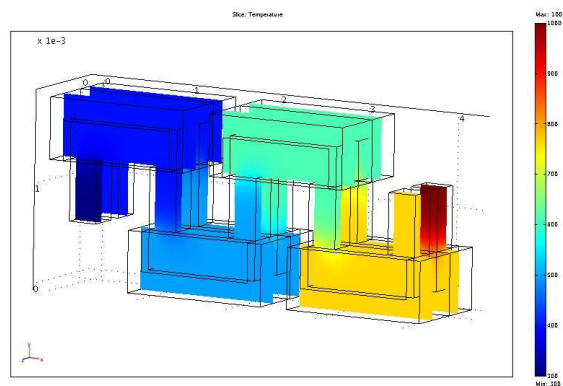


Figure 2-31: Temperature profile for an inlet velocity of 1 m/s.

Simulations under various values of the inlet velocity and reactor effluent temperature are performed to check the validity of these assumptions. For small perturbations of the inlet velocity and reactor outlet temperature the default solver converges in one step using the previous solution as an initial guess; the computation requirement in this case is around 30-60 minutes, depending on the degree of perturbation. For a wide range of values for the inlet velocity and reactor outlet temperature the outlets of the slabs are close to isothermal. As Figure 2-32 shows, for low velocities a significant temperature profile in the tubes is observed, whereas for higher velocities the tubes are nearly isothermal.

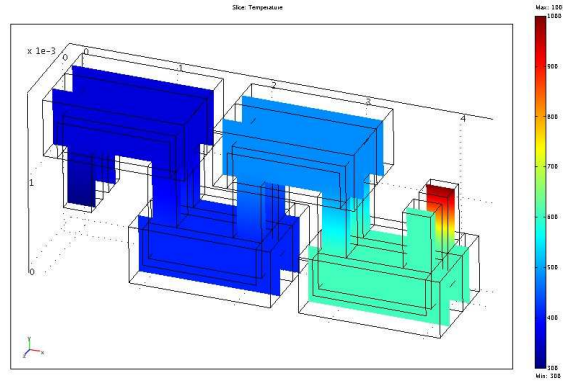


Figure 2-32: Temperature profile for an inlet velocity of 0.01 m/s.

2.8.2 Reduced Model

As the previous results indicate, solving a complete three-dimensional model of the entire heat exchanger geometry is very computationally expensive and possibly intractable with current technology. In addition, within FEMLAB version 3 radiation between individual tubes and slabs is cumbersome to implement. Note also that including this effect would break the symmetry pattern of the model and reduce the sparsity, further increasing the computational requirements and causing convergence problems.

This motivates the development of a reduced, algebraic model, which is more flexible to system geometry and parameters, and also quicker and more robust in its solution. Using equation-oriented modeling languages, it is straightforward to include inner radiation and the presence of wires.

The main assumption for the algebraic model is that the outlet temperatures of the Silicon slabs are close to isothermal. The model consists of three parts, namely a unit model for a single Silicon slab, a unit model for a single SiN tube, and an overall model which defines the connections and energetic interactions between all tubes and slabs. The physical properties are calculated as in the system-level models. The details of this model are given in Appendix A.6. Note that the developed model requires the specification of view factors between slabs and tubes as well as between the tubes and the reactor and the ambient. In [144] we used simplifying rules and

automated the calculation with a Perl script. Note also that for simplicity we neglect the conductive heat transfer from slab to the tube but this can be approximately accounted for.

The reduced model is a valuable tool for improvements of the reactor geometry such as determining a number of slabs that is sufficient for good heat exchanger performance but at the same time is relatively easy to fabricate [211]. Moreover it can easily be implemented and extended in different modeling frameworks [211].

2.9 Intermediate Fidelity Modeling

Acknowledgments. *The work described in this section is mostly due to Dr. Benoît Chachuat. The author's main contributions are in the initial formulation, justification of modeling assumptions and the interpretation of the results.*

Because man-portable power generation processes must operate fully autonomously and automatically without the intervention of operators, steady-state and dynamic operations must be considered carefully. These operational considerations are indeed so important that they are likely to influence the optimal design, following the paradigm of interaction of design and operation. As just an example, increasing the operating temperature significantly increases the heat losses per unit surface area, but also drastically enhances the kinetic rates, therefore decreasing the required reactor and/or fuel cell volume. Accordingly, the design and sizing of the units must be determined simultaneously with the operating policy. Since most power consuming devices are not operated constantly and have rapidly changing power demands, the dynamics and automated operation of portable power production are very important and must be considered thoroughly. For example, it might be necessary to oversize certain units relative to the optimal steady-state design, or exclude processes that exhibit poor transient behavior. Nevertheless, the study of optimal operation at steady-state is a first prerequisite step towards the development of transient models. Furthermore, for those processes exhibiting a rapid transient behavior, the average performance will be most likely dominated by the steady-state performance.

Recall that at the intermediate fidelity we consider fixed process configurations and the models include spatial dependence whenever necessary, but do not require a fully defined geometry. The geometry is captured in a minimal number of design parameters, such as the volume or the surface-area-to-volume ratio of the units. In the following a summary of the methodology and findings from the case studies are presented. For further details the reader is referred to Chachuat et al. [64].

2.9.1 Modeling

We focus on an example process, that consists of a fuel processing reactor, a solid-oxide fuel cell (SOFC) and two burners, fabricated in a single silicon stack fed with ammonia and butane fuels. Ammonia is first catalytically decomposed into nitrogen and hydrogen; the produced gases are fed into the anode of the SOFC and a first air stream is fed to the cathode, and electrical power is produced from the electrochemical reaction. The anode and cathode effluents are finally mixed and fed into burner I, along with a second air stream, for catalytic oxidation. In parallel, a mixture of butane (C_4H_{10}) and air is fed into burner II for catalytic oxidation to produce heat, thus maintaining the stack at a desired, sufficiently high temperature, despite the heat losses and the fact that ammonia decomposition is an endothermic reaction.

This simple process is used as an illustrative example and the proposed methodology can be readily applied to other micropower generation process concepts, provided kinetic mechanisms are available for the reactions considered. Note that although hydrocarbons such as propane or butane have significantly higher theoretical energy densities than ammonia [201], propane/butane partial oxidation for hydrogen production has only been demonstrated partially [174] in microfabricated reactors, whereas ammonia decomposition has been successfully performed with conversions exceeding 90% [26, 118] and kinetic mechanisms for ammonia decomposition [87] and oxidation [228] have been proposed. These considerations therefore justify the choice of ammonia fuel in our initial study, while hydrogen generation from the partial oxidation of hydrocarbon fuels will be the topic of future work.

An alternative to the SOFC would be to use a PEM; since this operates at low

temperatures it would have to be thermally isolated from the ammonia decomposition reactor and the cooling requirement would need to be considered. We consider ammonia decomposition prior to the SOFC, rather than direct oxidation of ammonia inside the SOFC, e.g., [255, 282], to avoid the generation of nitric oxide (NO).

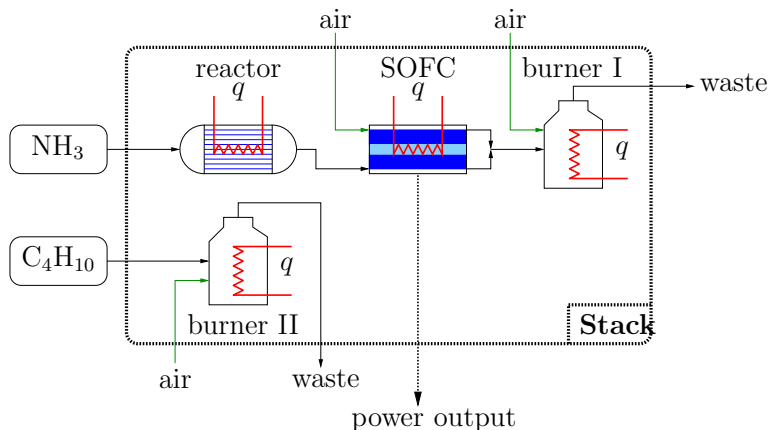


Figure 2-33: Conceptual process flowsheet.

Based on the justifications in Section 2.7, the ammonia decomposition reactor, the solid-oxide fuel cell, and the fuel cell residual burner are modeled as isothermal and isobaric plug-flow reactors with variable density. The pressure inside the stack is assumed uniform, equal to atmospheric pressure and the four units operate at a common temperature. The gas phase is assumed ideal, which is plausible because of the low pressure and high temperature.

Ammonia Decomposition Reactor

For the kinetics of the ammonia decomposition reactor the reduced one-step expression developed by Deshmukh et al. [87] is used. The predictions provided by the reduced rate expression were shown to be in very good agreement with the experimental data obtained by Ganley et al. [118] for an alumina-based microreactor with posts as catalyst support. Note that this expression does not predict chemical equilibria; since at the temperatures considered the chemical equilibrium favors the products and optimal operation is in general sufficiently far away from equilibrium, this is not a significant limitation. Note that it may be limiting if strict discharge constraints

are used for the ammonia. We were also able to reproduce approximately the experimental data reported by Arana et al. [26], although they were obtained for a different catalyst (Ir/Al₂O₃) and a smaller reactor. These considerations justify the choice of the reduced expression rather than the full elementary reaction mechanism.

Fuel Cell

The solid oxide fuel cell (SOFC) consists of two porous electrodes, namely the cathode and the anode, separated by a solid electrolyte. Here, Yttria stabilized Zirconia (YSZ) is assumed for the electrolyte, Nickel (Ni) + YSZ cermet for the anode, and Strontium doped Lanthanum Manganite (LSM) + YSZ cermet for the cathode [158]. In SOFCs, the oxygen ions formed at the cathode migrate through the ion-conducting electrolyte to the anode where they react with the hydrogen (or other syngas) contained in the reformed fuel, producing water (and CO₂) while liberating electrons that flow back to the cathode via an external circuit. Ammonia is considered as an inert in the fuel cell as its residual concentration from the reactor is generally low, because of the discharge constraints.

The open-circuit potential, denoted as U° , is a local quantity as it depends on the actual gas composition at both the anode and the cathode, temperature and pressure. For SOFCs, the open circuit potential is usually very close to the Nernst potential [76], i.e., the difference between the thermodynamic potentials of the electrode reactions:

$$U^\circ = -\frac{\Delta G^\circ(T)}{zF} - \frac{RT}{zF} \ln \left(\frac{y_{\text{H}_2\text{O}}^{\text{an}}}{y_{\text{H}_2}^{\text{an}}} \right) + \frac{RT}{2zF} \ln (y_{\text{O}_2}^{\text{ca}}) \quad (2.10)$$

where $F = 96487 \text{ C/mol}$ stands for Faraday's constant; $z = 2$, the number of electrons transferred in the reaction; ΔG° the standard Gibbs free energy associated with the electrochemical conversion of hydrogen to water. The thermodynamic calculations are the same as in the system-level modeling and are described in Appendix A.

When an electrical current is passed through the cell, irreversibilities arise and the voltage U is reduced due to internal resistances. Such irreversibilities include Ohmic losses, activation polarizations and concentration overpotentials. Ohmic losses arise

due to the resistances encountered by electrical charges in their paths internal to the cell; they are due to (i) in-plane conduction in the electrodes and (ii) cross-plane conduction through the electrolyte [178]. The former contribution is generally low since the electrodes are very good electronic conductors. The high conductivity also makes the cell voltage U approximately uniform throughout the electrodes. On the other hand, resistance to the flow of oxygen ions in the electrolyte is typically among the major contributions to voltage decrease within the cell [178]. Activation polarizations are induced by charge transfer between the electronic and ionic conductors at the anode-electrolyte and cathode-electrolyte interfaces, respectively. They are normally expressed, in implicit form, by the Butler-Volmer equation [217]. Concentration overpotentials are developed due to mass transfer limitations from the gas phase through the electrodes. It is worth mentioning that since the electrodes in the micro SOFC are very thin, of the order of micrometers, mass transfer in the electrodes does not become significantly limiting unless the mole fraction of residual hydrogen (at the anode side) and/or oxygen (at the cathode side) is very low, close to zero [64]. High conversions in the fuel cell are unlikely and therefore concentration overpotentials are not accounted for in the electrochemical model of the micro SOFC. Overall, the losses associated with the irreversibilities of the electrochemical reaction are given by the sum of the overpotential polarizations at the electrode/electrolyte interfaces and the resistance to conduction of ions through the electrolyte:

$$U^\circ - U = \eta_{ohm} + U_{act}^{an} + U_{act}^{ca} . \quad (2.11)$$

The overall system of DAEs for the fuel cell unit contains 12 differential equations as well as 3 algebraic equations. The gas flow in the air channel is considered to be co-current to the one in the fuel channel. In the case of counter-current flow a two-point boundary value problem (TPBVP) would be needed.

Fuel Cell Effluent Burner

The anode and cathode fuel cell outlet streams are mixed together with an air stream, and then fed into burner I for catalytic oxidation of the residual hydrogen and ammonia (Figure 2-33). For the oxidation, the kinetic mechanism proposed by Pignet and Schmidt [228] is used. Nitrous oxide (N_2O) production is not accounted for and the unimolecular decomposition of NO is neglected. The reactions rates for the global reactions are expressed in terms of Langmuir-Hinshelwood kinetics based on Pignet and Schmidt [229]. Note that Hsing et al. [155] use the same model and showed good agreement between simulated and observed results. An excess of oxygen is always assumed.

Butane Burner

Since the ammonia decomposition reaction for hydrogen production is endothermic, and the high operating temperature of cracking and SOFC leads to significant heat losses, the process is complemented with a second burner that produces heat from the combustion of an air-butane mixture (Figure 2-33). A lumped model, based on global mass and species balances is formulated to describe butane catalytic combustion in burner II. The choice of this simplified model is motivated by the fact that butane catalytic oxidation is generally fast. Note also that sustained autothermal propane/butane combustion over platinum in a microreactor has recently been demonstrated experimentally by Arana et al. [27], with high conversions. It is assumed that the combustion reaction takes place to a fixed conversion ζ^{burnII} , and an excess of oxygen Φ^{burnII} is always considered, i.e., $\Phi^{burnII} > 1$ (Table 2.10). The steady-state model for burner II contains 10 algebraic equations. Unlike the distributed model formulated previously for the three units of the ammonia line, the present model does not require specification of burner II's volume for simulation purposes, since the conversion ζ^{burnII} is fixed. However, a value V^{burnII} will be assumed for the energy balance calculation.

2.9.2 Optimal Operation and Design

Optimization Problem Statement

The formulated model has three degrees of freedom related to the design, namely the volumes of the reactor, fuel cell and burner I. In addition, there are seven operational decision variables, namely the stack temperature, the operating voltage and the fuel rates of ammonia, butane and air (into fuel cell and burner I). The resulting mathematical program is classified as a constrained nonlinear optimization problem with hybrid discrete/continuous DAEs embedded. Currently no algorithm exists guaranteeing the global solution of such problems and local optimization methods are applied here. Therefore we have no rigorous guarantee that the solutions obtained are globally optimal for the formulated optimization problem; nevertheless through the use of multistart methods and numerical experience we are confident that they indeed are.

For the case study, as an objective we chose the maximization of fuel energy density, assuming long mission durations. Neglecting the energetic penalty for the air, this objective is equivalent to minimization of the total fuel mass flow

$$M_{\text{NH}_3} F_{in}^{reac} + M_{\text{C}_4\text{H}_{10}} F_{\text{C}_4\text{H}_{10}}^{burnII}.$$

For the heat losses the lumped model described in Section 2.6 is used, including heat recovery $T^{out} = \frac{1}{2}(T - T^{amb})$. For the burner II a volume of $V^{burnII} = 9.6 \times 10^{-10} \text{ m}^3$ is used, which is small compared to the other volumes, and therefore does not significantly affect the heat losses.

For safety reasons, two constraints are defined to limit the emissions of ammonia and nitric oxide (NO) to 25ppm. These threshold limit values are very conservative since they correspond to exposure levels in a typical work environment. Moreover, they are over-restrictive for the transient case [62] or the case that variable power demand is considered [284].

Optimal Operation and Design Results

The model is implemented in Fortran and is solved using the SQP solver NPSOL [121] in connection with DAEPACK [269, 271]. We first discuss results from the base-case and then some parametric case studies, showing the effect of operating temperature and technological improvements. Finally we show that maximizing the energy efficiency leads to a different design than maximizing the energy density.

Base case

The base case of our study corresponds to production of a nominal electrical power $PW = 1$ W. Unless stated otherwise, all the results presented subsequently are obtained for the parameters values given in Table 2.10. The optimization decision vari-

Table 2.10: Parameter values for the steady-state model

| Parameter | Value |
|--|--|
| Surface area in the reactor | $A_{spec}^{react} = 2.20 \times 10^4 \text{ m}^2/\text{m}^3$ |
| Catalyst sites in the reactor | $C_{sites} = 10^{19} \text{ m}^{-2}$ |
| Surface area in the fuel cell | $A_{spec}^{fc} = 2.08 \times 10^3 \text{ m}^2/\text{m}^3$ |
| Electrolyte thickness | $\delta^{io} = 10^{-6} \text{ m}$ |
| Surface area in the fuel cell effluent | $A_{spec}^{burnI} = 2.08 \times 10^3 \text{ m}^2/\text{m}^3$ |
| Volume butane burner | $V^{burnII} = 9.6 \times 10^{-10} \text{ m}^3$ |
| Conversion butane burner | $\zeta^{burnII} = 0.95$ |
| Air excess butane burner | $\Phi^{burnII} = 1.2$ |
| Overall heat loss coefficient | $U^{loss} = 3 \text{ W}/\text{m}^2/\text{K}$ |
| Overall emissivity | $\epsilon^{loss} = 0.2$ |

ables and some performance parameters for the base case are reported in Table 2.11. The temperature that maximizes the fuel energy density is very high, around 1445 K and the achievable energy density is 1180Wh/kg. Operating the system at such a high temperature is however unrealistic, mainly because of material constraints [259]. Therefore, the operating temperature is removed from the list of decision variables and is considered a parameter in the following. However, the development of materials that can operate at higher temperatures is motivated. The performance factors shown in Table 2.11 also warrant some comments. The conversion of ammonia in the reactor (ζ^{react}) and burner I (ζ^{burn}) are very high, close to 100% due to the tight

constraints defined with regard to emissions of NH_3 and NO . The conversions of hydrogen in the fuel cell (ζ^{an}) is also large, greater than 80%. Note that the optimal fuel cell efficiency is significantly lower than 1 and close to the values used in the system-level studies, see Section 2.6. One also notes that the oxygen excess in the cathode compartment (Φ^{ca}) is large enough to provide the required oxygen in burner I for the catalytic oxidation reaction. Here, neither concentrations of hydrogen in the anode compartment nor oxygen in the cathode compartment are limiting, which validates our assumption. Finally, most of the residual hydrogen from the fuel cell is oxidized in burner I.

Table 2.11: Optimal operation and design results for $PW = 1$ W.

| Design decision variables | |
|--|--|
| Reactor volume | $V^{reac} = 0.71 \text{ mm}^3$ |
| Fuel cell volume | $V^{fc} = 35.6 \text{ mm}^3$ |
| Fuel cell effluent burner | $V^{burnI} = 1.67 \text{ mm}^3$ |
| Operation decision variables | |
| Temperature | $T = 1445 \text{ K}$ |
| Voltage | $U = 0.478 \text{ V}$ |
| Ammonia flowrate | $F_{in}^{reac} = 11.76 \text{ sccm}$ |
| Air flowrate to fuel cell | $F_{in}^{ca} = 47.51 \text{ sccm}$ |
| Air flowrate to fuel cell effluent burner | $F_{in}^{burnI} = 0 \text{ sccm}$ |
| Butane flowrate | $F_{C_4H_{10}}^{burnII} = 2.08 \text{ sccm}$ |
| Performance factors | |
| Conversion in reactor | $\zeta^{reac} = 0.999$ |
| Conversion in fuel cell | $\zeta^{an} = 0.839$ |
| Air excess in fuel cell | $\Phi^{ca} = 1.350$ |
| Fuel cell efficiency | $\eta^{fc} = 0.567$ |
| Hydrogen conversion in fuel cell effluent burner | $\zeta_{H_2}^{burnI} = 0.921$ |

Effect of Operating Temperature

A parametric study is carried out by varying the operating temperature in the range $T \in [1000 \text{ K}, 1300 \text{ K}]$. This parameter study indicates that the maximum achievable energy density is very sensitive to the operating temperature as it monotonically increases from 840 Wh/kg at 1000 K, to 1150 Wh/kg at 1300 K.

The effect of temperature on the performance of the system results from a trade-

off between the heat losses and the chemical/electrochemical kinetics. On one hand, the heat losses per unit area are substantially increased when the system operates at a higher temperature and there is a significant loss in the outlet stream, because not all the internal energy of the outlet gases can be recovered. On the other hand, as temperature increases, so do the chemical/electrochemical reaction rates; in other words, increasing the operating temperature allows one to obtain the same conversions while significantly reducing the size of the units. An illustration of these considerations can be found in Figure 2-34. Note that the optimal volume of the system is reduced by a factor of 10 when the temperature passes from 1000 K to 1300 K (as anticipated, one sees that more than 90% of that volume is relative to the fuel cell unit, as the electrochemical kinetics are much slower than the ammonia decomposition and hydrogen/ammonia oxidation reactions). Furthermore, as the size of the system is reduced, the heat losses are reduced from 7 W to 4 W (left plot), thus supporting the counterintuitive result that increasing the operating temperature decreases the heat losses at the same time. For higher temperatures, the significance of the enthalpic loss in the outlet gases overcomes the gains of smaller radiative heat losses. Overall, the large values reported for the heat-losses-to-electrical-power ratio clearly indicate that a large part of the fuel mass is used by the system to maintain the stack at the prescribed temperature. Note that at a higher temperature, the device is more efficient and therefore dissipates less heat, in contrast to the perception that high temperature leads to increased heat dissipation.

Effect of Electrochemistry

In Figure 2-35 we compare the base-case parameter values by Achenbach [13] with recently proposed electrochemical data by Aguiar et al. [16]. These results confirm the very large effect of electrode material and microstructure on the system performance. With the parameter values proposed by [16], a fuel energy density as high as 2000 Wh/kg is obtained, therefore improving the performance by more than 70%. Incidentally, the corresponding fuel cell efficiency increases by approximately 15%, Figure 2-35 (right plot). With these new electrodes, it would then become possible to operate the system at a lower temperature, thus improving the mechanical stability

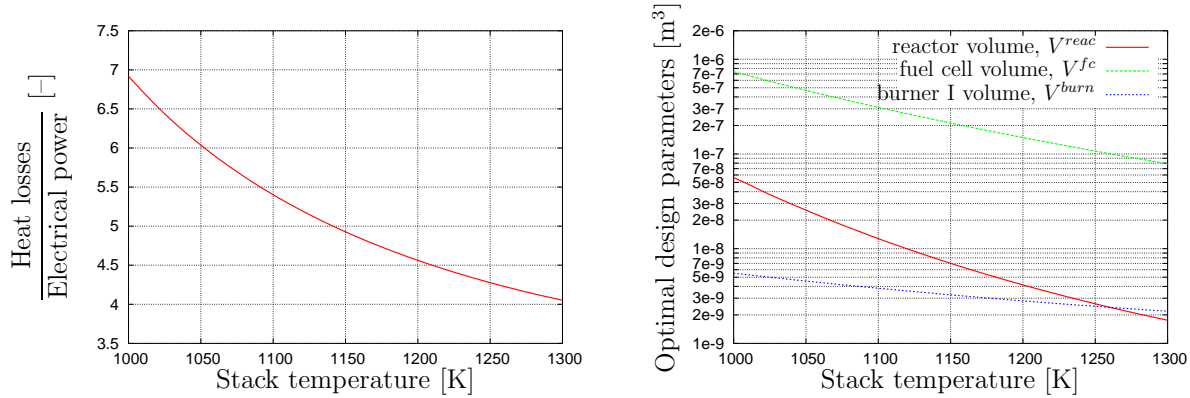


Figure 2-34: Ratio between heat losses and electrical power (left plot) and optimal design parameters (right plot) as a function of the operating temperature, for $PW = 1W$.

of the micromachined SOFC, while still achieving high energy densities.

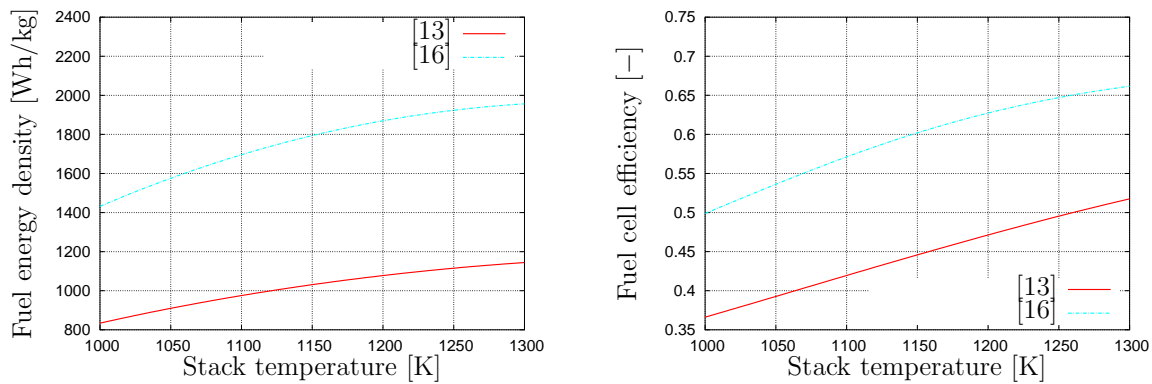


Figure 2-35: Fuel energy density (left plot) and fuel cell efficiency (right plot) as a function of the operating temperature, for literature exchange current density values.

Effect of Electrolyte Thickness

In Figure 2-36 we study the effect of electrolyte thickness. The influence of the electrolyte thickness on system performance is rather small. The variation of the fuel energy density for 200 nm to 10 μm -thick electrolytes is limited to a few percent, even at an operating temperature of 1100 K. Surprisingly enough, the fuel cell efficiency increases as the electrolyte gets thicker. This counterintuitive observation also results from the interaction between design and operation of the system, and warrants further explanations. From the design point of view, the volume of the micro fuel cell increases correlatively to the electrolyte thickness, thus maintaining about the same hydrogen

conversion (around 86% at any temperature); in other words, the current intensity flowing out of the fuel cell remains nearly constant. But as the fuel cell volume increases, so does the active electrochemical surface (since the specific area is fixed, see Table 2.10), and the average current density flowing through the electrolyte is therefore decreased. Accordingly, the activation overpotentials are reduced, which in turn results in a more efficient operation of the fuel cell.

Mechanical stability of the electrodes/electrolyte membrane is among the key issues in the fabrication of micromachined SOFCs. In particular, it has been shown by [259] that the maximum permissible temperature change that prevents the membrane from fracturing or buckling is highly related to the membrane thickness. In this context, the results presented in Figure 2-36 are of the utmost interest, as they would allow one to fabricate slightly thicker membranes, therefore increasing mechanical stability without significantly affecting the process performance.

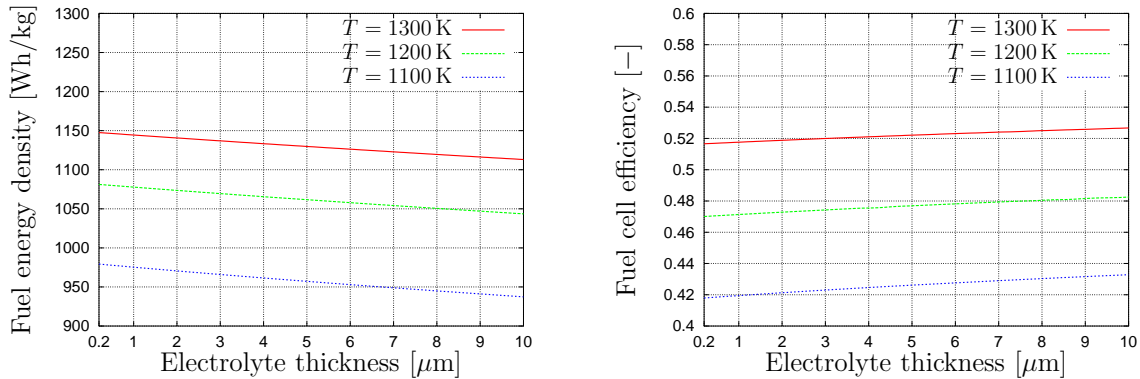


Figure 2-36: Fuel energy density (left plot) and fuel cell efficiency (right plot) as a function of electrolyte thickness.

Maximizing Energy Efficiency vs. Maximizing Energy Density

We have argued that for man-portable power generation devices, the system energy efficiency is not equivalent to the energy density. In this case study we study the difference of the two conceivable objective functions for the ammonia-butane system. For simplicity we will define the energy efficiency as the power output divided by the product of heating value and molar flowrates; maximizing this metric for a given power output is equivalent to minimizing $N_{\text{NH}_3} + 8.4N_{\text{C}_4\text{H}_{10}}$. Maximizing the energy

density on the other hand is equivalent to minimizing $N_{\text{NH}_3} + 3.4N_{\text{C}_4\text{H}_{10}}$. It is clear that the two objectives bear a different weight on each fuel flowrate and running the optimization problems shows that the respectively optimal designs and operations are drastically different, which also motivates the need for consumer oriented engineering.

Table 2.12 summarizes the key findings for an operating temperature of 1000K and a power output of 10W. Note that the energy densities are significantly higher than in the base case where production of 1W was considered. When maximizing efficiency the achievable efficiency is approximately 25% higher compared to the case of maximizing energy density at the expense of a 10% reduction of the energy density. As expected by the weighing of the two fuel flowrates, maximizing the energy efficiency leads to a smaller butane flowrate at the expense of the ammonia flowrate.

A very interesting result is that the optimal device size is significantly smaller in the case of optimal efficiency, mostly due to a decrease in the fuel cell volume. Note that in the case of maximal energy density butane is not as “expensive” and therefore the optimal design allows for a bigger device with a better hydrogen conversion in the fuel cell but higher heat losses, resulting in higher butane flowrates. On the other hand in the case of maximal energy efficiency, the optimal design is such that the ammonia line is nearly autothermal and a small butane flowrate is required to close the energy balance.

The fuel cell voltage in the case of maximal efficiency is significantly lower than in the case of maximal energy density. The most likely explanation is that in the case of optimal efficiency more heat generation is required from the fuel cell, which results in a lower efficiency and therefore lower voltage. This results also emphasizes the point that maximization of the efficiency of one component does not necessarily lead to maximization of the system efficiency.

2.10 Conclusions

Micropower generation devices based on fuel cells are products with the potential of outperforming batteries for man-portable power generation by an order of mag-

Table 2.12: Maximizing energy efficiency vs. maximizing energy density

| Property | Maximal energy density | Maximal energy efficiency |
|----------------------|------------------------|---------------------------|
| Energy density | 1247 Wh/kg | 1107 Wh/kg |
| Energy efficiency | 17% | 21% |
| Fuel cell conversion | 92% | 74% |
| Fuel cell voltage | 0.43 V | 0.33 V |
| Device inner volume | 10.5 cm ³ | 6.9 cm ³ |
| Ammonia flowrate | 89 μ mol/s | 145 μ mol/s |
| Butane flowrate | 12 μ mol/s | 0.8 μ mol/s |

nitude in terms of energy density. There is a plethora of conceivable applications and processes and this results in the need for product engineering. We have presented a methodology for the optimal design and operation consisting of three levels of modeling detail.

At the system-level we consider comparison of alternatives, identification of the most promising technologies and investigation of the influence of technological parameters. Since process components are highly integrated and the heat losses strongly influence the optimal design, the problems of flowsheet design, physical layout, and integration of heat sinks and sources need to be solved simultaneously. The optimal process design depends on product specifications and technological advances.

The performance of water consuming reactions depends strongly on the ability to separate and recycle the waste water. Heat losses are a key issue, especially at small power (<1W) production, and therefore concepts for thermal insulation and optimal heat integration are necessary. Moreover for high-temperature processes efficient operation is only expected for higher power outputs in the order of 10W. As a consequence of the importance of heat management, we can conclude that exothermic fuel processing reactions and fuels with high heat value are more promising and more likely to be efficient. For applications where atmospheric air is not available and compressed oxygen or air has to be used the achievable energy densities are lower and the gas storage required is volume determining. In this case exceeding the energy density achieved by state-of-the-art batteries is only possible for processes operating under

very favorable conditions. For some processes recycling can lead to a relatively small improvement in the performance, provided that efficient mechanisms for the necessary compression are available. The efficiency of power production in the fuel cell influences the energy balance substantially and a threshold value may be observed, above which improvement of the fuel cell efficiency will not lead to improvement of the overall process performance.

The case studies at the intermediate fidelity modeling level identified a strong interaction between device sizing and operation and several counterintuitive influences of the operating conditions on process performance. In particular for a class of devices, increasing the operating temperature reduces the heat losses, due to smaller residence time requirement. The influence of technological improvements, such as a reduction of the electrode overpotentials can be quantified based on our methodology and the most critical goals for development can be identified.

Detailed modeling, in particular using Computational Fluid Dynamics, is a very valuable tool for identifying geometrical improvements. Moreover because of the material properties and dimensions it is possible to derive and justify simplifying models. In particular for a class of devices, the temperature can be assumed uniform in space, which is very helpful as a modeling assumption.

2.11 Future Work

Our modeling framework at the system-level is flexible and could be expanded to include more fuels, e.g., formic acid [241] or different fuel processing mechanisms, e.g., autothermal reforming [258]. An other interesting example is the combination of an exothermic reaction for syngas generation, such as the partial oxidation of hydrocarbons, with a thermophotovoltaic cell that would transform part of the heat excess into power. Also, the consideration of hybrid system with a fuel cell and a battery or capacitor is of interest.

At the system-level we considered separately simulation of the full set of alternatives including several sources of nonlinearity and parametric optimization of a sim-

plified set of alternatives via newly developed algorithms, see Section 3.5. Currently models implemented in the process simulator ABACUSS are manually rewritten to be used in the parametric optimization algorithms. This procedure is tedious and error prone and the full potential of the two tools can only effectively be used if the same modeling framework is used. One possibility is to automatically transform the model from one modeling language to the other and the other is to use advanced interfaces. Moreover, the application of (parametric) mixed-integer nonlinear programming considering the full system of alternatives could lead to useful insights. At the intermediate fidelity, the parametric studies were run by a series of dynamic optimization problems for different parameter values; a parametric optimization algorithm may be significantly more efficient and also provide some guarantees for the entire parameter range.

An important aspect which was not included in this thesis are pressure considerations, including pressure drops and dependence of the process performance on the operating pressure (e.g., at high elevations). Similarly the influence of peripheral components, such as valves and pumps was neglected; in contrast to the macroscale, the energy consumption and influence of these components may be substantial and should be considered. Structural stability considerations are of extreme importance and it would be interesting to include these considerations. Also degradation and durability issues could be considered. Since microfabricated fuel cell systems are at an early stage of development, very limited experimental results are available to validate the models presented. As further experimental results become available, the model predictions need to be compared with the experimental results for validation of the models. This procedure would possibly also suggest refinements to the models.

A very interesting question is the flexibility and robustness of the designs obtained. We believe that the most appropriate tool to study this is the intermediate fidelity models, since these include kinetic models. Flexibility problems are formulated as max-min problems, e.g., [131, 138], and due to the embedded DAEs cannot be addressed rigorously with state-of-the-art algorithms. It would be of interest to either develop algorithms that can address these types of problems, or at least consider these problems heuristically.

Under the assumption that rapid start-up operation is possible, the average performance mainly depends on the steady-state performance of the processes; nevertheless, the transient behavior is extremely important [65, 62]. It is likely that certain processes exhibiting poor transient behavior must be excluded. It might also be necessary to oversize certain units relative to optimal steady-state design, or even include additional units for the sole purpose of start-up, e.g., a catalytic oxidation reactor to generate heat for start-up. In addition to start-up other important transients include switch-overs between power outputs and shut-down. Furthermore, all the case studies consider a fixed power-demand whereas in reality portable electric devices operate over a range of power demands. For instance cell-phones have two main modes, namely talk-mode consuming a few W and standby-mode consuming a few dW. The influence of variable power demand on the optimal design is very interesting [284]. The next step is to also include control considerations and possibly identify interactions between design, operation and control.

The methodology presented is flexible regarding the design objective; nevertheless mostly energy densities were considered in the case studies presented. It would be interesting to consider further design objectives, such as flexibility with respect to the ambient conditions or undesired heat generation. Furthermore economic and environmental calculations were not included in the methodology, but are worth investigating. Also the influence of multiple objectives is interesting.

Finally, it should be noted that the methodology proposed is not the only conceivable and it would be interesting to consider alternatives. For instance automatic connection of the different levels of modeling detail following the paradigm of multi-scale modeling, e.g., [52], is conceivable in the future. At the system-level it would be interesting to study whether and how alternative ideas to the process superstructure can be applied to man-portable power generation; possibilities include attainable regions [152, 122, 107], phenomena-based process synthesis [220] and the state-space approach [32]. Reduced models based on linearization methods or input-output models are also conceivable. At the CFD-level it would be interesting to consider the application of structural optimization techniques.

Chapter 3

Parametric Optimization

3.1 Introduction and Literature Review

Mathematical programs often involve unknown parameters and the task of parametric optimization is, in principle, to solve the mathematical program for each possible values of these unknown parameters $\mathbf{p} \in P$. Discretization of the parameter range, is not rigorous in general, since there is no guarantee for optimality between the mesh points. Moreover discretization on a fine mesh is a very expensive procedure, especially for highly dimensional parameter spaces.

Algorithms for parametric optimization typically divide the parameter range $P \subset \mathbb{R}^{n_p}$ into regions of optimality, also called areas [139], or critical regions [117]; for each region either the problem is infeasible or a qualitatively invariant solution, typically a smooth function of the parameters, is optimal. The notion of qualitatively invariant solution depends on the specific case. In mixed-integer linear programs for instance, it means an optimal integer realization along with an optimal basis for this realization. In general, the number of optimality regions cannot be bounded above by a polynomial in the instance size even for a single parameter [207].

Parametric optimization has several applications [117] including waste management [162] and fleet planning [163]. Recently, Eppstein [102] introduced the notion of inverse parametric optimization where the values of parameters that result in a given solution are searched for. Wallace [277] has argued that parametric optimiza-

tion is valuable for decision making when the value of the parameters is not known during the optimization phase but known during the decision making phase. Within chemical engineering parametric optimization has been mainly used for applications of model-predictive control, e.g., [230, 231, 40] and for process synthesis under uncertainty, e.g., [224, 225, 10, 34]. Our interest for parametric optimization is as a tool for resource allocation decisions, as described in Section 3.2.

The focus of this thesis is on parametric mixed-integer linear programs (MILP)

$$\begin{aligned}
f^*(\mathbf{p}) &= \min_{\mathbf{x}, \mathbf{y}} (\mathbf{c}^x(\mathbf{p}))^T \mathbf{x} + (\mathbf{c}^y(\mathbf{p}))^T \mathbf{y} \\
\text{s.t. } & \mathbf{A}^{1x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{1y}(\mathbf{p}) \mathbf{y} = \mathbf{b}^1(\mathbf{p}) \\
& \mathbf{A}^{2x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{2y}(\mathbf{p}) \mathbf{y} \leq \mathbf{b}^2(\mathbf{p}) \\
& \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{y} \in \{0, 1\}^{n_y}
\end{aligned} \tag{3.1}$$

where $\mathbf{x}^{UP} \in \mathbb{R}^{n_x}$, $\mathbf{c}^x(\mathbf{p}) \in \mathbb{R}^{n_x}$, $\mathbf{c}^y(\mathbf{p}) \in \mathbb{R}^{n_y}$, $\mathbf{A}^{1x}(\mathbf{p}) \in \mathbb{R}^{m_1 \times n_x}$, $\mathbf{A}^{1y}(\mathbf{p}) \in \mathbb{R}^{m_1 \times n_y}$, $\mathbf{A}^{2x}(\mathbf{p}) \in \mathbb{R}^{m_2 \times n_x}$, $\mathbf{A}^{2y}(\mathbf{p}) \in \mathbb{R}^{m_2 \times n_y}$, $\mathbf{b}^1(\mathbf{p}) \in \mathbb{R}^{m_1}$, $\mathbf{b}^2(\mathbf{p}) \in \mathbb{R}^{m_2}$. Note that in deviation from the standard form we allow inequality constraints and upper bounds on the variables, to show how these can be treated efficiently. Including nonzero lower bounds would make some of the discussions more cumbersome, but would not alter anything essential. In some cases we restrict the discussion to finite upper bounds $\mathbf{x}^{UP} \in \mathbb{R}^{n_x}$, while in other cases no bounds are required. As is done in most algorithmic contributions, the host set of the parameters is assumed to be a unit hypercube $P = [0, 1]^{n_p}$. This is essentially equivalent to assuming that the host set is compact, and excludes unbounded parameter ranges, e.g., $p \in (-\infty, +\infty)$.

When the integer variables are fixed, or relaxed to an interval, a parametric linear program is obtained, which is an important problem in its own right, and can also be used as a subproblem in an algorithm for the solution of (3.1).

Two interesting special cases of the general case are the cost vector case and the right hand side case. In both of these cases the matrices \mathbf{A}^{1x} , \mathbf{A}^{2x} , \mathbf{A}^{1y} , \mathbf{A}^{2y} do not depend on the parameters \mathbf{p} ; in the former case also the right hand side vectors

$\mathbf{b}^1, \mathbf{b}^2$ are parameter independent, while in the latter case the cost vectors $\mathbf{c}^x, \mathbf{c}^y$ are parameter independent. The cost vector case has the benign property that the feasibility region does not depend on the parameter. As a consequence the optimality regions of a given basis are (convex) polyhedra and the optimal solution function is piecewise-affine and concave, see Section 3.4.1. In the right hand side case convexity of the optimal solution function is easily established [119]; moreover the optimality region of a given basis can be calculated relatively easily [224, 78].

Parametric optimization is a mature field with many contributions. Most of the theoretical properties were established by the 1980's but in recent years there has been a significant activity in algorithmic contributions. There are several textbooks and review articles; Gal [117] and Dinkelbach [91] consider linear programs; Bank et al. [35] and Fiacco [110] consider the nonlinear case; Geoffrion and Nauss [119] and Greenberg [128] consider mixed-integer programs.

The right hand side case has been addressed assuming an affine dependence on the parameter for mixed-integer linear programs for one or many parameters [119, 162, 216, 224, 225, 12] and also for parametric nonlinear programs [93, 95, 94]. For the cost vector case several theoretical results are available [119]. Gusfield [135], Fernández-Baca [108] and Fernández-Baca and Srinivisan [109] consider purely integer programs with two parameters affecting the objective function. Haneveld et al. [139] consider an LP with two coefficients in the objective function. Gal [117] discusses parametric LP with a single and many parameters. Hale and Qin [137] consider nonlinear programs. For mixed-integer linear programs a well-known algorithm for a single parameter affecting the cost vector is based on intersections of the objective functions of feasible points [101, 162, 164]. In Section 3.4.3 we describe an extension of this algorithm for the case of multiple parameters; the reason that such an extension is not available in the literature is most likely in the large number of MILP calls required. We give a bound on this number in terms of the number of vertices of the optimality polyhedra.

The algorithmic approaches for parametric MILP can be divided into two broad classes. In the first class, algorithms for the solution of a MILP are altered to solve the parametric MILP. For instance Ohtake and Nishida [216] solve the right hand

side case of parametric MILP by a branch-and-bound (B&B) on the integer variables with a parametric linear program at each node. Methods based on this principle have the promise of being relatively computationally efficient if the formulated parametric subproblems are only slightly more expensive than their fixed-parameter counterparts. The other broad class is to use MILP calls for fixed parameter values and process the result postoptimally. This is for instance employed in the well-known intersection-based algorithm for the cost vector case, see Section 3.4.2. Methods based on this principle can take advantage of sophisticated commercial MILP solvers and are also relatively easy to implement.

To our best knowledge no algorithm exists for the solution of the general case of parametric MILPs. Extension of the available algorithms for the right hand side and cost vector case is nontrivial because the general case does not have the benign properties of the special cases. Dinkelbach [91] proposed an algorithm for parametric LPs based on an extension of the simplex method from real valued coefficients to rational functions of the parameters. Post-optimal sensitivity analysis of the matrix coefficients of nonbasic columns is covered in linear programming textbooks, e.g., [42]. Gal [117] reviews the case that a single column or a single row of the matrix depends on the parameter; in this case an analytical inversion of the parametric matrix is possible based on a formula by Bodewig [51]. Freund [115] proposed to obtain post-optimal sensitivity information for parametric linear programs through Taylor series expansions and Greenberg [129] considers post-optimal sensitivity analysis from interior solutions via duality. In Section 3.5 we first propose two algorithms for parametric LP, based on the algorithm by Dinkelbach [91] and then two algorithms for parametric MILP. We then discuss extensions to the nonlinear and multiparametric case.

3.1.1 Complexity of Parametric Optimization

Murty [207] has shown that the complexity of parametric optimization cannot be bounded by a polynomial even in parametric linear programs with a single parameter. Murty constructs a right hand-side parametric linear program with $2n$ variables and

n constraints:

$$\begin{aligned}
& \min_{\mathbf{w}, \mathbf{z}} \mathbf{c}^T \mathbf{z} \\
& \text{s.t. } \mathbf{I}\mathbf{w} - \mathbf{M}\mathbf{z} = -\mathbf{b}^0 + \lambda\mathbf{b}^1 \\
& \mathbf{w} \geq 0, \quad \mathbf{z} \geq 0, \quad \mathbf{w}, \mathbf{z} \in \mathbb{R}^n,
\end{aligned} \tag{3.2}$$

where \mathbf{I} is the identity matrix of size n ($I_{ij} = 1$ for $i = j$ and $I_{ij} = 0$ for $i \neq j$) and \mathbf{M} is a lower triangular matrix of size n with all elements on the diagonal equal to 1 and all below the diagonal equal to 2 ($M_{ij} = 1$ for $i = j$, $M_{ij} = 0$ for $i < j$ and $M_{ij} = 2$ for $i > j$). The constant part of the right hand side is the unity vector ($b_i^0 = 1$ for $i = 1, \dots, n$) and the parameter dependent part contains decreasing powers of 2 ($b_i^1 = 2^{n-i}$ for $i = 1, \dots, n$). The cost vector \mathbf{c} contains decreasing powers of 4 ($c_i = 4^{n-i-1}$). The parameter λ is allowed to take all real values $\lambda \in (-\infty, \infty)$. Murty proves that for $\lambda \in [0, 2^{n+1}]$ there are 2^n intervals in each of which a different solution is optimal. We can therefore scale the parameter using $\lambda = 2^{n+1}p$ and obtain the standard form where $p \in [0, 1]$. The computational requirement of storing and returning the answer to the parametric program is nonpolynomial and therefore no polynomial-time algorithm is possible.

A subtle point is that in (3.2) the data $(\mathbf{b}^0, \mathbf{b}^1, \mathbf{c})$ contain numbers that are exponential in n but nevertheless the size of the instance is a polynomial of n . The size of a problem instance according to Nemhauser and Wolsey [208, chapter I.5] is the number of bits required to store the data. The amount of data to store is $3n$ for the cost vector and right hand side, n for the identity matrix and $\frac{n(n+1)}{2}$ for the matrix \mathbf{M} . The largest integer to store is 4^{n-1} and this requires $2n$ bits. Therefore the size of the instance is in the order of n^3 . Thus the claim that the complexity can not be bounded by a polynomial is true. Note also that polynomial-time algorithms exist for the solution of (3.2) for a fixed parameter value despite large values for the data. Moreover the exponential size does not affect the performance of these algorithms. Compare Section 1.6 in Wolsey and Nemhauser [208] where it is stated that the dependence on $\max |b_i|$ and $\max |c_i|$ can be eliminated.

Note also that, since a polyhedron in standard form has a factorial number of vertices, it is conceivable to construct an example with a factorial number of optimal bases without resorting to coefficients of exponential size, nor to real coefficients.

An extension of the results of Murty [207] to the cost vector case seems straightforward by duality. Moreover the general case and the mixed-integer case are generalizations of the right hand side parametric LP. As a consequence any algorithm for the problems considered in this thesis cannot show polynomial complexity. Rather than basing the computational complexity on the size of the instance, it is probably more appropriate to compare the computational requirement with the computational requirement of solving as many optimization problems at fixed parameter values as there are optimality regions. For instance, in the cost vector case of MILP with a single parameter the intersection-based algorithm [101, 162, 164] requires a number of MILP calls that is approximately twice the number of optimality regions of the particular instance.

3.2 Parametric Optimization for Resource Allocation in R&D

In research and development (R&D) resources are limited and their optimal allocation is desired. In particular the choice between which alternative products/processes to develop is of interest. In the example of micropower generation questions of interest include “should solid-oxide fuel cells be developed?”, “is it better to focus on separation membranes or on catalysis development?”, “should heat insulation or heat recovery be improved?”. Suppose in general that a model of a system under development with many potential components is given and the uncertain parameters describe the performance of the various components. Parametric optimization quantifies the influence of these parameters on the system performance and optimal design including configuration of the components. Identifying the most important parameters enables determination of whether it is worthwhile to pursue improvement of a given compo-

ment. Post-optimality sensitivity analysis provides the correct parameter dependence for infinitesimally small changes to the parameter values, i.e., only local information. In contrast, parametric optimization provides a correct estimate of the influence of parameters over a whole range, i.e., global sensitivity information.

The resource allocation problem described here is qualitative rather than quantitative such as the formulations in [161, 190, 263]. In these references the authors consider chemical products such as pharmaceuticals with quantifiable resource needs for development and testing. The focus of this thesis is on identifying the components of a microfabricated electrochemical process that should be developed. Attempts to estimate the resources needed to advance a given technology are conceivable, but a proper quantification is not always possible. We instead propose that the answer provided by the parametric optimization algorithm has to be evaluated by the decision maker, e.g., the program manager, who can balance the tradeoff between expenditure of resources and potential of improving the systems performance. If the effect of resource allocation can be quantified, the unknown parameters can be replaced with their function of the resources, and the resources can be added to the variable list. In that case, the resulting program would directly furnish the optimal resource allocation.

3.3 MILP Optimality Range

Recall that parametric optimization algorithms identify a set of qualitatively invariant solutions and their respective optimality regions. In parametric mixed-integer linear programs a qualitatively invariant solution is an integer realization along with a basis for this integer realization. Suppose that such a solution is given by $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$ and its feasibility has been established for a subset of the parameter range $\mathbf{p} \in P' \subset P$. Unlike linear programs, no explicit optimality conditions are available and therefore an interesting question is whether or not a given pair $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$ is optimal over $\mathbf{p} \in P' \subset P$.

Note that in general, obtaining P' and the function $\bar{\mathbf{x}} : P' \rightarrow \mathbb{R}^{n_x}$ is a nontrivial

task, see also Section 3.5.2. Assuming though that the data are given as rational functions of the parameter (quotients of multivariate polynomials), it can be shown that as long as the basis matrix considered remains nonsingular, the qualitatively invariant solution $\bar{\mathbf{x}}$ is a smooth rational function of \mathbf{p} . In the cost vector case, the feasible set does not depend on the parameter and $\bar{\mathbf{x}}$ is constant. In the right hand side case assuming an affine dependence of the right hand side vector on the parameters, it can be shown that $\bar{\mathbf{x}}$ is an affine function of the parameters \mathbf{p} .

Pertsinidis et al. [224, 225] proposed formulating the determination of the optimality range as a new optimization problem for which the parameters are added to the variable list and called this “sensitivity analysis”. To avoid confusion with the parametric dependence of the optimal solution we will refer to this problem as finding the optimality region. Pertsinidis et al. considered the right hand side case with one parameter and a uniqueness assumption. We consider and discuss generalizations of this formulation.

Starting from (3.1) the parameters are added to the list of variables

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}, \mathbf{p}} (\mathbf{c}^x(\mathbf{p}))^T \mathbf{x} + (\mathbf{c}^y(\mathbf{p}))^T \mathbf{y} - \bar{f}(\mathbf{p}) \\
& \text{s.t. } \mathbf{A}^{1x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{1y}(\mathbf{p}) \mathbf{y} = \mathbf{b}^1(\mathbf{p}) \\
& \quad \mathbf{A}^{2x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{2y}(\mathbf{p}) \mathbf{y} \leq \mathbf{b}^2(\mathbf{p}) \\
& \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{y} \in \{0, 1\}^{n_y} \\
& \quad \mathbf{p} \in P',
\end{aligned} \tag{3.3}$$

where $\bar{f}(\mathbf{p}) \equiv (\mathbf{c}^x(\mathbf{p}))^T \bar{\mathbf{x}}(\mathbf{p}) + (\mathbf{c}^y(\mathbf{p}))^T \bar{\mathbf{y}}$. If the optimal value of (3.3) is less than zero then $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$ is not optimal for the parameter value \mathbf{p}^* furnished by (3.1). The solution point $(\mathbf{x}^*, \mathbf{y}^*)$ is optimal for \mathbf{p}^* . If the optimal value of (3.3) is greater or equal to zero, $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$ is an optimal solution to the MILP (3.1) for all $\mathbf{p} \in P'$. Note that by feasibility of $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$, the optimal value of (3.3) cannot be greater than zero, unless some integer cuts are added.

An alternative formulation is to also include a metric for the parameter $g(\mathbf{p})$, e.g.,

the distance from a given parameter value and solve

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}, \mathbf{p}} g(\mathbf{p}) \\
& \text{s.t. } (\mathbf{c}^x(\mathbf{p}))^\top \mathbf{x} + (\mathbf{c}^y(\mathbf{p}))^\top \mathbf{y} - \bar{f}(\mathbf{p}) \leq -\varepsilon \\
& \quad \mathbf{A}^{1x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{1y}(\mathbf{p}) \mathbf{y} = \mathbf{b}^1(\mathbf{p}) \\
& \quad \mathbf{A}^{2x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{2y}(\mathbf{p}) \mathbf{y} \leq \mathbf{b}^2(\mathbf{p}) \\
& \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{y} \in \{0, 1\}^{n_y}, \\
& \quad \mathbf{p} \in P',
\end{aligned} \tag{3.4}$$

where ε is a prespecified optimality tolerance. If (3.4) is infeasible, $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$ is an optimal solution for the MILP (3.1) for all $\mathbf{p} \in P'$. Otherwise $(\bar{\mathbf{x}}(\mathbf{p}), \bar{\mathbf{y}})$ is not optimal in (3.1) for $\mathbf{p} = \mathbf{p}^*$ (the parameter value furnished by (3.4)). Note that the tolerance ε introduces an overestimation of the optimality range. Note also that the solution point $(\mathbf{x}^*, \mathbf{y}^*)$ furnished by (3.4) need not be optimal in (3.1) for \mathbf{p}^* .

If the LP-optimality range for a fixed integer realization has been established for $\mathbf{p} \in P'$, then an integer cut $\mathbf{y} \neq \mathbf{y}^*$ can be added to either (3.3) or (3.4), likely accelerating the convergence. Such a cut can be formulated as in Balas and Jeroslow [33]

$$\sum_{i \in O} y_i - \sum_{i \in Z} y_i \leq |O| - 1,$$

where O, Z are index sets for the elements of \mathbf{y}^* with values of one and zero, respectively.

3.3.1 Range of Infeasibility

A similar question to the optimality range considered in the previous subsection is the “range of infeasibility”. In linear programs, the introduction of surplus variables allows the infeasibility to be treated similarly to a feasible basis, see Section 3.5.2. This is not possible for MILPs, because parameter variation may render some integer

realization feasible. One possibility to obtain the infeasibility range is to consider a variant of (3.4) where the bound on the optimal solution is dropped

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}, \mathbf{p}} g(\mathbf{p}) \\
& \text{s.t. } \mathbf{A}^{1x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{1y}(\mathbf{p}) \mathbf{y} = \mathbf{b}^1(\mathbf{p}) \\
& \quad \mathbf{A}^{2x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{2y}(\mathbf{p}) \mathbf{y} \leq \mathbf{b}^2(\mathbf{p}) \\
& \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x} \quad \mathbf{y} \in \{0, 1\}^{n_y}, \\
& \quad \mathbf{p} \in P'.
\end{aligned} \tag{3.5}$$

If (3.5) is feasible for some parameter value \mathbf{p}^* then so is (3.1). Alternatively, to check that a program is infeasible over a parameter range a variant of (3.3) can be used

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}, \mathbf{p}} (\mathbf{c}^x(\mathbf{p}))^T \mathbf{x} + (\mathbf{c}^y(\mathbf{p}))^T \mathbf{y} \\
& \text{s.t. } \mathbf{A}^{1x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{1y}(\mathbf{p}) \mathbf{y} = \mathbf{b}^1(\mathbf{p}) \\
& \quad \mathbf{A}^{2x}(\mathbf{p}) \mathbf{x} + \mathbf{A}^{2y}(\mathbf{p}) \mathbf{y} \leq \mathbf{b}^2(\mathbf{p}) \\
& \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{y} \in \{0, 1\}^{n_y} \\
& \quad \mathbf{p} \in P'.
\end{aligned} \tag{3.6}$$

If (3.6) is infeasible, so is (3.1) for all $\mathbf{p} \in P'$.

3.3.2 Classification of Optimality Region Formulations

In general, both formulations (3.3) and (3.4) contain nonconvex nonlinear functions and are characterized as separable nonconvex mixed-integer nonlinear programs (MINLP). The global solution of nonconvex MINLPs is very computationally expensive but there exist algorithms [14, 169, 267] and at least one commercial program [247] that can do this with optimality guarantees. The specific structure of the MINLP considered could be exploited by specialized algorithms. The nonlinearity originates

from $\bar{f}(\mathbf{p})$ as well as from the products between the parameter dependent data and the parameters. In that sense, the parameters \mathbf{p} can be considered as complicating variables and typically there is a small number of parameters ($n_p \ll n_x$ and $n_p \ll n_y$).

Optimality Region as MILP

Pertsinidis et al. [224, 225] considered the right hand side case with one parameter and assumed an affine dependence of the right hand side vector $\mathbf{b}_1 : P' \rightarrow \mathbb{R}^m$ on the parameter. In that case $\bar{f} : P' \rightarrow \mathbb{R}$ is affine and both (3.3) and (3.4) are mixed-integer linear programs. This result can be generalized to the case that also the cost vector and matrix corresponding to the integer variables $\mathbf{c}^y : P' \rightarrow \mathbb{R}^{n_p}$, $\mathbf{A}^y : P' \rightarrow \mathbb{R}^{m \times n_y}$ are affine functions of the parameters. For simplicity here we use standard form

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{y}} \quad & (\mathbf{c}^x)^\top \mathbf{x} + (\mathbf{c}^{y,0} + \sum_{i=1}^{n_p} \mathbf{c}^{y,i} p_i)^\top \mathbf{y} \\
\text{s.t.} \quad & \mathbf{A}^x \mathbf{x} + (\mathbf{A}^{y,0} + \sum_{i=1}^{n_p} \mathbf{A}^{y,i} p_i) \mathbf{y} = \mathbf{b}^0 + \sum_{i=1}^{n_p} \mathbf{b}^i p_i \\
& \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\
& \mathbf{y} \in \{0, 1\}^{n_y},
\end{aligned} \tag{3.7}$$

where $\mathbf{c}^x \in \mathbb{R}^{n_x}$, $\mathbf{A}^x \in \mathbb{R}^{m \times n_x}$, and for $i = 0, \dots, n_p$, $\mathbf{c}^{y,i} \in \mathbb{R}^{n_y}$, $\mathbf{b}^i \in \mathbb{R}^m$, $\mathbf{A}^{y,i} \in \mathbb{R}^{m \times n_y}$. Formulation (3.3) becomes

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{y}, \mathbf{p}} \quad & (\mathbf{c}^x)^\top \mathbf{x} + (\mathbf{c}^{y,0} + \sum_{i=1}^{n_p} \mathbf{c}^{y,i} p_i)^\top \mathbf{y} - \bar{f}(\mathbf{p}) \\
\text{s.t.} \quad & \mathbf{A}^x \mathbf{x} + (\mathbf{A}^{y,0} + \sum_{i=1}^{n_p} \mathbf{A}^{y,i} p_i) \mathbf{y} = \mathbf{b}^0 + \sum_{i=1}^{n_p} \mathbf{b}^i p_i \\
& \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\
& \mathbf{y} \in \{0, 1\}^{n_y}, \quad \mathbf{p} \in P'.
\end{aligned} \tag{3.8}$$

It can be shown that $\bar{f} : P' \rightarrow \mathbb{R}$ is affine, and the only nonlinearities in (3.3) are the bilinear products between the parameters \mathbf{p} and the binary variables \mathbf{y} . These products can be reformulated by the exact linearizations of Glover and Woolsey [124,

125, 123] to obtain a MILP of increased size. Assuming that $P' \subset [0, 1]^{n_p}$ each bilinear term $p_i y_j$ is replaced by a new real valued variable $w_j^i \in [0, 1]$ and three new linear inequality constraints are introduced, that enforce $w_i = p_i$ for $y_j = 1$ and $w_i = 0$ for $y_j = 0$. For simplicity we assume that all the possible products exist (worse-case scenario) and obtain

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}, \mathbf{p}, \mathbf{w}^i} (\mathbf{c}^x)^\top \mathbf{x} + \mathbf{c}^{y,0} \mathbf{y} + \sum_{i=1}^{n_p} \sum_{j=1}^{n_y} c_j^{y,i} w_j^i - \bar{f}(\mathbf{p}) \\
\text{s.t. } & \mathbf{A}^x \mathbf{x} + \mathbf{A}^{y,0} \mathbf{y} + \sum_{i=1}^{n_p} \sum_{j=1}^{n_y} \mathbf{A}_j^{y,i} w_j^i = \mathbf{b}^0 + \sum_{i=1}^{n_p} \mathbf{b}^i p_i \quad (3.9) \\
& w_j^i \leq y_j, \quad \forall i = 1, \dots, n_p \quad \forall j = 1, \dots, n_y \\
& w_j^i \leq p_i, \quad \forall i = 1, \dots, n_p \quad \forall j = 1, \dots, n_y \\
& p_i + y_j - 1 \leq w_j^i, \quad \forall i = 1, \dots, n_p \quad \forall j = 1, \dots, n_y \\
& \mathbf{w}^i \in [0, 1]^{n_y}, \quad \forall i = 1, \dots, n_p \\
& \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\
& \mathbf{y} \in \{0, 1\}^{n_y}, \quad \mathbf{p} \in P',
\end{aligned}$$

where $\mathbf{A}_j^{y,i}$ denotes the j th column of $\mathbf{A}^{y,i}$. An alternative is to use the transformation on the original program (3.7) and convert this to a right hand-side parametric program.

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}, \mathbf{w}^i} (\mathbf{c}^x)^\top \mathbf{x} + \mathbf{c}^{y,0} \mathbf{y} + \sum_{i=1}^{n_p} \sum_{j=1}^{n_y} c_j^{y,i} w_j^i \\
\text{s.t. } & \mathbf{A}^x \mathbf{x} + \mathbf{A}^{y,0} \mathbf{y} + \sum_{i=1}^{n_p} \sum_{j=1}^{n_y} \mathbf{A}_j^{y,i} w_j^i = \mathbf{b}^0 + \sum_{i=1}^{n_p} \mathbf{b}^i p_i \quad (3.10) \\
& w_j^i - y_j \leq 0, \quad \forall i = 1, \dots, n_p \quad \forall j = 1, \dots, n_y \\
& w_j^i \leq p_i, \quad \forall i = 1, \dots, n_p \quad \forall j = 1, \dots, n_y \\
& y_j - w_j^i \leq p_i + 1, \quad \forall i = 1, \dots, n_p \quad \forall j = 1, \dots, n_y \\
& \mathbf{w}^i \in [0, 1]^{n_y}, \quad \forall i = 1, \dots, n_p \\
& \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\
& \mathbf{y} \in \{0, 1\}^{n_y}.
\end{aligned}$$

¹Otherwise, a similar reformulation is performed.

Applicability of MINLP Algorithms

We briefly discuss the applicability of MINLP algorithms to the optimality region formulations (3.3) and (3.4) assuming an affine dependence on the parameters \mathbf{p} .

Branch and Reduce (Tawarmalani and Sahinidis [267])

The only limitation mentioned in [267] is that the functions are “factorable, i.e., functions that are recursive sums and products of univariate functions”. As the formulated MINLP contains only bilinear terms and a univariate polynomial or rational function the algorithms are indeed applicable.

Outer Approximation (Kesavan et al. [169])

The proposed MINLP formulation falls under problem P in [169] if one reformulates the bilinear terms between the parameter p and the integer variables (for the nonzero entries of \mathbf{A}^{1y} , \mathbf{A}^{2y} , \mathbf{c}^y). Note though that his reformulation is not necessary [170].

The constraints g_2 are linear and the constraints g_1 contain bilinear terms xp . The functions f and g_2 are linear and g_1 has bilinear terms and they are all twice continuously differentiable. The functions L_1 and L_2 can be formulated as twice continuously differentiable. For the construction of the lower bounding MINLP P1 the convex relaxations of bilinear terms can be formulated as a set of four linear inequalities for each bilinear term. The rational function $\bar{f}(\mathbf{p})$ can either be reformulated with a set of linear inequalities or as a strictly convex function $\bar{g}(\mathbf{p})$. In the former case a constraint qualification holds because all constraints are linear. In the latter case, the KKT conditions may not be necessary for a local minimum. The following is a counterexample

$$\begin{aligned} \min_{x,p} p \\ \text{s.t. } -x + p^2 &\leq 0 \\ x &\leq 0 \\ -x &\leq 0. \end{aligned}$$

The only feasible point is $x^* = p^* = 0$ and therefore it is the unique local and

global minimum. The gradient of the constraints evaluated at this point are given by $(0, 1)^T$ and the gradient of the objective function is given by $(1, 0)^T$ and therefore the optimal point does not satisfy the KKT conditions, or the KKT conditions are not necessary. Essentially the constraint qualifications is violated when at the minimum the constraint on the objective function is active and its gradient is linearly dependent on the other active constraints. Otherwise all active constraints are linear and the constraint qualification holds.

SMIN- α BB (Adjiman et al. [14])

The algorithm described in [14] is applicable. In particular for the objective function we have $f(\mathbf{x}) = p$, $A_f = 0$, $c_f^T = 0$. The equality constraints have only bilinear terms and therefore $h_i(\mathbf{x}) = 0$, $i = 1, \dots, p$. The same holds for the inequality constraints except for the one involving the constraint on the objective function ($g_i(\mathbf{x}) = 0$, $i = 1, \dots, m - 1$) and $g_m(\mathbf{x}) = \bar{f}(p)$ which is a univariate rational or polynomial function defined over the whole range of the variable p and as such twice continuously differentiable.

3.4 Multiparametric Cost Vector Case

In this section we consider the cost vector case with affine dependence of the coefficients on the parameters. For simplicity we describe the algorithms for a MILP in standard form:

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}} (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^T \mathbf{x} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^T \mathbf{y} \\
& \text{s.t. } \mathbf{A}^x \mathbf{x} + \mathbf{A}^y \mathbf{y} = \mathbf{b} \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\
& \quad \mathbf{y} \in \{0, 1\}^{n_y},
\end{aligned} \tag{3.11}$$

where $\mathbf{p} \in [0, 1]^{n_p}$, $\mathbf{c}^x \in \mathbb{R}^{n_x}$, $\mathbf{c}^y \in \mathbb{R}^{n_y}$, $\mathbf{C}^x \in \mathbb{R}^{n_x \times n_p}$, $\mathbf{C}^y \in \mathbb{R}^{n_y \times n_p}$, $\mathbf{A}^x \in \mathbb{R}^{m \times n_x}$, $\mathbf{A}^y \in \mathbb{R}^{m \times n_y}$, $\mathbf{b} \in \mathbb{R}^m$.

The cost vector case is much simpler than the general case both from a theoretical

and algorithmic point of view, because the feasible set is not affected by changes in the parameter value [119]. Together with the assumption of affine parameter dependence, the parameter range can be divided in a finite collection of polyhedra, for each of which a point is optimal. The notion of a qualitatively invariant solution is thus not needed. Throughout this section we assume that every instance considered satisfies two properties:

Assumption 3.1 (Feasible set nonempty). The feasible set is non-empty.

Assumption 3.2 (Optimal solution bounded). For all $\mathbf{p} \in [0, 1]^{n_p}$ the optimal solution is finite.

Feasibility is assumed for the sake of simplicity. Since the feasible set does not depend on the parameter, infeasibility can be detected by solving the MILP at any fixed parameter value. On the other hand allowing unbounded solutions would require some changes for the algorithms presented. Furthermore, we make a restriction on the MILP solvers used to ensure that we obtain a finite number of solutions.

Assumption 3.3 (MILP solvers). There exists a MILP solver which for any fixed value of the parameters furnishes an optimal solution at a vertex of the feasibility region.

3.4.1 Theoretical Properties

Most properties described here are well-known, see e.g., [119, 139], but are nevertheless included because of their relevance to the algorithms presented and because in the literature proofs are mostly provided for a single parameter.

Note that as the interior of a set P^k we denote the set of interior points, i.e.,

$$\text{int}(P^k) = \{\mathbf{p} \in P^k : N_\delta(\mathbf{p}) \subset P^k\},$$

for some open neighborhood $N_\delta(\mathbf{p})$.

We first prove the following lemma

Lemma 3.1. *Let $P^l \subset [0, 1]^{n_p}$ be a closed set with an empty interior and K be the index set of a finite collection of closed sets $P^k \subset [0, 1]^{n_p}$ and denote $P^K = \bigcup_{k \in K} P^k$.*

Then $[0, 1]^{n_p} \subset P^l \cup P^K$ implies $[0, 1]^{n_p} \subset P^K$.

Proof. Proof by contraposition

Let $[0, 1]^{n_p} \not\subset P^K$. Therefore there exists $\bar{\mathbf{p}} \in [0, 1]^{n_p}$, such that $\bar{\mathbf{p}} \notin P^k$ for any $k \in K$. Since P^k are closed, $\bar{\mathbf{p}}$ is not a limit point of any P^k . Therefore for each $k \in K$, there exists $\delta^k > 0$ such that the neighborhood $N_{\delta^k}(\bar{\mathbf{p}})$ gives

$$N_{\delta^k}(\bar{\mathbf{p}}) \cap P^k = \emptyset.$$

Take $\delta = \min_{k \in K} \delta^k$ and since K is finite $\delta > 0$. Consider the corresponding neighborhood $N_\delta(\bar{\mathbf{p}})$. It is therefore possible to find a set $N' \subset N_\delta(\bar{\mathbf{p}}) \cap [0, 1]^{n_p}$ with a nonempty interior. Since N' has a nonempty interior it cannot be a subset of P^l . Therefore there exists $\mathbf{p} \in N'$ such that $\mathbf{p} \notin P^l$. Since $N' \subset [0, 1]^{n_p}$ we have $\mathbf{p} \in [0, 1]^{n_p}$ and since $N' \subset N_\delta(\bar{\mathbf{p}})$ we have $\mathbf{p} \notin P^K$. Therefore we obtain $[0, 1]^{n_p} \not\subset P^K \cup P^l$. \square

Lemma 3.2. *The parameter range can be divided into a finite collection with index set K of bounded polyhedra P^k and associated points $(\mathbf{x}^k, \mathbf{y}^k)$, such that*

1. $[0, 1]^{n_p} = \bigcup_{k \in K} P^k$ and for any $k \in K$, the point $(\mathbf{x}^k, \mathbf{y}^k)$ is an optimal solution² of (3.11) for all $\mathbf{p} \in P^k$.
2. For any $k \in K$, the polyhedron P^k has a nonempty interior
3. For any $k_1 \in K, k_2 \in K$, the interiors do not overlap $\text{int}(P^{k_1}) \cap \text{int}(P^{k_2}) = \emptyset$

Proof. A direct consequence of Assumption 3.2 is that an optimal solution exists for all \mathbf{p} . Moreover for each parameter value, there exists an optimal integer realization along with an optimal basis of the linear program that results by fixing the binary variables. Therefore, it suffices to consider the vertices as candidate optimal solutions and there is a finite number of vertices $\mathbf{x}^i, \mathbf{y}^i, i = 1, \dots, k^{ver}$ with $k^{ver} \leq 2^{n_y} \frac{n_x!}{m!(n_x-m)!}$.

1. Consider any feasible point $\bar{\mathbf{x}}, \bar{\mathbf{y}}$, irrespectively of whether it is a vertex or not. The optimality region of this point is defined by the following set of linear

²Not necessarily the unique optimal solution though.

inequality constraints

$$\begin{aligned} (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \bar{\mathbf{x}} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \bar{\mathbf{y}} &\leq (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^i + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^i, \quad i = 1, \dots, k^{ver} \\ \mathbf{0} &\leq \mathbf{p} \leq \mathbf{1} \end{aligned}$$

which implies that the optimal region is a bounded polyhedron. Note that for some feasible points $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ the above inequalities define an empty set, for other points a polyhedron in the parameter space whose interior is empty (single point, line, polygon, etc.) and for other points a polyhedron in the parameter space with a nonempty interior.

Recall now that it suffices to consider vertices of the feasible set as candidate optimal solutions. Since there is a finite number of vertices, the parameter range $[0, 1]^{n_p}$ can be divided into a finite collection of closed bounded polyhedra (with index set K) such that $[0, 1]^{n_p} = \bigcup_{k \in K} P^k$.

2. Consider now that a polyhedron with index $\bar{k} \in K$ has an empty interior. Since the polyhedra are closed and $\bigcup_{k \in K} P^k = [0, 1]^{n_p}$, by Lemma 3.1 we have $\bigcup_{k \in K, k \neq \bar{k}} P^k = [0, 1]^{n_p}$ or $P^{\bar{k}}$ can be excluded from the collection.
3. Finally consider two polyhedra $k_1 \in K, k_2 \in K$ with overlapping interiors $\text{int}(P^{k_1}) \cap \text{int}(P^{k_2}) \neq \emptyset$. Let

$$f^{k_1}(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^{k_1} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^{k_1}$$

and

$$f^{k_2}(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^{k_2} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^{k_2}.$$

Take any point $\bar{\mathbf{p}} \in \text{int}(P^{k_1}) \cap \text{int}(P^{k_2})$. There exists $\delta > 0$, such that

$$\mathbf{p} \in P^{k_1} \cap P^{k_2}, \quad \forall \|\mathbf{p} - \bar{\mathbf{p}}\| < \delta.$$

For all $\mathbf{p} \in P^{k_1} \cap P^{k_2}$ both $(\mathbf{x}^{k_1}, \mathbf{y}^{k_1})$ and $(\mathbf{x}^{k_2}, \mathbf{y}^{k_2})$ are optimal solutions of (3.11) and therefore $f^{k_1}(\mathbf{p}) = f^{k_2}(\mathbf{p})$. Since f^{k_1}, f^{k_2} are linear, it follows that

$f^{k_1}(\mathbf{p}) = f^{k_2}(\mathbf{p})$ for all $\mathbf{p} \in [0, 1]^{n_p}$. This can be easily verified, e.g., by taking points along the unit directions $\hat{p} = \bar{p} + \delta/2e_j$, from which it follows that $\frac{\partial(f^{k_1}-f^{k_2})}{\partial p_j} = 0$. Therefore the two polyhedra are identical and one of the two can be eliminated from the collection.

□

Lemma 3.3. *The optimal cost function is piecewise-affine, continuous, and concave over the parameter space.*

Proof. From the perspective of the optimal cost function it suffices to consider the vertices as candidate optimal solutions and recall that there is a finite number of vertices $\mathbf{x}^i, \mathbf{y}^i, i = 1, \dots, k^{ver}$ with $k^{ver} \leq 2^{n_y} \frac{n!}{m!(n-m)!}$. The optimal cost function is therefore given by

$$\min_{i=1, \dots, k^{ver}} (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^T \mathbf{x}^i + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^T \mathbf{y}^i,$$

and therefore is piecewise-affine, continuous and concave [42, p. 217]. Note also that for each of the optimality regions the optimal cost function is affine. □

Often large-scale MILPs are not solved to optimality, but rather to ε -optimality, i.e., the solver furnishes a feasible point $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ along with an ε -optimality guarantee

$$(\mathbf{c}^x)^T \bar{\mathbf{x}} + (\mathbf{c}^y)^T \bar{\mathbf{y}} \leq (\mathbf{c}^x)^T \mathbf{x} + (\mathbf{c}^y)^T \mathbf{y} + \varepsilon, \quad \forall (\mathbf{x}, \mathbf{y}) \text{ feasible.}$$

Therefore the following lemma is of importance.

Lemma 3.4. *Suppose that the parameter set has been divided into a finite collection of polyhedra P^k and associated points $(\mathbf{x}^k, \mathbf{y}^k)$. Suppose further that for each polyhedron k the given point $(\mathbf{x}^k, \mathbf{y}^k)$ is ε -optimal at each vertex of the polyhedron k . Then $(\mathbf{x}^k, \mathbf{y}^k)$ is ε -optimal on the polyhedron k .*

Proof. Note first that since $(\mathbf{x}^k, \mathbf{y}^k)$ is ε -optimal at some parameter value it is

feasible for all parameter values. Let

$$f^k(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^k + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^k.$$

Let \mathbf{p}^j , $j = 1, \dots, N_k$ be the vertices of polyhedron k . Let $\bar{\mathbf{p}}$ be an arbitrary point in polyhedron k , and as such a convex combination of the vertices

$$\bar{\mathbf{p}} = \sum_{j=1}^{N_k} \lambda_j \mathbf{p}^j, \quad \sum_{j=1}^{N_k} \lambda_j = 1, \quad \lambda_j \geq 0, \quad j = 1, \dots, N_k.$$

Let $\bar{\mathbf{x}}$, $\bar{\mathbf{y}}$ be any feasible solution and $\bar{f}(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \bar{\mathbf{x}} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \bar{\mathbf{y}}$ the corresponding objective value function. The assumption of ε -optimality at the vertices implies

$$f^k(\mathbf{p}^j) \leq \bar{f}(\mathbf{p}^j) + \varepsilon, \quad j = 1, \dots, N_k.$$

Since f^k and \bar{f} are affine in the parameters \mathbf{p} and $\sum_{j=1}^{N_k} \lambda_j = 1$ we obtain

$$\begin{aligned} f^k(\bar{\mathbf{p}}) &= f^k\left(\sum_{j=1}^{N_k} \lambda_j \mathbf{p}^j\right) = \sum_{j=1}^{N_k} \lambda_j f^k(\mathbf{p}^j) \\ \bar{f}(\bar{\mathbf{p}}) &= \bar{f}\left(\sum_{j=1}^{N_k} \lambda_j \mathbf{p}^j\right) = \sum_{j=1}^{N_k} \lambda_j \bar{f}(\mathbf{p}^j), \end{aligned}$$

and therefore

$$\begin{aligned} f^k(\bar{\mathbf{p}}) - \bar{f}(\bar{\mathbf{p}}) &= \sum_{j=1}^{N_k} \lambda_j f^k(\mathbf{p}^j) - \sum_{j=1}^{N_k} \lambda_j \bar{f}(\mathbf{p}^j) \\ &= \sum_{j=1}^{N_k} \lambda_j (f^k(\mathbf{p}^j) - \bar{f}(\mathbf{p}^j)) \\ &\leq \sum_{j=1}^{N_k} \lambda_j \varepsilon \\ &= \varepsilon \sum_{j=1}^{N_k} \lambda_j \\ &= \varepsilon. \end{aligned}$$

Since the above inequality holds for any $\bar{\mathbf{p}}$ in the polyhedron and any feasible $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ ε -optimality is established. \square

3.4.2 Intersection-Based Algorithm for a Single Parameter

In this subsection, we briefly repeat a well-known algorithm [101, 162, 164] for the case of one parameter ($p \in [0, 1]$). This will facilitate the presentation of the multi-parametric algorithm.

The intersection-based algorithm utilizes the concavity and continuity of the optimal objective function value and the convexity of the optimality regions. It successively identifies optimal solutions and their optimality regions (intervals) by solving MILPs at fixed parameter values. The parameter values are chosen based on assumed optimality intervals; if the optimal solution value of the MILP at a given parameter value is equal to the assumed optimal solution, the optimality region is confirmed, otherwise a new optimal solution is obtained and added to the list of existing solutions. At any iteration, the candidate optimal solutions provide both a lower and an upper bound to the optimal objective value for all $p \in [0, 1]$, see below.

The algorithm uses a set R of optimal solutions along with their optimality region. Each element R_k of the set R is a triplet $\left(p^{R_k}, (\mathbf{x}^{R_k}, \mathbf{y}^{R_k}), (f_0^{R_k}, f_1^{R_k})\right)$ composed of a parameter value p^{R_k} , a solution of the MILP (3.11) $(\mathbf{x}^{R_k}, \mathbf{y}^{R_k})$, and the coefficients $f_0^{R_k}, f_1^{R_k}$ of the objective value function $f^{R_k}(p) = f_0^{R_k} + f_1^{R_k} p$. The elements of R are ordered by the parameter values p^{R_k} . The number of different optimal solutions in the parameter range is denoted by n_s , while p_t is a temporary parameter value and j is used as an iteration counter. $|R|$ denotes the cardinality of R . The solution of the MILP (3.11) at a parameter value p_t is denoted by $(\mathbf{x}^*, \mathbf{y}^*)$ and the corresponding optimal cost with $f^*(p)$, calculated as $f^*(p) = (\mathbf{c}^x + \mathbf{C}^x p)^\top \mathbf{x}^* + (\mathbf{c}^y + \mathbf{C}^y p)^\top \mathbf{y}^*$.

Algorithm 3.1. Intersection Algorithm for Cost Vector Parametric MILP, $n_p = 1$

1. **(Initialization)**

Set $n_s = 0$, $j = 1$ and $R = \emptyset$.

Solve MILP (3.11) at $p = 0$.

Create a new triplet with $p = 0$, $\mathbf{x} = \mathbf{x}^*$, $\mathbf{y} = \mathbf{y}^*$ and $f_0 = (\mathbf{c}^x)^\top \mathbf{x}^* + (\mathbf{c}^y)^\top \mathbf{y}^*$,
 $f_1 = (\mathbf{C}^x)^\top \mathbf{x}^* + (\mathbf{C}^y)^\top \mathbf{y}^*$
 Insert the triplet in R .

2. REPEAT

(a) **(Get next parameter value)**

IF $j = 1$ **THEN**

- Set $p_t = 1$.

ELSE

- Set $p_t = -\frac{f_0^{R_{n_s+1}} - f_0^{R_{n_s+2}}}{f_1^{R_{n_s+1}} - f_1^{R_{n_s+2}}}$.

(b) **(Solution at new parameter value)**

Solve MILP (3.11) at $p = p_t$.

IF $f^* < f_0^{R_{n_s+1}} + f_1^{R_{n_s+1}} p_t$

- Create a new triplet with $p = p_t$, $\mathbf{x} = \mathbf{x}^*$, $\mathbf{y} = \mathbf{y}^*$,
 and $f_0 = (\mathbf{c}^x)^\top \mathbf{x}^* + (\mathbf{c}^y)^\top \mathbf{y}^*$, $f_1 = (\mathbf{C}^x)^\top \mathbf{x}^* + (\mathbf{C}^y)^\top \mathbf{y}^*$.

Insert the triplet in R (ordered according to the parameter value).

ELSE IF $n_s < |R|$

- Set $n_s = n_s + 1$.
- Set $p^{R_{n_s+1}} = p_t$.
- **IF** $n_s = |R| - 1$ **THEN** set $n_s = |R|$.

(c) Set $j = j + 1$.

UNTIL $n_s = |R|$.

At termination, the set R contains the optimal solutions along with their optimality regions, i.e., for $k = 1, \dots, n_s$ the point $(\mathbf{x}^{R_k}, \mathbf{y}^{R_k})$ is optimal for $p \in [p^{R_k}, p^{R_{k+1}}]$, with $p^{R_{|R|+1}} = 1$. If there is one optimality region, two MILP calls are needed. For $n_s > 1$ the last optimality interval requires the solution of one MILP and all others two MILP calls, i.e., one to identify the solution point and one to verify its optimality

interval. Therefore for $n_s > 1$ the algorithm requires $2n_s - 1$ calls to a MILP solver [162].

During the iterations, for all $k = 1, \dots, n_s$ the k -th element of set R contains the optimal solution for $p \in [p^{R_k}, p^{R_{k+1}}]$. For $k = n_s + 1, \dots, |R|$, the k -th element contains the optimal solution for $p = p^{R_k}$. After the second iteration, by concavity, an upper bound of the objective function at any parameter value \bar{p} can be calculated as the minimum of the two adjacent optimal solutions

$$f^*(\bar{p}) \leq \min(f^{R_k}(\bar{p}), f^{R_{k+1}}(\bar{p})) \text{ for } k : p^{R_k} \leq \bar{p} \leq p^{R_{k+1}}.$$

Also by concavity the secant can be used as a lower bound

$$f^*(\bar{p}) \geq f^{R_k}(p^{R_k}) + (f^{R_{k+1}}(p^{R_{k+1}}) - f^{R_k}(p^{R_k})) \frac{\bar{p} - p^{R_k}}{p^{R_{k+1}} - p^{R_k}} \text{ for } k : p^{R_k} \leq \bar{p} \leq p^{R_{k+1}}.$$

These upper and lower bounds can be used for acceleration of the MILP solution.

Illustrative Example

To demonstrate how Algorithm 3.1 operates, consider a simple (IP)

$$\begin{aligned} \min (\mathbf{c} + \mathbf{C}p)^T \mathbf{y} \\ \text{s.t. } \sum_{i=1}^4 y_i &= 1 \\ \mathbf{y} &\in \{0, 1\}^4, \end{aligned} \tag{3.12}$$

where $\mathbf{c} = (0, 0.1, 0.5, 0.9)$ and $\mathbf{C} = (1, 0.6, -0.3, -0.9)$. The optimality regions of the four integer realizations are

- $(1, 0, 0, 0)$ is optimal for $p \in [0, 1/4]$
- $(0, 1, 0, 0)$ is optimal for $p \in [1/4, 4/9]$
- $(0, 0, 1, 0)$ is optimal for $p \in [4/9, 2/3]$
- $(0, 0, 0, 1)$ is optimal for $p \in [2/3, 1]$.

The steps of Algorithm 3.1 are summarized below, and graphically illustrated in Figure 3-1, where each optimal solution function is plotted with a different color. For the numbering of the solutions, note the reordering as new solutions are obtained.

Initialization

Solve MILP (3.12) for $p = 0$ and obtain $\mathbf{y}^* = (1, 0, 0, 0)$.

Set $f^*(p) = p$.

Insert the triplet $p = 0, \mathbf{y} = \mathbf{y}^*, f(p) = f^*$ into set R .

Main loop

Solve MILP (3.12) for $p = 1$ and obtain $\mathbf{y}^* = (0, 0, 0, 1)$ and $f^* = 0$.

Since $0 = f^* < f^{R_1}(1) = 1$

Set $f^* = 0.9 - 0.9p$.

Insert the triplet $p = 1, \mathbf{y} = \mathbf{y}^*, f(p) = f^*$ into set R .

$p = f^{R_1}(p) = f^{R_2}(p) = 0.9 - 0.9p$ gives $p_t \approx 0.473$

Solve MILP (3.12) for $p = p_t$ and obtain $\mathbf{y}^* = (0, 0, 1, 0)$ and $f^* \approx 0.358$.

Since $0.358 = f^* < f^{R_1}(0.473) = 0.474$

Set $f^* = 0.5 - 0.3p$.

Insert the triplet $p = 0.5, \mathbf{y} = \mathbf{y}^*, f(p) = f^*(p)$ into set R .

$p = f^{R_1}(p) = f^{R_2}(p) = 0.5 - 0.3p$ gives $p_t \approx 0.384$

Solve MILP (3.12) for $p = p_t$ and obtain $\mathbf{y}^* = (0, 1, 0, 0)$ and $f^* = 0.333$.

Since $0.331 = f^* < f^{R_1}(0.384) = 0.385$

Set $f^* = 0.1 + 0.6p$.

Insert the triplet $p = 0.384, \mathbf{y} = \mathbf{y}^*, f(p) = f^*(p)$ into set R .

$p = f^{R_1}(p) = f^{R_2}(p) = 0.1 + 0.6p$ gives $p_t = 0.25$

Solve MILP (3.12) for $p = p_t$ and obtain $f^* = 0.25$.

Since $0.25 = f^* = f^{R_1}(0.25) = 0.25$

the optimality region of $f^1 = p$ is confirmed.

Set $p^{R_2} = 0.25$. Set $n_s = 1$.

$0.1 + 0.6p = f^{R_2}(p) = f^{R_3}(p) = 0.5 - 0.3p$ gives $p_t \approx 0.444$

Solve MILP (3.12) for $p = p_t$ and obtain $f^* = 0.367$.

Since $0.367 = f^* = f^{R_2}(0.444) = 0.367$

the optimality region of $f^{R_2} = 0.1 + 0.6p$ is confirmed

Set $p^3 = 0.444$. Set $n_s = 2$.

$0.5 - 0.3p = f^{R_3}(p) = f^{R_4}(p) = 0.9 - 0.9p$ gives $p_t \approx 0.667$

Solve MILP (3.12) for $p = p_t$ and obtain $f^* = 0.3$.

Since $0.3 = f^* = f^{R_3}(0.667) = 0.3$

the optimality region of $f^{R_3} = 0.5 - 0.3p$ is confirmed

Set $p^4 = 0.667$ and $n_s = 3$.

Since $|R| - n_s = 4 - 3 = 1$ set $n_s = 4$.

Since $|R| = n_s$ terminate.

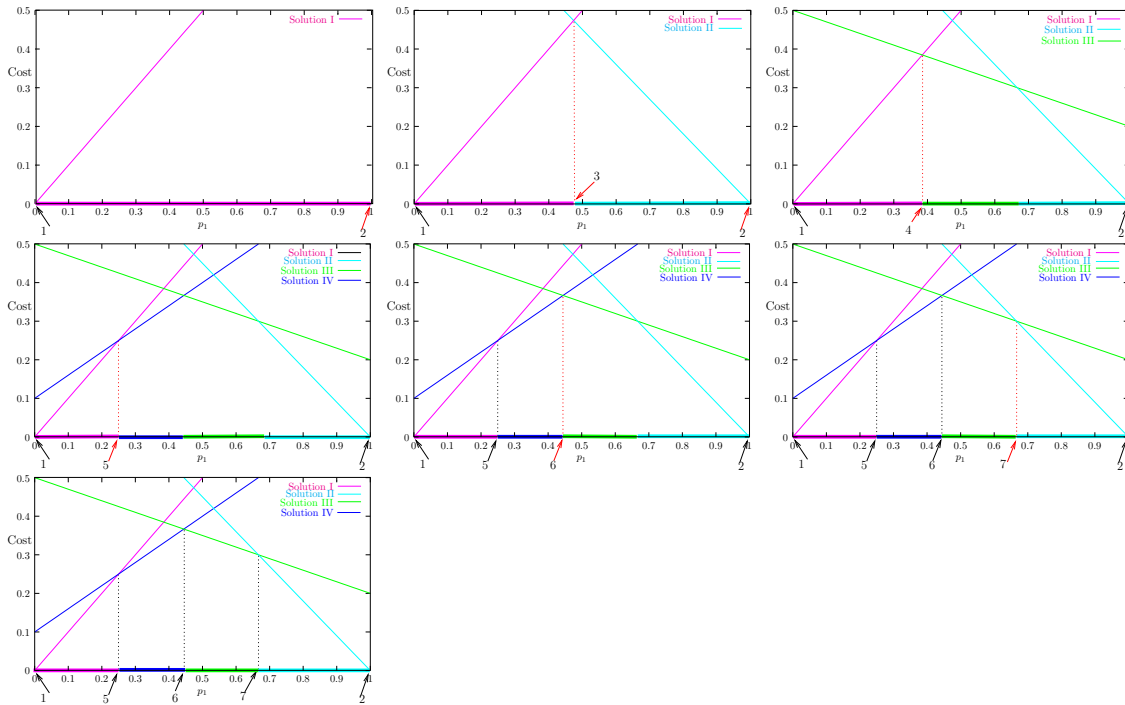


Figure 3-1: Graphical illustration of one-dimensional intersections-based algorithm for the parametric optimization in the cost vector case of (3.12). Solution I (magenta) corresponds to $\mathbf{y} = (1, 0, 0, 0)$, solution II (cyan) to $\mathbf{y} = (0, 0, 0, 1)$, solution III (green) to $\mathbf{y} = (0, 0, 1, 0)$, and solution IV (blue) to $\mathbf{y} = (0, 1, 0, 0)$.

Pathological Example

A subtle point is that the some of the optimality regions obtained by Algorithm 3.1 may be degenerate intervals (single points). As an example consider the simple IP example:

$$\begin{aligned} \min & (\mathbf{c} + \mathbf{C}p_1)^\top \mathbf{y} \\ \text{s.t.} & \sum_{i=1}^5 y_i = 1 \\ & \mathbf{y} \in \{0, 1\}^5, \end{aligned}$$

where $\mathbf{c} = (-0.4, -0.2, 0., 0.2, 0.6)$ and $\mathbf{C} = (1, 0.4, 0., -0.4, 1)$ The optimality regions of the five integer realizations are

$$\begin{aligned} (1, 0, 0, 0, 0) & \text{ is optimal for } p \in [0, \frac{1}{3}] \\ (0, 1, 0, 0, 0) & \text{ is optimal for } p \in [\frac{1}{3}, \frac{1}{2}] \\ (0, 0, 1, 0, 0) & \text{ is optimal for } p = \frac{1}{2} \\ (0, 0, 0, 1, 0) & \text{ is optimal for } p \in [\frac{1}{2}, \frac{2}{3}] \\ (0, 0, 0, 0, 1) & \text{ is optimal for } p \in [\frac{2}{3}, 1]. \end{aligned}$$

Algorithm 3.1 will call the IP solver for $p = 0.5$, which may return y_3 as the optimal solution. In this case a degenerate interval will be created. After termination of the algorithm, these degenerate intervals can be excluded.

3.4.3 Multiparametric Intersection-Based Algorithm

The basic idea of our proposed extension to the multiparametric case is to obtain the solutions that are optimal for some parameter value and construct the optimality polyhedra in the parameter space by identifying their vertices. The vertices are stored in set V and the solutions in set S . The set V is initialized with the vertices of P . At each iteration a vertex is picked and the MILP is called for the corresponding

parameter value. If the obtained optimal objective value is better than the existing solutions, the solution point is added to the set S . In Subroutine 3.1, existing vertices are checked to see if the new solution obtained is better than the assumed optimal and if it is, the vertices are removed from the list since they are no longer needed. In Subroutine 3.2 all points are constructed, that are possibly vertices of the optimality region of this solution. After an exclusion test, based on the existing solutions, the candidate vertices are added in Subroutine 3.3. Note that typically the calls to the MILP solver are much more expensive than the additional computations, and therefore these computations do not significantly affect the computational requirement.

Each element $S_j \in S$ is a pair composed of the solution of MILP (3.11) $(\mathbf{x}^{S_j}, \mathbf{y}^{S_j})$, and the coefficients $f_0^{S_j}, f_1^{S_j}, \dots, f_{n_p}^{S_j}$, of the objective value function $f^{S_j}(\mathbf{p}) = f_0^{S_j} + \sum_{k=1}^{n_p} f_k^{S_j} p_k$. The coefficients are stored to avoid recalculation every time the function is needed. Each element $V_k \in V$ is a quadruplet composed of the vertex coordinates \mathbf{p}^{V_k} , a boolean variable sol^{V_k} , which denotes if the MILP has been solved at this vertex, a value for the objective function f^{V_k} and finally a list of indices L^{V_k} , that are possibly optimal at this vertex. The elements of each list L^{V_k} are indices to elements of the set S . When the MILP at a given vertex has been solved ($sol^{V_k} = \text{true}$), the solutions are guaranteed optimal for the corresponding parameter value, otherwise ($sol^{V_k} = \text{false}$) they are only assumed optimal. As a termination criterion the number of unsolved vertices is stored in the integer n_V . Vertices at which the MILP has been solved are kept, since they are needed for the output of the algorithm. For simplicity we assume that the numbering of the vertices is not changed when vertices are removed.

Algorithm 3.2. Multiparametric Intersection-Based Algorithm

1. Initialization

- Solve (3.11) for $\mathbf{p} = \mathbf{0}$.
- Set $f^*(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^* + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^*$

- Create S with the pair $((\mathbf{x}^*, \mathbf{y}^*), f^*(\mathbf{p}))$.
- Initialize V with the 2^{n_p} vertices of the parameter space, with $sol^{V_1} = \text{true}$ and $sol^{V_{k \neq 1}} = \text{false}$, $f^{V_k}(\mathbf{p}) = f^*(\mathbf{p})$ and $L^{V_k} = \{1\}$ for $k = 1, \dots, 2^{n_p}$.
- Set $n_V = 2^{n_p} - 1$.

2. REPEAT

- Pick a vertex $V_k \in V$ with $sol^{V_k} = \text{false}$ and solve MILP (3.11) for $\mathbf{p} = \mathbf{p}^{V_k}$.
- Set $f^*(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^* + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^*$
- **IF** $f^*(\mathbf{p}^{V_k}) < f^{V_k}$ **THEN**
 - (a) **CALL** *Check existing vertices(k)* (Subroutine 3.1).
 - (b) **CALL** *Obtain candidate points* (Subroutine 3.2).
 - (c) Add $((\mathbf{x}^*, \mathbf{y}^*), f^*(\mathbf{p}))$ to S .
- **ELSE**
 - (a) Set $sol^{V_k} = \text{true}$.
 - (b) Set $n_V = n_V - 1$.

UNTIL $n_V = 0$

On termination, V contains all vertices of the optimality regions, and the entire parameter range has been characterized. At this point, the sets V and S can be reordered, to explicitly define the polyhedra. If this is done, obtaining an optimal solution at a given parameter point is a relatively simple procedure by successively checking each polyhedron to see if the point belongs to it.

Finite termination is guaranteed since by assumption the MILP solver furnishes vertices of the feasible set and therefore a finite number of solutions is considered. Cycling is not possible because a new solution is only introduced if at a vertex it is better than all existing ones.

The number of MILP calls needed is equal to the number of vertices of the divided parameter host set, plus the number of solutions identified in the interior of P . To minimize the number of required MILP calls the vertices of P should be visited first.

Note that at least 2^{n_p} MILP calls are needed for the vertices of P making Algorithm 3.2 at least exponential in the number of parameters. Moreover for a large number of optimal solutions, the number of vertices in the interior of P can be quite high.

The following subroutine ensures that parameter points that are not vertices of a polyhedron are eliminated. This is done by first comparing the newly obtained objective value function with the assumed optimal objective value function for this parameter point. If it is not better, the parameter point remains unchanged. If the new objective value function is better than the presumed optimal, then the parameter point can not be a vertex of the polyhedron corresponding to the old objective value function. In that case, if the parameter point is a vertex of P , then it is assigned to the new solution, otherwise it is eliminated from the set of vertices.

Subroutine 3.1. Check existing vertices(k)

FORALL $V_l \in V$ with $sol^{V_l} = \text{false}$

• **IF** $f^*(\mathbf{p}^{V_l}) < f^{V_l}$ **THEN**

– **IF** $\mathbf{p}^{V_l} \notin \{0, 1\}^{n_p}$ **THEN**

* Remove V_l from V .

* Set $n_V = n_V - 1$.

ELSE

* $L^{V_l} = \{|S| + 1\}$ (assign this vertex to the new solution)

* **IF** $l = k$ **THEN** set $sol^{V_k} = \text{false}$, $n_V = n_V - 1$.

END

At a given iteration we have $|S|$ previous solutions as well as the new solution. For $|S| = 0$ the vertices in the parameter space are assigned to the new solution and no new candidate points are needed. For $|S| > 0$ we need to construct all the possible vertices that involve the new solution by solving a set of n_p equations. To that extent for $i = 1, \dots, \max(n_p, |S|)$ we set up an n_p dimensional index set J^i ; the first i elements are indices corresponding to elements of the solution set S , i.e., J_j^i corresponds to $S^{J_j^i}$; the last $n_p - i$ elements are indices corresponding to the elements

of \mathbf{p} . For every possible value of the index set J^i , 2^{n_p-i} systems of n_p linear equations are solved to identify new candidate vertices. Note that this is a highly combinatorial step, but the computational time will be insignificant compared to the solution of the MILPs.

Subroutine 3.2. Obtain candidate points

- . **FOR** $i = 1, \dots, \max(n_p, |S|)$
 - **FORALL** $\binom{|S|}{i} \times \binom{n_p}{n_p-i}$ possible values of J^i
 - **FORALL** $\mathbf{d} \in \{0, 1\}^{n_p-i}$
 1. (Set up matrix \mathbf{A}^t and right hand side vector \mathbf{b}^t)
 - * **FOR** $j = 1, \dots, i$
 - Set $b_j^t = -f_0^* + f^{S_{J_j^i}}$
 - **FOR** $k = 1, \dots, n_p$ set $A_{jk}^t = f_k^* - f_k^{S_{J_j^i}}$.
 - * **FOR** $j = i + 1, \dots, n_p$
 - Set $b_j^t = d_{j-i}$.
 - **FOR** $k = 1, \dots, n_p$ set $A_{jk}^t = \begin{cases} 1 & \text{if } k = J_j^i \\ 0 & \text{otherwise.} \end{cases}$
 2. Solve $\mathbf{A}^t \mathbf{p}_t = \mathbf{b}^t$.
 3. **IF** $\mathbf{p}_t \in P$ and $f^*(\mathbf{p}_t) \leq \min_{S_j \in S} f^{S_j}(\mathbf{p}_t)$ **THEN CALL** *Add point*.

A word of caution is needed for Step 2 of Subroutine 3.2. The matrix \mathbf{A}^t need not be nonsingular and $\mathbf{A}^t \mathbf{p}_t = \mathbf{b}^t$ may have no solution, or an infinite number of solutions. Nevertheless, neither case is of concern. In the case of no solution no point needs to be considered. The case of infinitely many solutions occurs when for the given fixed parameter values (the last $n_p - i$ elements of J^i) some of the objective value functions considered are linearly dependent. In that case a subspace of P satisfies $\mathbf{A}^t \mathbf{p}_t = \mathbf{b}^t$. Out of this subspace, only the points that intersect with one of the faces of P correspond to vertices of a polyhedron. These can be obtained by adding constraints $p_j \in \{0, 1\}$. Since in Subroutine 3.2 all possible J^i are considered, when

singularity of \mathbf{A}^t is detected, this index set J^i can be ignored for all $\mathbf{d} \in \{0, 1\}^{n_p-i}$. In a previous iteration more parameters were fixed, and less functions were considered and these furnished the vertices required.

Subroutine 3.3. Add point

1. Create $L^{newV} = \{|S| + 1\}$
FOR $j = 1, \dots, |S|$
 - **IF** $f^{S_j}(\mathbf{p}_t) = f^*(\mathbf{p}_t)$ **THEN** $L^{newV} = L^{newV} \cup \{j\}$.
2. Create a new quadruplet with $\mathbf{p}^{newV} = \mathbf{p}_t$, $sol^{newV} = \text{false}$, $f(\mathbf{p})^{newV} = f^*(\mathbf{p})$, and L^{newV} .
3. Insert the new quadruplet in set V .

Illustrative Example

To illustrate the behavior of Algorithm 3.2 consider a simple (IP)

$$\begin{aligned}
 & \min_{\mathbf{y}} (\mathbf{c} + \mathbf{C}\mathbf{p})^T \mathbf{y} \\
 & \text{s.t.} \quad \sum_{i=1}^3 y_i = 1 \\
 & \quad \quad \mathbf{y} \in \{0, 1\}^3,
 \end{aligned} \tag{3.13}$$

where

$$\mathbf{c} = \begin{pmatrix} 0 \\ 1.0 \\ 0.9 \end{pmatrix} \text{ and } \mathbf{C} = \begin{bmatrix} 1 & -0.8 \\ 0.1 & -2 \\ -0.9 & -0.6 \end{bmatrix}.$$

Solution $(1, 0, 0)$ is optimal for values close to $\mathbf{p} = \mathbf{0}$, while solution $(0, 1, 0)$ is optimal for large values of p_2 , and solution $(0, 0, 1)$ is optimal for large values of p_1 .

The steps of Algorithm 3.2 are summarized below. Figure 3-2 contains a graphical illustration. The optimality regions are drawn as polygons of different color in the

two-dimensional parameter space. Vertices at which the MILP has been solved are marked with a circle and the other vertices with a square.

Initialization

Solve (3.13) for $\mathbf{p} = 0$ and obtain the solution $\mathbf{y}^* = (1, 0, 0)$.

Create the solution set S with $\mathbf{y}^{S_1} = \mathbf{y}^*$ and $f^{S_1}(\mathbf{p}) = 0 + 1p_1 - 0.8p_2$.

Add the four vertices of P to the set of vertices V

$$\mathbf{p}^{V_1} = (0, 0), \text{sol}^{V_1} = \text{true}, f^{V_1} = 0, L^{V_1} = \{1\}$$

$$\mathbf{p}^{V_2} = (0, 1), \text{sol}^{V_2} = \text{false}, f^{V_2} = -0.8, L^{V_2} = \{1\}$$

$$\mathbf{p}^{V_3} = (1, 0), \text{sol}^{V_3} = \text{false}, f^{V_3} = 1, L^{V_3} = \{1\}$$

$$\mathbf{p}^{V_4} = (1, 1), \text{sol}^{V_4} = \text{false}, f^{V_4} = 0.2, L^{V_4} = \{1\}.$$

Set $n_V = 3$.

Main loop

Select vertex V_2 with $p_1 = 0, p_2 = 1$

Solve MILP (3.13) and obtain $\mathbf{y}^* = (0, 1, 0)$ and $f^* = -1.0$.

Set $f^*(\mathbf{p}) = 1.0 + 0.1p_1 - 2p_2$.

Since $-1.0 = f^*(\mathbf{p}^{V_2}) < f^{V_2} = -0.8$

Check existing vertices

Assign vertex V^2 to the new solution and mark as solved

$$\text{sol}^{V_2} = \text{true}, f^{V_2} = -1.0, L^{V_k} = \{2\}, n_V = 2.$$

Since $1.1 = f^*(\mathbf{p}^{V_3}) > f^{V_3} = 1$ do not reassign vertex V^3 .

Since $0.2 = f^*(\mathbf{p}^{V_4}) < f^{V_4} = -0.9$ assign vertex V^4 to the new solution

$$f^{V_4} = -0.9, L^{V_4} = \{2\}.$$

Obtain candidate vertices

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 0$ gives $p_1 = 1.111$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 1$ gives $p_1 = -0.222$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 0$ gives $p_2 = 0.8333$.

Set $n_V = 3$ and create a new vertex.

$$\mathbf{p}^{V_5} = (0, 0.833), \text{sol}^{V_5} = \text{true}, f^{V_5} = -0.666, L^{V_5} = \{1, 2\}.$$

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 1$ gives $p_2 = 0.08333$.

Set $n_V = 4$ and create a new vertex.

$$\mathbf{p}^{V_6} = (0, 0.833), \text{sol}^{V_6} = \text{false}, f^{V_6} = -0.666, L^{V_6} = \{1, 2\}.$$

Create S_2 with $\mathbf{y}^{S_2} = \mathbf{y}^*$ and $f^{S_2}(\mathbf{p}) = f^*(\mathbf{p})$ and add it to S .

Select vertex V_3 with $p_1 = 1, p_2 = 0$

Solve MILP (3.13) and obtain $\mathbf{y}^* = (0, 0, 1)$ and $f^* = 0.0$.

Set $f^*(\mathbf{p}) = 0.9 - 0.9p_1 - 0.6p_2$.

Since $0.0 = f^*(\mathbf{p}^{V_3}) < f^{V_3} = 1.0$.

Check existing vertices

Assign vertex V^3 to the new solution and marked as solved

$$\text{sol}^{V_3} = \text{true}, f^{V_3} = 0.0, L^{V_3} = \{3\}, n_V = 3.$$

Since $-0.6 = f^*(\mathbf{p}^{V_4}) > f^{V_4} = -0.9$ do not reassign vertex V^4 .

Since $0.4 = f^*(\mathbf{p}^{V_5}) > f^{V_5} = -0.6664$ do not reassign vertex V^5 .

Since $0.933 = f^*(\mathbf{p}^{V_6}) < f^{V_6} = -0.04998$ delete vertex V^6 and set $n_V = 2$.

Obtain candidate vertices

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 0$ gives $p_2 = -4.5$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 1$ gives $p_2 = 5.0$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 0$ gives $p_1 = 0.474$.

Since $0.474 = f^*(0.474, 0) < f^{S_2}(0.474, 0) = 1.047$

Set $n_V = 3$ and create a new vertex

$$\mathbf{p}^{V_7} = (0.474, 0), \text{sol}^{V_7} = \text{false}, f^{V_7} = 0.474, L^{V_7} = \{1, 3\}.$$

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 1$ gives $p_1 = 0.578$.

Since $-0.221 = f^*(0.578, 1) > f^{S_2}(0.578, 1) = -0.942$ no new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_1 = 0$ gives $p_2 = 0.071$.

Since $0.857 = f^*(0, 0.071) > f^{S_1}(0, 0.071) = -0.057$ no new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_1 = 1$ gives $p_2 = 0.786$.

Since $-0.471 = f^*(1, 0.786) < f^{S_1}(1, 0.786) = 0.371$

set $n_V = 4$ and create a new vertex

$$\mathbf{p}^{V_8} = (1, 0.786), \text{sol}^{V_8} = \text{false}, f^{V_8} = -0.471, L^{V_8} = \{2, 3\}.$$

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_2 = 0$ gives $p_1 = -0.1$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_2 = 1$ gives $p_1 = 1.3$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$

gives $p_1 = 0.520$ and $p_2 = 0.443$.

Set $n_V = 5$ and create a new vertex

$$\mathbf{p}^{V_9} = (0.52, 0.443), \text{sol}^{V_9} = \text{false}, f^{V_9} = 0.166, L^{V_9} = \{1, 2, 3\}$$

Create S_3 with $\mathbf{y}^{S_3} = \mathbf{y}^*$ and $f^{S_3}(\mathbf{p}) = f^*(\mathbf{p})$ and add it to S .

Select vertex V_4 with $p_1 = 1, p_2 = 1$

Solve MILP (3.13) and obtain $f^* = -0.9$.

Since $f^*(\mathbf{p}^{V_4}) = f^{V_4}$ set $\text{sol}^{V_4} = \text{true}$, $n_V = 4$.

Select vertex V_5 with $p_1 = 0, p_2 = 0.833$

Solve MILP (3.13) and obtain $f^* = -0.6664$.

Since $f^*(\mathbf{p}^{V_5}) = f^{V_5}$ set $\text{sol}^{V_5} = \text{true}$, $n_V = 3$.

Select vertex V_7 with $p_1 = 0.474, p_2 = 0$.

Solve MILP (3.13) and obtain $f^* = 0.474$.

Since $f^*(\mathbf{p}^{V_7}) = f^{V_7}$ set $\text{sol}^{V_7} = \text{true}$, $n_V = 2$.

Select vertex V_8 with $p_1 = 1, p_2 = 0.786$

Solve MILP (3.13) and obtain $f^* = -0.471$.

Since $f^*(\mathbf{p}^{V_8}) = f^{V_8}$ set $\text{sol}^{V_8} = \text{true}$, $n_V = 1$.

Select vertex V_9 with $p_1 = 0.52, p_2 = 0.443$

Solve MILP (3.13) and obtain $f^* = -0.166$.

Since $f^*(\mathbf{p}^{V_9}) = f^{V_9}$ set $\text{sol}^{V_9} = \text{true}$, $n_V = 0$.

Since $n_V = 0$ terminate.

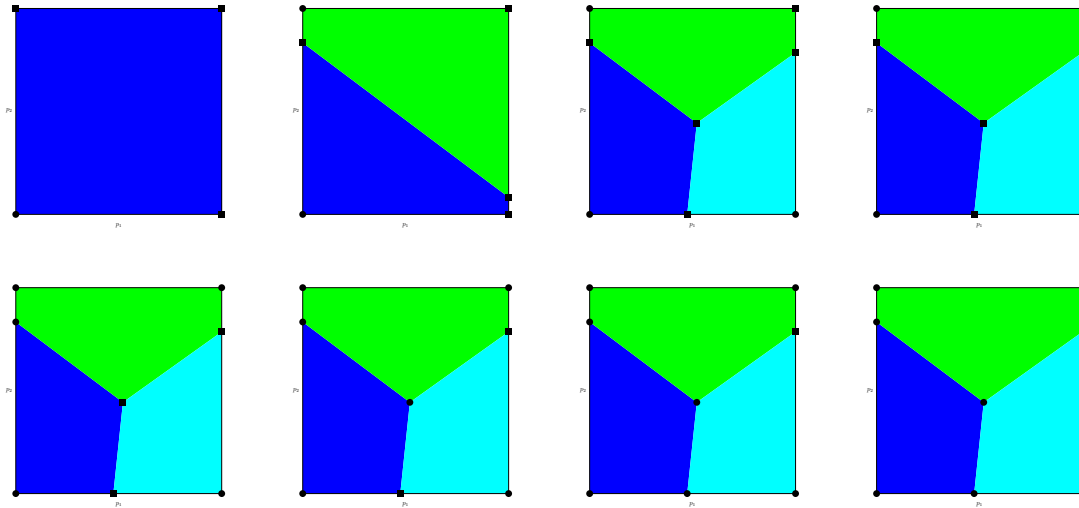


Figure 3-2: Graphical illustration of Algorithm 3.2 for example (3.13)). Blue corresponds to $\mathbf{y} = (1, 0, 0)$, green to $\mathbf{y} = (0, 1, 0)$, and cyan to $\mathbf{y} = (0, 0, 1)$. The algorithm requires 8 calls for 3 solutions and 8 vertices. Vertices in the parameter space are marked with a square when the optimal solution is not verified, and with a circle when it has.

3.4.4 Multiparametric Optimality-Region Algorithm

In Algorithm 3.2 the optimality region of a given solution is confirmed by solving the MILP at the vertices of the optimality region. In the multiparametric case, even a small number of optimal solutions can create a large number of vertices in the parameter space and the intersection-based algorithm requires many MILP calls. Here we propose an algorithm based on formulation (3.3) that verifies an assumed optimality region. This algorithm requires a number of iterations that is equal to the number of solutions, and therefore typically much less than the number of vertices. Whether this algorithm can outperform the intersection-based algorithm depends on the instance considered because (3.14) is typically more expensive to solve than (3.11). A particularly interesting case, is that only the cost vector of binary variables is affected, because in that case (3.14) can be reformulated as a MILP, see Section 3.3.2.

Since the feasible region does not depend on the parameters, given a set of solutions

S with the feasible points $\mathbf{x}^{S_j}, \mathbf{y}^{S_j}$ and the corresponding (affine) objective functions

$$f^{S_j}(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^{S_j} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^{S_j},$$

the following formulation checks whether an additional solution needs to be considered

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}, \mathbf{p}, u} \quad & u \\ \text{s.t.} \quad & (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y} - f^{S_j}(\mathbf{p}) \leq u, \quad \forall S_j \in S \\ & \mathbf{A}^x \mathbf{x} + \mathbf{A}^y \mathbf{y} = \mathbf{b} \quad (3.14) \\ & \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\ & \mathbf{p} \in [0, 1]^{n_p}, \quad \mathbf{y} \in \{0, 1\}^{n_y} \\ & u \in \mathbb{R}. \end{aligned}$$

If the optimal solution value of (3.14) is less than zero, $\mathbf{x}^*, \mathbf{y}^*$ has to be added to the set of solutions. We assume again that the solver used for (3.14) furnishes a vertex of the feasible region of (3.11). If that is not the case (3.11) needs to be solved for the parameter value furnished by (3.14).

The same sets as in Algorithm 3.2 are used, but here the integer sol^{V_k} is not needed for the vertices in set V .

Algorithm 3.3. Multiparametric Optimality Region Algorithm

1. Initialization

- Solve (3.11) for $\mathbf{p} = \mathbf{0}$.
- Set $f^*(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x}^* + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y}^*$
- Create S with the pair $(\mathbf{x}^*, \mathbf{y}^*), f^*(\mathbf{p})$.
- Initialize V with the 2^{n_p} vertices of the parameter space, with $f^{V_k}(\mathbf{p}) = f^*(\mathbf{p})$ and $L^{V_k} = \{1\}$ for $k = 1, \dots, 2^{n_p}$.

2. LOOP

- Solve (3.14)
 - IF** $u^* < 0$
 - (a) Set $f^*(\mathbf{p}) = (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^T \mathbf{x}^* + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^T \mathbf{y}^*$
 - (b) **CALL** *Check existing vertices*(0) (Subroutine 3.1 for a dummy k).
 - (c) **CALL** *Obtain candidate points* (Subroutine 3.2).
 - (d) Insert $((\mathbf{x}^*, \mathbf{y}^*), f^*(\mathbf{p}))$ in set S .
 - ELSE**
 - (a) Terminate.
- END**

END

The number of calls to the optimality region formulation (3.14) needed is equal to the number of optimal solutions. Additionally, a single call to the original MILP (3.11) is needed. As in Algorithm 3.2, finite termination is guaranteed since by assumption the MILP solver furnishes vertices of the feasible set and therefore a finite number of solutions is considered. Cycling is not possible because a new solution is only introduced if a parameter point exists for which the new solution is better than all existing solutions.

Illustrative Example

The steps of Algorithm 3.3 for example (3.13) are summarized below. Figure 3-3 contains a graphical illustration. The optimality regions are drawn as polygons of different color in the two-dimensional parameter space. Vertices at which we know the optimal solution are marked with a circle and the other vertices with a square. Note that since (3.13) is a purely integer program, MINLP (3.14) can be reformulated to a MILP and solved with any MILP solver.

Initialization

Solve (3.13) for $\mathbf{p} = (0, 0)$ and obtain the solution $\mathbf{y}^* = (1, 0, 0)$.

Create the solution set S with $\mathbf{y}^{S_1} = \mathbf{y}^*$ and $f^{S_1}(\mathbf{p}) = 0 + 1p_1 - 0.8p_2$.

Add the four vertices of P to the set of vertices V

$$\mathbf{p}^{V_1} = (0, 0), f^{V_1} = 0, L^{V_1} = \{1\}$$

$$\mathbf{p}^{V_2} = (0, 1), f^{V_2} = -0.8, L^{V_2} = \{1\}$$

$$\mathbf{p}^{V_3} = (1, 0), f^{V_3} = 1, L^{V_3} = \{1\}$$

$$\mathbf{p}^{V_4} = (1, 1), f^{V_4} = 0.2, L^{V_4} = \{1\}.$$

Main loop

Solve (3.14) and obtain $u^* = -1.1$, $\mathbf{p}^* = (1, 1)$ and $\mathbf{y}^* = (0, 1, 0)$.

Set $f^*(\mathbf{p}) = 1.0 + 0.1p_1 - 2p_2$.

Check existing vertices

Since $-1.0 = f^*(\mathbf{p}^{V_2}) < f^{V_2} = -0.8$ assign vertex V^2 to the new solution.

$$f^{V_2} = -1.0, L^{V_2} = \{2\}.$$

Since $1.1 = f^*(\mathbf{p}^{V_3}) > f^{V_3} = 1$ do not reassign vertex V^3 .

Since $0.2 = f^*(\mathbf{p}^{V_4}) < f^{V_4} = -0.9$ assign vertex V^4 to the new solution

$$f^{V_4} = -0.9, L^{V_4} = \{2\}.$$

Obtain candidate vertices

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 0$ gives $p_1 = 1.111$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 1$ gives $p_1 = -0.222$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 0$ gives $p_2 = 0.8333$.

Create a new vertex. $\mathbf{p}^{V_5} = (0, 0.833)$, $f^{V_5} = -0.666$, $L^{V_5} = \{1, 2\}$.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 1$ gives $p_2 = 0.08333$.

Create a new vertex $\mathbf{p}^{V_6} = (1, 0.833)$, $f^{V_6} = -0.666$, $L^{V_6} = \{1, 2\}$.

Create S_2 with $\mathbf{y}^{S_2} = \mathbf{y}^*$ and $f^{S_2}(\mathbf{p}) = f^*(\mathbf{p})$ and add it to S .

Solve (3.14) and obtain $u^* = -1.0$, $p_1^* = 1.0$, $p_2^* = 0.0$ and $\mathbf{y}^* = (0, 0, 1)$.

Set $f^*(\mathbf{p}) = 0.9 - 0.9p_1 - 0.6p_2$.

Check existing vertices

Since $0 = f^*(\mathbf{p}^{V_3}) > f^{V_3} = 1$ assign vertex V^3 to the new solution.

$$f^{V_3} = 0.0, L^{V_3} = \{3\}.$$

Since $-0.6 = f^*(\mathbf{p}^{V_4}) > f^{V_4} = -0.9$ do not reassign vertex V^4 .

Since $0.4 = f^*(\mathbf{p}^{V_5}) > f^{V_5} = -0.6664$ do not reassign vertex V^5 .

Since $0.933 = f^*(\mathbf{p}^{V_6}) < f^{V_6} = -0.04998$ delete vertex V^6 .

Obtain candidate vertices

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 0$ gives $p_2 = -4.5$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_1 = 1$ gives $p_2 = 5.0$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 0$ gives $p_1 = 0.474$.

$$\text{Since } 0.474 = f^*(0.474, 0) < f^{S_2}(0.474, 0) = 1.047$$

Create a new vertex $\mathbf{p}^{V_7} = (0.474, 0)$, $f^{V_7} = 0.474$, $L^{V_7} = \{1, 3\}$.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $p_2 = 1$ gives $p_1 = 0.578$.

Since $-0.221 = f^*(0.578, 1) > f^{S_2}(0.578, 1) = -0.942$ no new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_1 = 0$ gives $p_2 = 0.071$.

Since $0.857 = f^*(0, 0.071) > f^{S_1}(0, 0.071) = -0.057$ no new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_1 = 1$ gives $p_2 = 0.786$.

$$\text{Since } -0.471 = f^*(1, 0.786) < f^{S_1}(1, 0.786) = 0.371$$

Create a new vertex $\mathbf{p}^{V_8} = (1, 0.786)$, $f^{V_8} = -0.471$, $L^{V_8} = \{2, 3\}$.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_2 = 0$ gives $p_1 = -0.1$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$ with $p_2 = 1$ gives $p_1 = 1.3$,

which does not satisfy $\mathbf{p} \in [0, 1]^2$. No new vertex is created.

The intersection of $f^*(\mathbf{p}) = f^{S_1}(\mathbf{p})$ with $f^*(\mathbf{p}) = f^{S_2}(\mathbf{p})$

gives $p_1 = 0.520$ and $p_2 = 0.443$.

Create a new vertex $\mathbf{p}^{V_9} = (0.52, 0.443)$, $f^{V_9} = 0.166$, $L^{V_9} = \{1, 2, 3\}$

Create S_3 with $\mathbf{y}^{S_3} = \mathbf{y}^*$ and $f^{S_3}(\mathbf{p}) = f^*(\mathbf{p})$ and add it to S .

Solve (3.14) and obtain $u^* = 0$

terminate.

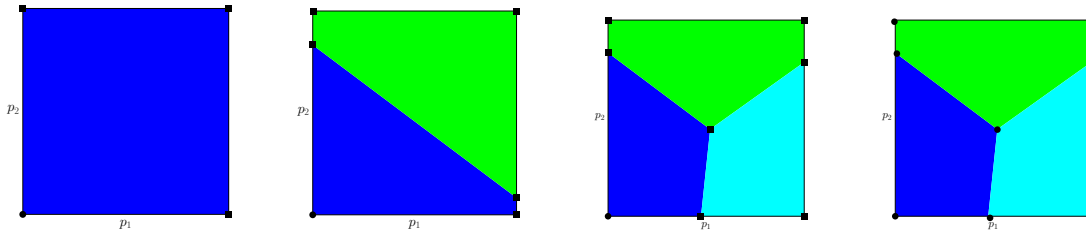


Figure 3-3: Graphical illustration of the Algorithm 3.3 for example (3.13)). Blue corresponds to $\mathbf{y} = (1, 0, 0)$, green to $\mathbf{y} = (0, 1, 0)$, and cyan to $\mathbf{y} = (0, 0, 1)$. The algorithm requires 3 optimality region formulations for 3 solutions and 8 vertices. Vertices in the parameter space are marked with a square when the optimal solution is not verified, and with a circle when it has.

3.4.5 Multiparametric Branch-and-Bound Algorithm

An alternative to algorithms based on MILP calls, is to perform branch-and-bound (B&B) on the binary variables and solve a parametric linear program at each node as has been proposed for the right hand side case by Ohtake and Nishida [216]. The promise of this method is that the cost vector case of parametric linear programs is in principle slightly more expensive than a linear program [117]. A drawback is that advanced features of MILP algorithms such as branch and cut strategies are not available for the parametric case. Here we describe a basic B&B procedure where the parametric linear programs are treated as a black box that return a set of optimal solutions. We allow the possibility to also inherit a set of solutions from the parent node, but do not explore the option of inheriting optimality regions. For better computational performance it may be beneficial to do so, but this would make the description much more cumbersome.

Let A denote an index set that contains the indices i of the active nodes. At each

node i a parametric LP is solved for $\mathbf{p} \in [0, 1]^{n_p}$

$$\begin{aligned}
f^i(\mathbf{p}) &= \min_{\mathbf{x}, \mathbf{y}} (\mathbf{c}^x + \mathbf{C}^x \mathbf{p})^\top \mathbf{x} + (\mathbf{c}^y + \mathbf{C}^y \mathbf{p})^\top \mathbf{y} \\
\text{s.t. } &\mathbf{A}^x \mathbf{x} + \mathbf{A}^y \mathbf{y} = \mathbf{b} \\
&\mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x} \geq \mathbf{0} \\
&y_j = 0, \quad \forall j \in Z^i \\
&y_j = 1, \quad \forall j \in O^i \\
&y_j \in [0, 1], \quad \forall j \notin (Z^i \cup O^i),
\end{aligned} \tag{3.15}$$

where Z^i and O^i are index sets corresponding to the binary variables fixed to integer values at node i . The optimal objective value function $f^i(\mathbf{p})$ is a lower bound for node i in the branch-and-bound procedure. The answer of the parametric LP is a set of optimal solutions S^i (empty if (3.15) is infeasible). Each element of S^i is a triplet composed of the solution of the relaxed LP (3.15) $(\mathbf{x}^{S_j^i}, \mathbf{y}^{S_j^i})$, the coefficients $f_0^{S_j^i}, f_1^{S_j^i}, \dots, f_{n_p}^{S_j^i}$, of the objective value function $f^{S_j^i}(\mathbf{p}) = f_0^{S_j^i} + \sum_{k=1}^{n_p} f_k^{S_j^i} p_k$ and a boolean $\text{bin}^{S_j^i}$ which denotes if the solution is binary feasible

$$\text{bin}^{S_j^i} = \begin{cases} \text{true} & \text{if } \mathbf{y}^{S_j^i} \in \{0, 1\}^{n_y} \\ \text{false} & \text{otherwise.} \end{cases}$$

The set of solutions covers the whole parameter space, i.e.,

$$f^i(\mathbf{p}) = \min_{S_j^i \in S} f^{S_j^i}(\mathbf{p}), \quad \forall \mathbf{p} \in [0, 1]^{n_p}.$$

For the upper bound a solution set S^U is used. Each element S_j^U of S^U is a pair composed of a solution $(\mathbf{x}^{S_j^U}, \mathbf{y}^{S_j^U})$ and the coefficients $f_0^{S_j^U}, f_1^{S_j^U}, \dots, f_{n_p}^{S_j^U}$, of the objective value function $f^{S_j^U}(\mathbf{p}) = f_0^{S_j^U} + \sum_{k=1}^{n_p} f_k^{S_j^U} p_k$.

For the comparison of the lower bound of a node i with the upper bound every solution $f^{S_j^i}(\mathbf{p})$ of S^i has to be checked to see if for some parameter value it provides a better solution than the minimum of the solutions of S^U . This check can be performed

with the following LP

$$\begin{aligned}
& \max_{\mathbf{p}} u \\
& \text{s.t. } u \leq f^{S_k^U}(\mathbf{p}) - f^{S_j^i}(\mathbf{p}), \quad \forall S_k^U \in S^U \\
& \mathbf{p} \in [0, 1]^{n_p}.
\end{aligned} \tag{3.16}$$

If the optimal solution u^* is greater than zero, $f^{S_j^i}(\mathbf{p})$ is better than the upper bound for the optimal parameter value \mathbf{p}^* . This means that if $\text{bin}^{S_j^i} = \text{true}$ then the upper bound needs to be updated.

There are two fathoming criteria for a node and both extend ideas from regular branch-and-bound for MILP. One criterion is that the lower bound to the node is not better than the existing upper bound. The other criterion is that all solutions are binary feasible which means that the upper and lower bound of the node have converged, and therefore this node cannot provide a better upper bound.

Algorithm 3.4. Multiparametric Branch-and-Bound Algorithm

1. **Initialization**

- Set $A = \{0\}$
- Set $Z^0 = O^0 = \emptyset$
- Set $k = 1$.
- Set $S^U = \emptyset$.
- Set $S^0 = \emptyset$.

2. **(Termination Test)** If $A = \emptyset$ then terminate.

3. **(Node Selection)** Select and delete a node i from A .

4. **(Check lower bound)** For all $S_j^i \in S^i$

- Solve (3.16)

- **IF** $u^* \leq 0$ **THEN** Remove solution S_j^i from S^i .

IF $S^i = \emptyset$ and $k > 1$ **THEN GOTO** 2.

5. (**Relaxation**) Solve the parametric LP (3.15) of node i and obtain the set of solutions S^i .

6. (**Update upper bound**) For all $S_j^i \in S^i$

- Solve (3.16)
- **IF** $u^* > 0$ and $\text{bin}^{S_j^i} = \text{true}$ **THEN** Add solution S_j^i to S^U .
- **IF** $u^* \leq 0$ **THEN** Remove solution S_j^i from S^i .

7. (**Branching**)

IF $S^i \neq \emptyset$ and $\text{bin}^{S_j^i} = \text{false}$ for some $S_j^i \in S^i$ **THEN**

- Select a free binary variable $j \notin Z^i \cup O^i$
- Create subproblems with $Z^k = Z^i \cup j$, $O^k = O^i$ and $Z^{k+1} = Z^i$, $O^k = O^i \cup j$.
Set $S^{k+1} = S^k = S^i$.
- Add nodes k and $k + 1$ to A .
- Set $k = k + 2$.

8. **GOTO** 2.

At termination S^U contains all optimal solutions over the parameter range. A partition of the parameter set P can be done by successively creating the vertices as in Algorithms 3.2 and 3.3.

Illustrative Example

To demonstrate the behavior of Algorithm 3.4 consider a simple (IP) with a single parameter

$$\begin{aligned} \min_{\mathbf{y}} (\mathbf{c} + \mathbf{C} \mathbf{p})^T \mathbf{y} \\ \text{s.t. } 0.8x_1 + x_2 \leq 1.5 \\ \mathbf{y} \in \{0, 1\}^2, \end{aligned} \tag{3.17}$$

where

$$\mathbf{c} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} \text{ and } \mathbf{C} = \begin{bmatrix} -3 \\ 2 \end{bmatrix}.$$

There are three feasible integer realizations

$$\begin{aligned} (1, 0) \text{ is optimal for } p \in [0, 0.2] \\ (0, 1) \text{ is optimal for } p \in [0.2, 1] \\ (0, 0). \end{aligned}$$

The steps of Algorithm 3.4 after initialization are summarized below.

Select the root node $\mathbf{y} \in [0, 1]^2$.

The parametric relaxation gives four solutions

$$\begin{aligned} y_1 = 0, y_2 = 1, f = -1 + 2p, \text{ bin} = \text{true} \\ y_1 = \frac{2}{3}, y_2 = 1, f = -1, \text{ bin} = \text{false} \\ y_1 = 1, y_2 = 0.75, f = -1.5p - 0.75, \text{ bin} = \text{false} \\ y_1 = 1, y_2 = 0, f = -3p, \text{ bin} = \text{true}. \end{aligned}$$

Add the two binary feasible solutions to the upper bound

$$\begin{aligned} y_1 = 0, y_2 = 1, f = -1 + 2p, \text{ bin} = \text{true} \\ y_1 = 1, y_2 = 0, f = -3p, \text{ bin} = \text{true}. \end{aligned}$$

Branch the node

Select variable y_1 for branching

Termination criterion not met.

Select node 1: $y_1 = 0, y_2 \in [0, 1]$.

The parametric relaxation gives two solutions

$$y_1 = 0, y_2 = 1, f = -1 + 2p, bin = \text{true}$$

$$y_1 = 0, y_2 = 0, f = 0, bin = \text{true}.$$

Neither solution better than existing upper bound

Do not update the upper bound.

Do not branch the node.

Select node 2: $y_1 = 1, y_2 \in [0, 1]$.

The parametric relaxation gives two solutions

$$y_1 = 1, y_2 = 0.75, f = -1.5p - 0.75, bin = \text{false}$$

$$y_1 = 1, y_2 = 0, f = -3p, bin = \text{true}.$$

Binary feasible solution not better than upper bound

do not update the upper bound.

Branch the node

Select variable y_2 for branching

Termination criterion not met

Select node 3: $y_1 = 1, y_2 = 0$.

The parametric relaxation gives one solution

$$y_1 = 1, y_2 = 0, f = -3p, bin = \text{true}.$$

Solution not better than existing upper bound

Do not update the upper bound.

Do not branch the node.

Select node 4: $y_1 = 1, y_2 = 1$.

The parametric relaxation gives no solution (infeasible)

Termination criterion met.

Note that the optimal solutions were found at the root node, but 4 further nodes were required to confirm these.

3.5 General Case with a Single Parameter

In this section we consider (3.1) with a single parameter $p \in [0, 1]$. We first consider a parametric linear program and then the mixed-integer linear case. Note that in either case we allow violation of the primal constraints and marginal values by a tolerance $\varepsilon > 0$. Finally we discuss extensions to the nonlinear case and to the multiparametric case.

3.5.1 Assumptions and Theoretical Properties

Throughout this section we will exclude unbounded optimization problems:

Assumption 3.4 (Bounded Problems). For any parameter value $p \in [0, 1]$ (3.1) is bounded, i.e., it is either infeasible or has a finite optimal objective value. The same holds for the LP-relaxation of (3.1).

Assumption 3.5 (Data are Rational Functions of the Parameter). The data (matrix, cost vector, right hand side vector) are continuous rational functions of the parameter $p \in [0, 1]$, i.e., quotients of polynomial functions with nonzero denominator.

Note that an extension to other functional forms is possible.

Remark 3.1. The parameter space P can be divided into a finite number of intervals or segments, such that for each interval the problem is either infeasible or the optimal solution is also a rational function in the parameter p corresponding to a constant basis. At the changes of optimal basis, discontinuity may be observed. This is a consequence of the finite number of integer realization and bases and the assumption that the data are continuous functions of the parameters. In case that one constraint is a linear combination of other constraints $\forall p \in [0, 1]$ this constraint can be eliminated altogether. Otherwise for each parameter value there exists an optimal basis, since constraints that are linearly dependent for some parameter values can be eliminated for these values. Some optimality intervals are degenerate (singletons), but there are

also others of finite length. The marginal costs \bar{c} in the LP case are also rational functions.

Recall that in the general case no convexity or concavity properties exist for the optimal objective value as a function of the parameter. For parametric linear programs Dinkelbach [91, p. 118] claims that if the LP is feasible for all parameter values, the optimal objective value is a continuous function of the parameter. As Example 3.1 shows, additional assumptions are needed to ensure this.

Example 3.1 (Linear dependence for a single parameter value can lead to discontinuity). The parametric LP

$$\begin{aligned} \min_x \quad & -x \\ \text{s.t.} \quad & (p - 0.5)x = 0 \\ & x \in [0, 1] \end{aligned} \tag{3.18}$$

is not continuous at $p = 1/2$. Indeed, for $p \neq 1/2$ we obtain the unique, parameter-independent optimal solution $x = 0$ and the optimal objective function 0. For $p = 1/2$ there are infinitely many feasible points $x \in [0, 1]$ and the optimal solution is $x = 1$ with an objective value of -1 .

The reason for the discontinuity in Example 3.1 is that an equation becomes linearly dependent and redundant and the point-to-set mapping from the parameter space to the feasible space is not closed. A discussion of point-to-set mappings is outside of the scope of this thesis. The reader is referred to Bank et al. [35] for a discussion and for the implications on parametric optimization. Note only that for all parameter values for which (3.1) is feasible, the optimal objective value is lower semi-continuous. To deal with linearly dependent equations we introduce surplus and slack variables and then make an assumption on the augmented systems.

3.5.2 Parametric Linear Program

Consider the following parametric linear program

$$\begin{aligned}
 & \min_{\mathbf{x}} \mathbf{c}(p) \mathbf{x} \\
 & \text{s.t. } \mathbf{A}^1(p) \mathbf{x} = \mathbf{b}^1(p) \\
 & \quad \mathbf{A}^2(p) \mathbf{x} \leq \mathbf{b}^2(p) \\
 & \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP},
 \end{aligned} \tag{3.19}$$

where the data $\mathbf{c}(p) \in \mathbb{R}^n$, $\mathbf{A}^1(p) \in \mathbb{R}^{m_1 \times n}$, $\mathbf{A}^2(p) \in \mathbb{R}^{m_2 \times n}$, $\mathbf{b}^1(p) \in \mathbb{R}^{m_1}$, and $\mathbf{b}^2(p) \in \mathbb{R}^{m_2}$ are assumed to be continuous rational functions of the parameter $p \in [p_l, p_u]$. Note that we deviate from LP in standard form, to show how inequalities should be treated efficiently.

Algorithm 3.5 gives a high-level description of the algorithm. We then describe the mathematical subproblems needed and present two alternatives for these subproblems. The first is based on operations with rational functions and the second is based on continuation methods. As discussed later on, there are significant problems with error propagation of rational operations, particularly for increasing problem size.

Both our proposals are inspired by the algorithm by Dinkelbach [91], who essentially extends the full tableau implementation of the simplex method to the parametric case. As such it is cumbersome to implement, cannot take advantage of state-of-the-art LP solvers and requires a large number of operations with rational functions. We instead propose to solve the linear program (3.19) at the breakpoints (for fixed parameter values), and move from one breakpoint to the next by only considering the basis matrix along with the feasibility and optimality conditions. As a consequence the worst-case number of consecutive operations with rational functions is much smaller and this makes our proposal less vulnerable to error propagation. Essentially our proposal requires the inversion of a square matrix of size $m_1 + m_2$ and therefore at most of the order of $(m_1 + m_2)^3$ operations. The simplex method on the other hand can take an exponential number of iterations and operates on the entire matrices ($\mathbf{A}^1(p)$

and $\mathbf{A}^2(p)$) and error propagates throughout the iterations. For large problems our proposal is therefore expected to be significantly more robust than the algorithm by Dinkelbach.

Algorithm 3.5 (Parametric Linear Program).

Input to the algorithm are the parameter range p_l, p_u , tolerances for violation of the primal inequalities ε_{inf} and the marginal cost inequalities ε_{opt} , and a guess for the minimal parameter step δp . The algorithm uses a set R to store the optimal solutions. The elements R_i of R are quadruplets, composed of parameter values p^{R_i} , a boolean g^{R_i} , describing whether the problem is feasible for this element ($g^{R_i} = \text{true}$) or not ($g^{R_i} = \text{false}$), a point $\mathbf{x}^{R_i}(p)$, and the corresponding objective function $f^{R_i}(p)$.

1. Initialize with $p_s = p_l$.
2. **REPEAT**
 - (a) Solve LP (3.19) for $p = p_s + \delta p$ and obtain an optimal basis.
 - (b) **IF** infeasible **THEN** solve (3.24) for $p = \bar{p} = p_s + \delta p$ and obtain an optimal basis.
 - (c) Set up the parametric system of equations and inequalities.
 - (d) Obtain the parametric dependence of the solution for $p \in [p_s, p_u]$.
 - (e) Get the feasibility range in $p \in [p_s, p_u]$
Set p_t equal to the lowest parameter value for which a primal constraint is violated.
IF $p_t \leq p_s + \delta p$ **THEN** Set $\delta p = \delta p/2$. **GOTO** Step 2a.
Set $p_t = \min(p_t, p_u)$.
 - (f) Get the optimality range in $p \in [p_s, p_t]$
Set p_t equal to the lowest parameter value for which a marginal value constraint is violated.
IF $p_t \leq p_s + \delta p$ **THEN** Set $\delta p = \delta p/2$. **GOTO** Step 2a.
Set $p_t = \min(p_t, p_u)$.

(g) Store $p_s, \mathbf{x}(p), f(p)$ in R .

(h) Set $p_s = \min(p_t, p_u)$.

UNTIL $p_s \geq p_u$

At termination R contains the solution to the parametric mixed-integer program. For all elements R_l such that $g^{R_l} = \text{false}$ the program is infeasible for $p \in [p^{R_l}, p^{R_{l+1}}]$ while for $g^{R_l} = \text{true}$ the point $\mathbf{x}^{R_l}(p)$ satisfies the primal constraints within ε_{inf} -tolerance and the marginal cost constraints within ε_{opt} -tolerance. By convention $p^{R_{l+1}} = p_u$ for $l = |R|$.

Solving at a higher parameter value $p = p_s + \delta p$ ensures a change of basis and checking the feasibility and optimality in $p \in [p_s, p_u]$ ensures that the whole parameter range is covered. In the following we describe how to set up the parametric system of equations and check for feasibility range and optimality range. The parameter δp should be set initially to a sufficiently small value to avoid resetting it and in the following discussion we assume for simplicity that this has been done. Note also that between calls to the LP solver, it typically is advantageous to store the basis information and provide an initial basis for the next call.

Qualitative Invariant Solution

When solving a mixed-integer linear program for a specific parameter value \bar{p} , in general the solution is valid only for this parameter value. On the other hand, for all the parameter values for which an optimal solution exists for (3.19), a vertex of the feasible region is optimal. As a consequence it is plausible to define a qualitatively invariant solution as a fixed basis for the resulting linear program. This leads to a square system of parameterized equations that give the functional dependence of the basic variables on the parameter. The nonbasic variables remain fixed at their lower or upper bound.

Obtaining this functional dependence is part of the postoptimal sensitivity analysis. It is a problem in its own right and an important subproblem in algorithms. In the cost vector case it is a trivial task, since neither the matrix nor the right hand

side vectors depend on the parameter. As a consequence the qualitatively invariant solution is simply a parameter independent point. In the right hand side the matrix elements are parameter independent and the basis matrix can be inverted and then multiplied with the right hand side vector. Therefore, if the right hand side is affine in the parameters, the qualitatively invariant solution is given by an affine function of the parameter. In the general case a matrix has to be inverted as a function of the parameter p . This inversion can be, at least in principle, performed by elementary row operations on the matrix \mathbf{B} , e.g., in an LU factorization, with symbolic operations for the matrix elements. If all data are rational functions of the parameter p , so is the candidate optimal solution. Dinkelbach [91] used this idea to propose an algorithm for the general case of linear programs. This approach does not scale to large problems because of exploding numerical error. Note that the use of high or infinite precision arithmetic is prohibitively expensive. The reader is referred to literature on computer algebra for details, e.g., [71, 70, 81, 195]. We propose an alternative based on continuation and numerical solution of a set of equations along with detection of the violation of primal inequalities and marginal cost bounds. Recall that when only one row or one column of the matrix depend on the parameter, an analytical solution is possible based on a formula by Bodewig [51], see for example [117]. Recall also that Freund [115] proposed to obtain post-optimal sensitivity information of parametric linear programs through Taylor series expansions.

Often linear programs have linear independent equality constraints and this is one of the reasons that state-of-the-art LP solvers such as CPLEX [7] introduce surplus variables, similarly to the solution of Phase I problem in the simplex algorithm. At termination, the solvers furnish a set of $m_1 + m_2$ basic variables, some of which are original and some are surplus variables. The nonbasic variables are at either their lower ($x_j^i = 0$) or upper bound ($x_j^i = x_j^{UP}$). If an equation is infeasible the corresponding surplus variable is basic at a nonzero value. If on the other hand the equation is redundant the corresponding surplus variable is basic but at zero level. It is possible that without redundant constraints, the LP solver returns a surplus

variable as basic at zero level. Note first that (3.19) is equivalent to

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{s}} \mathbf{c}(p) \mathbf{x} \\
& \text{s.t. } \mathbf{A}^1(p) \mathbf{x} + \mathbf{I}^1 \mathbf{s} = \mathbf{b}^1(p) \\
& \quad \mathbf{A}^2(p) \mathbf{x} + \mathbf{I}^2 \mathbf{t} = \mathbf{b}^2(p) \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \quad \mathbf{s} \in \mathbb{R}^{m_1}, \quad \mathbf{s} = \mathbf{0} \\
& \quad \mathbf{t} \in \mathbb{R}^{m_2}, \quad \mathbf{0} \leq \mathbf{t},
\end{aligned} \tag{3.20}$$

where \mathbf{I}^1 and \mathbf{I}^2 are identity matrices of size m_1 and m_2 respectively. Clearly an optimal solution to (3.19) is optimal in (3.20) and vice-versa.

Suppose that at termination the LP solver returns two integer vectors $\mathbf{d}^v \in \{0, 1, 2\}^n$ and $\mathbf{d}^r \in \{0, 1\}^{m_1+m_2}$. The j th component of \mathbf{d}^v indicates if variable j is at its lower bound ($d_j^v = 0$), basic ($d_j^v = 1$), or at its upper bound ($d_j^v = 2$). The j th component of \mathbf{d}^r indicates if artificial variable j is nonbasic ($d_j^r = 0$), or basic ($d_j^r = 1$). Subroutine 3.4 describes how to set up the square system of equations storing a parameter dependence matrix $\mathbf{B}(p)$ and a parameter dependent right hand side vector $\mathbf{b}(p)$. Solving the system $\mathbf{B}(p)\mathbf{x}^B = \mathbf{b}(p)$ as a function of the parameter gives the desired functional dependence. For later use also the upper bounds of the basic variables are stored in $\mathbf{x}^{B,UP}$ and the cost coefficients of the basic variables are stored in $\mathbf{c}^B(p)$.

Subroutine 3.4 (Setting up Parametric System of Equations). The subroutine uses counters i, j and stores the number of basic variables in n_b .

1. Set $b_i(p) = b_i^1(p)$ for $i = 1, \dots, m_1$.
2. Set $b_{i+m_1}(p) = b_i^2(p)$ for $i = 1, \dots, m_2$.
3. Set $n_b = 0$.
4. **FOR** $i = 1, \dots, n$ **DO**
 - **IF** $d_i^v = 1$ **THEN**

- $n_b = n_b + 1$.
- Set $B_{j,n_b}(p) = A_{j,i}^1(p)$ for $j = 1, \dots, m_1$.
- Set $B_{j+m_1,n_b}(p) = A_{j,i}^2(p)$ for $j = 1, \dots, m_2$.
- Set $c_{n_b}^B(p) = c_i(p)$.
- Set $x_{n_b}^{B,UP} = x_i^{UP}$.
- **ELSE IF** $d_i^v = 2$ **THEN**
 - Set $b_j(p) = b_j(p) - A_{j,i}^1(p)x_i^{UP}$ for $j = 1, \dots, m_1$.
 - Set $b_{j+m_1}(p) = b_j(p) - A_{j,i}^2(p)x_i^{UP}$ for $j = 1, \dots, m_2$.

END

5. **FOR** $i = 1, \dots, m_1 + m_2$ **DO**

- **IF** $d_i^r = 1$ **THEN**
 - $n_b = n_b + 1$.
 - Set $B_{i,n_b}(p) = 1$.
 - Set $B_{j,n_b}(p) = 0$ for $j \neq i$.
 - **IF** $i \leq m_1$ **THEN** Set $x_{n_b}^{B,UP} = 0$ **ELSE** $x_{n_b}^{B,UP} = +\infty$.
 - Set $c_{n_b}^B(p) = 0$.

END

As demonstrated in Example 3.1 there are problems when $\mathbf{B}(p)$ becomes singular at some parameter values, because it is no longer a basis and the primal and marginal constraints do not guarantee optimality. To ensure that $\mathbf{B}(p)$ does not become singular during the iterations of Algorithm 3.5 we make an additional assumption.

Assumption 3.6. If an augmented basis $\mathbf{B}(p)$ is optimal for some \bar{p} and singular for some other parameter value $\tilde{p} \in P$ then there exists $\tilde{\varepsilon} > 0$, such that for

$$\forall p \in P : |\tilde{p} - p| \leq \tilde{\varepsilon}$$

the system $\mathbf{B}(p)\mathbf{x}^B = \mathbf{b}(p)$ does not have any solutions satisfying the primal constraints $\mathbf{x}^B \in [\mathbf{0}, \mathbf{x}^{B,UP}]$.

Solution with Rational Operations

The algorithm by Dinkelbach [91] solves the parametric linear program directly using rational operations. As a consequence at each iteration $(\mathbf{B}(p))^{-1}\mathbf{b}(p)$ is available. We propose instead to take the matrix and perform an LU factorization with rational operations. We assume the existence of an LU-factorization algorithm of the rational matrix $\mathbf{B}(p)$ obtaining $\mathbf{P}\mathbf{B}(p) = \mathbf{L}(p)\mathbf{U}(p)$, where \mathbf{P} is a permutation matrix, and \mathbf{L} , \mathbf{U} are respectively lower- and upper-triangular matrix with rational entries; the diagonal entries of \mathbf{L} are all equal to 1.

An advantage of this method is that we can directly check if the matrix becomes singular for some parameter point. This could be done by calculating the determinant

$$|\mathbf{B}(p)| = \prod_{i=1}^m \mathbf{U}_{i,i}(p)$$

and then solving for the first root of $|\mathbf{B}(p)|$. So in principle Assumption 3.6 is not needed if rational operations are performed.

Once the range for which the basis is invertible has been identified, the system $\mathbf{B}(p)\mathbf{x}^B = \mathbf{b}(p)$ can be solved in two steps. First forward elimination

$$\mathbf{L}(p)\mathbf{v} = \mathbf{P}\mathbf{b}(p),$$

is performed giving the temporary vector \mathbf{v} as a function of the parameter. Then by back substitution

$$\mathbf{U}(p)\mathbf{x}_B = \mathbf{v}(p)$$

the dependence of \mathbf{x}_B on the parameter is obtained.

Remark 3.2. Note that since the surplus and slack variables have only one nonzero entry, it is more efficient to reorder the variables and rows, putting the artificial variables in the upper left corner. For simplicity we did not describe this in Subroutine 3.4.

Solution with Continuation

A problem with the matrix inversion with rational operations, is that either multi-precision arithmetic has to be used, which is prohibitively expensive, or the numerical error explodes with increasing system size. We therefore propose an alternative approach based on continuation. This method avoids operations with rational functions and therefore scales to large systems. Note that even the calculation of the right hand side in Subroutine 3.4 can be done within the continuation method.

We here give a brief introduction to continuation. The reader is referred to [251, 22] for a thorough discussion. The general idea of continuation [251] is to follow an implicitly defined curve

$$\mathbf{f}(\mathbf{z}, \lambda) = \mathbf{0},$$

where $\mathbf{f} : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ is typically assumed smooth. Predictor-corrector methods [22] trace the curve by generating a sequence of points λ^i, \mathbf{z}^i that are on the curve within a given tolerance ($\|\mathbf{f}(\mathbf{z}, \lambda)\| \leq \varepsilon$). At a given iteration a step is taken from λ^{i-1} to λ^i and a predictor $\bar{\mathbf{z}}^i$ is calculated, e.g., by a polynomial spline approximation. The correction is typically performed by a Newton solver for the solution of $\mathbf{f}(\mathbf{z}^i, \lambda^i)$ with $\bar{\mathbf{z}}^i$ as the initial guess. The step size $\lambda^i - \lambda^{i-1}$ depends on the quality of approximation of the previous step(s).

Our proposal should not be confused with homotopy continuation, in which case only the final point (for $\lambda = 1$) of the curve is needed. We are interested in actually obtaining an estimate to the solution of

$$\mathbf{B}(p) \mathbf{x}_B = \mathbf{b}(p) \tag{3.21}$$

for a range of parameter values. We will use the predictor polynomials as an approximation to the solution. Therefore estimates are needed for the predictor quality at the generated points, as well as for the points in-between. While it is possible to have rigorous guarantees, see, e.g., Neubert [209], most methods rely on error estimates [238, 239]. Note that similar error estimates are successfully applied to the solution of differential-algebraic equation systems [29].

For a fixed parameter value a linear system of equations has to be solved. Therefore the corrector step can be performed by any linear solver (direct or iterative). Assuming that the predictor is a good approximation, it may be beneficial to use an iterative method, especially for large systems.

Because no direct inversion of the basis $\mathbf{B}(p)$ is performed, the algorithm does not necessarily detect parameter values for which the matrix is singular. For Example 3.1 the discontinuity would not be detected but Assumption 3.6 ensures that a primal constraint would be violated before the matrix becomes singular. It is possible to write formulations that would check the singularity, but they are overly expensive.

Feasibility Range

The feasibility range of a given basis and the corresponding basic solution $\mathbf{x}^B(p)$ is implicitly defined by

$$\begin{aligned} \mathbf{B}(p)\mathbf{x}^B &= \mathbf{b}(p) \\ \mathbf{0} &\leq \mathbf{x}^B \leq \mathbf{x}^{B,UP}(p). \end{aligned} \tag{3.22}$$

In the general case this defines a disjoint set for the parameter. Recall that in the cost vector case this step is not needed, since feasibility does not depend on the parameter. In the right hand side case with affine dependence on the parameter \mathbf{x}^B is an affine function of the parameter and a set of linear inequalities needs to be checked for. In Algorithm 3.5 we have established feasibility for a parameter point p_s and we want to find the smallest parameter value p_t for which (3.22) is violated by a prespecified tolerance ε_{inf} .

Solution with Rational Operations

Recall that with the use of forward elimination and back substitution we have obtained the functional dependence of \mathbf{x}^B on the parameter. Subroutine 3.5 determines the smallest parameter value p_t for which (3.22) is violated by a prespecified tolerance ε_{inf} .

Subroutine 3.5 (Obtain Feasibility Range).

1. Set $p_t = p_u$.
2. **FOR** $i = 1, \dots, m_1 + m_2$ **DO**
 - Set p_t equal to the smallest root of $x_i^B(p) = -\varepsilon_{inf}$ for $p \in [p_s, p_t]$.
 - **IF** $x_i^{B,UP} < +\infty$ **THEN** set p_t equal to the smallest root of $x_i^B(p) = x_i^{B,UP} + \varepsilon_{inf}$ for $p \in [p_s, p_t]$.

END

Solution with Continuation

In addition to tracking the solution of (3.21) we propose to identify ε_{inf} -violation of the primal constraints $\mathbf{0} \leq \mathbf{x}^B \leq \mathbf{x}^{B,UP}(p)$ by an event detection algorithm such as the one proposed by Park and Barton [221] for hybrid differential-algebraic equation systems. This algorithm first identifies an event by solving the interpolating polynomials within a continuation step and then accurately locates the event. Here once a violation is identified the continuation code exits.

Optimality Range

In linear programs explicit optimality conditions are available [42] and can be used to identify the range of optimality. Recall the meaning of the basis matrix $\mathbf{B}(p)$, cost vector of the basic solutions \mathbf{c}^B and the integer vector \mathbf{d}^v from Subroutine 3.4. The following system implicitly defines the range of optimality of a basis

$$\begin{aligned}
(\mathbf{B}(p))^T \mathbf{z} &= \mathbf{c}^B(p) \\
\bar{c}_j(p) &= c_j(p) - \sum_{i=1}^{m_1} z_i(p) \mathbf{A}_{i,j}^1(p) - \sum_{i=1}^{m_2} z_i(p) \mathbf{A}_{i+m_1,j}^2(p) \quad \forall j : d_j^v \neq 1 \\
\bar{c}_j(p) &\geq 0 \quad \forall j : d_j^v = 0 \\
\bar{c}_j(p) &\leq 0 \quad \forall j : d_j^v = 2 \\
z_j &\leq 0 \quad \forall j \leq m_1 : d_j^r = 0,
\end{aligned} \tag{3.23}$$

where $\bar{\mathbf{c}}$ is the vector of marginal costs and \mathbf{z} is a vector of auxiliary variables. Note that the marginal cost for the nonbasic slack variables can easily be calculated since they are equal to the auxiliary variables. Note also that the marginal cost of surplus variables need not be calculated.

Remark 3.3. Since the cost of the surplus variables is set equal to zero and the columns corresponding to the surplus variables are equal to the identity vector we can eliminate the \mathbf{z} -variables corresponding to the basic surplus variables from (3.23). These steps are omitted for the sake of simplicity.

Similarly to the feasibility range, in general (3.23) defines a disjoint set for the parameter. In the right-hand side case this step is not needed, since the marginal costs do not depend on the parameter. In the cost vector case with affine dependence on the parameter, a set of linear inequalities needs to be checked for. In Algorithm 3.5 we have established the optimality conditions for a parameter point p_s and we want to find the smallest parameter value p_t for which (3.23) is violated by a prespecified tolerance ε_{opt} .

Solution with Rational Operations

Recall that we have performed an LU factorization of the matrix $\mathbf{B}(p)$. We can use this to first calculate the auxiliary variables \mathbf{z} and then the marginal costs, as shown in Subroutine 3.6.

Subroutine 3.6 (Obtain Optimality Range).

1. Solve $\mathbf{U}^T(p) \mathbf{u}(p) = \mathbf{c}_B(p)$ by forward elimination.
2. Solve $\mathbf{L}^T(p) \mathbf{v}(p) = \mathbf{u}(p)$ by back-substitution.
3. Calculate $\mathbf{z}(p) = \mathbf{P}^T \mathbf{u}(p)$.
4. **FOR** $j = 1, \dots, m_1$
 - **IF** $d_j^r = 0$ **THEN** Set p_t equal to the first root of $z_j(p) = -\varepsilon_{opt}$ in $p \in [p_s, p_t]$.

END

5. **FOR** $j = 1, \dots, n$, $d_j^v \neq 1$

- Set: $t(p) = c_j(p) - \sum_{i=1}^{m_1} z_i(p) \mathbf{A}_{i,j}^1(p) - \sum_{i=1}^{m_2} z_i(p) \mathbf{A}_{i+m_1,j}^2(p)$.
- **IF** $d_j^v = 0$ **THEN** Set p_t equal to the first root of $t(p) = -\epsilon_{opt}$ in $p \in [p_s, p_t]$
ELSE Set p_t equal to the first root of $t(p) = \epsilon_{opt}$ in $p \in [p_s, p_t]$.

END

Solution with Continuation

We propose to identify violation of the marginal cost inequalities in a similar way as the violation of primal inequalities, i.e., solve

$$(\mathbf{B}(p))^T \mathbf{z} = \mathbf{c}^B(p)$$

using a continuation method until one of the following inequalities is violated

$$\begin{aligned} c_j(p) - \sum_{i=1}^{m_1} z_i(p) \mathbf{A}_{i,j}^1(p) - \sum_{i=1}^{m_2} z_i(p) \mathbf{A}_{i+m_1,j}^2(p) &\geq 0, & \forall j \in \{1, \dots, m_1 + m_2\} : d_j^v = 0 \\ c_j(p) - \sum_{i=1}^{m_1} z_i(p) \mathbf{A}_{i,j}^1(p) - \sum_{i=1}^{m_2} z_i(p) \mathbf{A}_{i+m_1,j}^2(p) &\leq 0, & \forall j \in \{1, \dots, m_1 + m_2\} : d_j^v = 2 \\ z_j &\leq 0, & \forall j \in \{1, \dots, m_1\} : d_j^x = 0. \end{aligned}$$

by a value equal to ϵ_{opt} .

Dealing with Infeasibility

For parameter values for which (3.19) is infeasible, an auxiliary problem needs to be solved where surplus variables are introduced. It corresponds to the Phase I of the

simplex method [42].

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{s}} \quad & \sum_{j=1}^{m_1+m_2} s_j \\
\text{s.t.} \quad & \sum_{j=1}^{n_x} A_{i,j}^1(p)x_j + \text{sign}(b_i(\bar{p}))s_i = b_i^1(p), \quad i = 1, \dots, m_1 \\
& \sum_{j=1}^{n_x} A_{i,j}^2(p)x_j - s_{i+m_1} = b_i^2(p), \quad i = 1, \dots, m_2 \\
& \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{0} \leq \mathbf{x} \leq \mathbf{x}^{UP} \\
& \mathbf{s} \in \mathbb{R}^{m_1+m_2}, \quad \mathbf{0} \leq \mathbf{s}.
\end{aligned} \tag{3.24}$$

By construction (3.24) is feasible for $p = \bar{p}$.

The solution of (3.24) gives an optimal basis, which can be treated as a basis for the feasibility and LP-optimality range. To deal with the surplus variables that LP solvers introduce we define a vector \mathbf{e} such that $e_i = 1$ if $s_i^* > 0$ and $e_i = 0$ otherwise. Then we base the solution of optimality and feasibility range problem on the following problem

$$\begin{aligned}
\min_{\mathbf{x}} \quad & \mathbf{e}^T \mathbf{s} \\
\text{s.t.} \quad & \mathbf{A}(p) \mathbf{x} + \mathbf{I} \mathbf{s} \leq \mathbf{b}(p) \\
& \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U \\
& 0 \leq s_i \leq 0, \quad \forall i \in \{1, \dots, m_1 + m_2\} : e_i = 0 \\
& 0 \leq s_i, \quad \forall i \in \{1, \dots, m_1 + m_2\} : e_i = 1.
\end{aligned} \tag{3.25}$$

Termination of algorithms

Due to continuity of the data and the fact that we allow ε_{inf} -violation of the primal inequalities and ε_{opt} -violation marginal costs at each iteration Algorithm 3.5 takes a finite step in the parameter interval $[p_l, p_u]$. Suppose that we have solved for $p = p_0$ and obtained a basis. Since the matrix elements are continuous and all entries are bounded, the determinant of the basis is also continuous. By the definition of a basis,

its determinant is nonzero for $p = p_0$. We can thus determine a parameter range $P^1 = [p_0, p_1]$ with $p_0 < p_1$ for which the determinant remains at nonzero value and therefore is invertible. The basic variables and marginal costs are also rational and continuous within P^1 . As a consequence we can find a parameter range $P^2 = [p_0, p_2]$ with $p_0 < p_2$ for which the basis does not violate the primal and marginal cost inequalities by more than the specified tolerances. At termination the parameter range p_l, p_u is divided into intervals in which a given basis satisfies the primal and marginal cost inequalities within the prespecified tolerances.

If one does not allow the ε -violation of the primal and dual constraints, degeneracy or linear dependence of equality constraints can result in one solution being optimal for a single parameter value. In that case the algorithms need to be slightly modified and a more elaborate argument for finite termination is needed. The use of finite precision arithmetic seems to make the use of the tolerances necessary. Also violation of the constraints seems acceptable since LP solvers typically give a solution within some tolerances. Obviously the tolerances used in our algorithms need to be larger than the ones used by the LP solvers.

3.5.3 Parametric Mixed-Integer Linear Program

Optimality-Region Algorithm for Parametric MILP

In addition to the assumptions on the availability of an LP solver, we will assume the existence of a MILP and a MINLP solver that either proves infeasibility or gives an integer optimal solution point. The general structure of the algorithm is the same as for the algorithms described for the parametric LP. Note that state-of-the-art MINLP solvers require bounded variables [266, 247], and therefore Algorithm 3.6 is only applicable to problems with finite \mathbf{x}^{UP} .

Input to the algorithm are the tolerances for violation of the primal inequalities ε_{inf} and the marginal cost inequalities ε_{opt} and a guess for the minimal parameter step δp . The algorithm uses a set R to store the optimal solutions. The elements R_l of R are quadruplets, composed of parameter values p^{R_l} , a boolean g^{R_l} , describing

whether the problem is feasible for this element ($g^{R_i} = \text{true}$) or not ($g^{R_i} = \text{false}$), a point $\mathbf{x}^{R_i}(p)$, and the corresponding objective function $f^{R_i}(p)$.

Algorithm 3.6 (Parametric Mixed Integer Linear Program via Optimality Region).

1. Initialize with $p_s = 0$.

2. **REPEAT**

(a) Solve MILP (3.1) for $p = p_s + \delta p$.

IF feasible **THEN**

i. Fix integer variables

ii. Solve LP (3.19) for $p = p_s + \delta p$ and obtain an optimal basis.

iii. Set up the parametric system of equations and inequalities.

iv. Obtain the parametric dependence of the solution $\mathbf{x}(p)$ for $p \in [p_s, p_u]$.

v. Get the feasibility range in $p \in [p_s, p_u]$

Set p_t equal to the lowest parameter value for which a primal constraint is violated.

IF $p_t \leq p_s + \delta p$ **THEN** Set $\delta p = \delta p/2$. **GOTO** Step 2(a)ii.

Set $p_t = \min(p_t, p_u)$.

vi. **Optional LP Optimality Step**

Get the optimality range in $p \in [p_s, p_t]$

Set p_t equal to the lowest parameter value for which a marginal value constraint is violated.

IF $p_t \leq p_s + \delta p$ **THEN** Set $\delta p = \delta p/2$. **GOTO** Step 2(a)ii.

Set $p_t = \min(p_t, p_u)$.

vii. Solve MINLP (3.4) with $g(p) = p$ for $p \in [p_s, p_t]$ and obtain optimal objective value p^*

IF feasible **THEN** $p_t = p^*$.

viii. **IF** $p_t \leq p_s + \delta p$ **THEN** Set $\delta p = \delta p/2$. **GOTO** Step 2a.

ix. Store $g = 1, p_s, \mathbf{x}(p), f(p)$ in R .

ELSE

- i. Solve MINLP (3.5) with $g(p) = p$ for $p \in [p_s, p_t]$ and obtain optimal objective value p^*
IF feasible **THEN** $p_t = p^*$.
- ii. **IF** $p_t \leq p_s + \delta p$ **THEN** Set $\delta p = \delta p/2$. **GOTO** Step 2a.
- iii. Store $p_s, g = 0, \mathbf{x}(p) = \mathbf{0}, f(p) = +\infty$ in R .

END

- (b) Set $p_s = \min(p_t, 1)$.

UNTIL $p_s \geq 1$

At termination R contains the solution to the parametric mixed-integer program. For all elements R_l such that $g^{R_l} = \text{false}$ the program is infeasible for $p \in [p^{R_l}, p^{R_{l+1}}]$ while for $g^{R_l} = \text{true}$, the point $\mathbf{x}^{R_l}(p)$ satisfies the primal constraints within ε_{inf} -tolerance and the marginal cost constraints within ε_{opt} -tolerance. By convention $p^{R_{l+1}} = p_u$ for $l = |R|$.

Similarly to Algorithm 3.5, finite termination of the algorithm is guaranteed due to the use of the tolerances ε_{inf} , ε_{opt} and ε , in the primal conditions, dual conditions and optimality region formulation respectively, which essentially overestimate the true bounds of the optimality range. As a consequence at each iteration there is a finite increase in the parameter value p_{cur} . Another consequence is that due to the overestimation the next MILP call will result in a new solution.

Branch-and-Bound Algorithm for Parametric MILP

Branch-and-bound (B&B) algorithms for parametric mixed-integer problems mimic the B&B algorithm for a regular MILP, by branching on the integer variables. Ohtake and Nishida [216] and later Acevedo and Pistikopoulos have applied this idea to the right hand side case [11]. In Section 3.4.5 we used B&B for the multiparametric cost vector case.

Each node corresponds to a parametric MILP with some binary variables fixed.

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}} \mathbf{c}_x^T(p) \mathbf{x} + \mathbf{c}_y^T(p) \mathbf{y} \\
& \text{s.t. } \mathbf{A}_{1x}(p) \mathbf{x} + \mathbf{A}_{1y}(p) \mathbf{y} = \mathbf{b}_1(p) \\
& \quad \mathbf{A}_{2x}(p) \mathbf{x} + \mathbf{A}_{2y}(p) \mathbf{y} \leq \mathbf{b}_2(p) \\
& \quad \mathbf{x} \in \mathbb{R}^{n_x}, \quad \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U \\
& \quad \mathbf{y}_j = 0, \quad j \in Z^i \\
& \quad \mathbf{y}_j = 1, \quad j \in O^i \\
& \quad \mathbf{y}_j \in \{0, 1\}, \quad j \in \{1, \dots, n_y\}, \quad j \notin Z^i \cup O^i \\
& \quad p \in [0, 1],
\end{aligned} \tag{3.26}$$

for some index sets Z^i, O^i . A lower bound can be obtained by the parametric solution to the LP-relaxation of (3.26), i.e., a problem of the form (3.19). Out of the solutions obtained over the parameter range some may be binary feasible and this provides an upper bound.

The upper bound is stored in the set R^0 . The elements R_l^0 of R^0 are quadruplets, composed of parameter values $p^{R_l^0}$, a boolean $g^{R_l^0}$, describing whether an upper bound exists for this element ($g^{R_l^0} = \text{true}$) or not ($g^{R_l^0} = \text{false}$), a point $(\mathbf{x}^{R_l^0}(p), \mathbf{y}^{R_l^0})$, and the corresponding objective function $f^{R_l^0}(p)$. The elements are ordered according to $p^{R_l^0}$, and by convention $p^{R_{l+1}^0} = 1$ for $l = |R^0|$. For $g^{R_l^0} = \text{true}$, the point $\mathbf{x}^{R_l^0}(p), \mathbf{y}^{R_l^0}$ is ε_{inf} -feasible for the parametric MILP for $p \in [p^{R_l^0}, p^{R_{l+1}^0}]$ and $f^{R_l^0}(p)$ is an upper bound to the optimal objective function in that interval.

Let the index set A contain the indices i of the currently active nodes, each corresponding to a parametric MILP ^{i} (3.26) and each associated with index sets Z^i, O^i . Also associated with each node i is a set R^i , with the same elements as R^0 . The boolean g^{R^i} describes if the element is active ($g^{R^i} = \text{true}$), i.e., branching needs to be performed, or inactive ($g^{R^i} = \text{false}$), i.e., no further branching is needed. An element can become inactive by an infeasible lower bounding problem or by value dominance. The point $(\mathbf{x}^{R^i}(p), \mathbf{y}^{R^i})$ is used to store optimal solutions to the LP-relaxation of

MILPⁱ for $p \in [p^{R_i}, p^{R_{i+1}}]$ and $f^{R_i}(p)$ for the corresponding objective function.

Algorithm 3.7 (Branch-and-Bound Algorithm for a General Parametric MILP).

1. **(Initialization)**

$$A = \{1\}$$

$$Z^1 = O^1 = \emptyset$$

$$p^{R_1} = 0, g^{R_1} = \text{true}, f^{R_1} = -\infty$$

$$p^{R_1^0} = 0, g^{R_1^0} = \text{false}, f^{R_1^0} = \infty$$

$$k = 1.$$

2. **(Termination Test)**

If $A = \emptyset$ then terminate.

3. **(Node Selection)**

Select and delete a node i from A .

4. **(Comparison of lower and upper bound)**

CALL *Check Lower Bound* (Subroutine 3.7)

5. **(Relaxation)**

$\forall R_i^l \in R^i$: $g^{i,l} = \text{true}$ solve the LP relaxation of node i and replace R^i with the answer of the parametric LP.

6. **(Update of upper bound and fathoming)**

CALL *Update Upper Bound* (Subroutine 3.8)

7. **(Branching)**

IF $g^{R_i} = \text{true}$ for some l **THEN**

- Select a free binary variable $j \notin Z^i \cup O^i$
- Create subproblems with $Z^{k+1} = Z^i \cap j$, $O^{k+1} = O^i$ and $Z^{k+2} = Z^i$,
 $O^{k+2} = O^i \cap j$.
- Set $R^{k+1} = R^i$ and $R^{k+2} = R^i$.

- Add nodes $k + 1$ and $k + 2$ to A .
- Set $k = k + 2$.

8. GOTO 2.

At termination R^0 contains the solution to the parametric mixed-integer program. For all elements R_l^0 such that $g^{R_l^0} = \text{false}$ the program is infeasible for $p \in [p^{R_l^0}, p^{R_{l+1}^0}]$ while for $g^{R_l^0} = \text{true}$, the point $(\mathbf{x}^{R_l^0}(p), \mathbf{y}^{R_l^0})$ is an optimal solution for $p \in [p^{R_l^0}, p^{R_{l+1}^0}]$ with the objective function $f^{R_l^0}(p)$. Since we have a finite number of integer variables the algorithm terminates finitely; in the worst case $2^{n_y+1} - 1$ nodes are visited.

Subroutine 3.7 (Check Lower Bound (integer i)).

- **FOR** $l = 1, \dots, |R^i|$
 - **IF** $p^{R_{l+1}^i} < p^{R_l^i}$ **THEN** split R_l^i at $p^{R_{l+1}^i}$
 - **ELSIF** $p^{R_{l+1}^i} > p^{R_l^i}$ **THEN** split R_l^i at $p^{R_{l+1}^i}$
- END**
- **FOR** $l = 1, \dots, |R^i|$
 - Set p_t equal to first root of $f^{R_l^i}(p) = f^{R_{l+1}^i}(p)$ in $[p^{i,l}, p^{i,l+1}]$.
 - **IF** $p_t < p^{i,l+1}$ **THEN**
 - * Split R_l^i at p_t .
 - * Split R_{l+1}^i at p_t .
 - **IF** $f^{R_l^i}(p) \geq f^{R_{l+1}^i}(p)$ **THEN** set $g^{R_l^i} = 0$.
- END**
- **FOR** $l = 1, \dots, |R^i|$
 - **IF** $g^{R_l^i} = g^{R_{l+1}^i}$ **THEN** merge³ R_l^i and R_{l+1}^i

³The purpose of this merging step is to minimize the number of LP calls required for the solution of the parametric LP. Whether or not to use this step is a heuristic, because if information of the parent node is used for the solution of the child node, merging the intervals may be detrimental for the computational speed.

END

Subroutine 3.8 (Update Upper Bound (integer i)).

1. **FOR** $l = 1, \dots, |R^i|$

- **IF** $p^{R_{l+1}^i} < p^{R_{l+1}^0}$ **THEN** split R_l^0 at $p^{R_{l+1}^i}$.
- ELSIF** $p^{R_{l+1}^i} > p^{R_{l+1}^0}$ **THEN** split R_l^i at $p^{R_{l+1}^0}$.

END

2. **FOR** $l = 1, \dots, |R^0|$: $g^{R_l^i} = 0$

- Set p_t equal to first root of $f^{R_l^i}(p) = f^{R_l^0}(p)$ in $[p^{R_l^0}, p^{R_{l+1}^0}]$.
- **IF** $p_t < p^{R_{l+1}^0}$ **THEN**
 - Split R_l^i at p_t .
 - Split R_l^0 at p_t .
- **IF** $g^{R_l^0} = \text{true}$ and $f^{R_l^i}(p) \geq f^{R_l^0}(p)$ **THEN** set $g^{R_l^i} = \text{false}$.
- **IF** $\mathbf{y}^{R_l^i} \in \{0, 1\}^{n_y}$ and ($g^{R_l^0} = \text{false}$ or $f^{R_l^i}(p) < f^{R_l^0}(p)$)
THEN set $f^{R_l^0}(p) = f^{R_l^i}(p)$, $g^{R_l^0} = \text{true}$ and $g^{R_l^i} = \text{false}$.

END

3. **FOR** $l = 1, \dots, |R^i|$

- **IF** $g^{R_l^i} = g^{R_{l+1}^i} = \text{false}$ **THEN** merge R_l^i and R_{l+1}^i .

END

4. **FOR** $l = 1, \dots, |R^0|$

- **IF** $\left(g^{R_l^0} = g^{R_{l+1}^0} = \text{true} \text{ and } \mathbf{x}^{R_l^0}(p) = \mathbf{x}^{R_{l+1}^0}(p) \text{ and } \mathbf{y}^{R_l^0} = \mathbf{y}^{R_{l+1}^0} \right)$
or $\left(g^{R_l^0} = g^{R_{l+1}^0} = \text{false} \right)$ **THEN** merge R_l^0 and R_{l+1}^0 .

END

Remark 3.4 (Speed improvement by using parent information). In analogy to B&B on a regular MILP, computational gains are possible when the solution to the parent node is used to solve the child node, e.g., by providing an initial basis for the LP-solver. A drawback is an increase in the storage requirements.

3.5.4 Implementation

In this section we briefly discuss some interesting points regarding the implementation.

Polynomials and rationals

Each polynomial $q(p)$ is stored as an array of coefficients a_i ; the array size is equal to the order of the polynomial.⁴ We allow the coefficients to be either a native C real number of double precision, or one of three types defined in the GNU Multiprecision Library [6] namely (i) long precision float, (ii) integer, or (iii) rational. When the absolute value of the leading coefficient is less than a specified tolerance the order of the polynomial is reduced to avoid explosion of the order of the polynomials. Note that the operations are sensitive to this tolerance, and tuning is needed to successfully solve systems of more than a few variables.

Implementation of addition and subtraction of polynomials is straightforward, by respectively adding or subtracting the coefficients and the order of the resulting polynomial is equal to the maximal order of the two polynomials. Multiplication of polynomials $q^3(p) = q^1(p) q^2(p)$ is a little more elaborate; the order of the resulting polynomial $q^3(p)$ is equal to the sum of the orders of the two multiplicands $q^1(p)$, $q^2(p)$ and the i -th coefficient is calculated as $q_i^3 = \sum_{j=0}^i q_j^1 q_{i-j}^2$.

Rationals are stored as the quotient of two polynomials $r(p) = \frac{q^n(p)}{q^d(p)}$ and we also introduce an integer that describes whether a rational is identical to zero. This integer allows to avoid unnecessary operations, by the use of the elementary rules

⁴Currently we use fixed-size arrays for simplicity but also store the order of the polynomial to avoid unnecessary operations.

$0 \pm r^2(p) = r^2(p)$ and $0 \times r^2(p) = 0$. This way we take partial advantage of the sparsity that typical MILPs have. The rational operations needed are defined based on elementary operations on the numerator and denominator polynomials. Consider two rationals $r^1(p) = \frac{q^{n1}(p)}{q^{d1}(p)}$, $r^2(p) = \frac{q^{n2}(p)}{q^{d2}(p)}$, and scalars α^1, α^2 . Multiplication and division of rationals is straightforward

$$\begin{aligned} r^1(p) \times r^2(p) &= \frac{q^{n1}(p) q^{n2}(p)}{q^{d1}(p) q^{d2}(p)} \\ r^1(p)/r^2(p) &= \frac{q^{n1}(p) q^{d2}(p)}{q^{d1}(p) q^{n2}(p)}. \end{aligned}$$

The linear combination of rationals (for the LU factorization) is given as

$$\alpha^1 r^1(p) + \alpha^2 r^2(p) = \frac{\alpha^1 q^{n1}(p) q^{d2}(p) + \alpha^2 q^{n2}(p) q^{d1}(p)}{q^{d1}(p) q^{d2}(p)}.$$

Simplification of the rationals is needed if the numerator and denominator have a common denominator. We have implemented this simplification with the simple Euclidean algorithm [71, 70], that finds the greatest common denominator without resorting to root finding. For the calculation of roots of polynomials we use the Harwell Subroutine pa17bd.

Optimization Subproblems

For LP and MILP at fixed parameter values we call the CPLEX library [7]. For the optimality region formulation as a MINLP we use BARON version 7.4 [247] available in GAMS version 22.0 [56]. The outer approximation algorithm by Kesavan et al. [169] was also tested through an in-house implementation [61] but no significant computational gains were observed.

Continuation

Instead of a continuation code with event detection, we use DSL48SE [270] available through DAEPACK [271], which is an integrator of hybrid differential-algebraic equation systems with state-event detection. As such, it is not the optimal tool, but it was

the most readily available. A general system of residuals is formulated in Fortran and the specific system is defined through parameters. This exploits the system sparsity without the need of recompilation. For simplicity the primal system (3.21) and the system of marginal costs (3.23) are solved together. Since DSL48SE can exploit sparsity and problem structure the overhead of this is very small. Consistent initialization is performed through a call to MA48 from the Harwell Subroutines [5].

3.5.5 Numerical Results

We have formulated some small size parametric LPs and MILPs with interesting theoretical properties as well as two case studies from man-portable power generation. They are all described in the Appendix B. Note that some of the problems have special structure, e.g., parameter affects only one row, so that specialized algorithms could be used. Table 3.1 compares the computational requirements for four alternatives. All optimization tolerances are set to the default value of 10^{-6} . The tolerances for violation of primal feasibility and of the marginal costs is set to 3×10^{-6} . The first alternative, labeled CPLEX, is a discretization of the parameter range with 1000 equidistant points. The second alternative, labeled B&B+Rational, is the B&B algorithm 3.4 with the parametric LP solved by rational operations. Polynomial coefficients with magnitude less than 10^{-10} are considered as zero and the polynomials are simplified. The third alternative, labeled B&B+DSL48, is the B&B algorithm 3.4 with the parametric LP solved by a continuation approach. The DAE integrator DSL48SE is used here with tolerances set to 10^{-6} . The fourth alternative, labeled DSL48+Baron, is the optimality region algorithm 3.6. This alternative is not applied on the examples that do not contain binary variables. Note that the MINLP optimization problems are solved separately from the continuation problems after a polynomial is fitted to the solution.

| Example | B.1 | B.2 | B.3 | B.4 | B.5 | B.6 | B.7 | B.8 | B.9 | B.10 | B.11 |
|--------------|------|------|------|------|------|------|------|------|------|------|------|
| CPLEX | 0.17 | 0.07 | 0.06 | 0.07 | 0.07 | 0.11 | 0.08 | 0.08 | 0.7 | 1.35 | 50 |
| B&B+Rational | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 | 0.08 | - | 0.07 | - |
| B&B+DSL48 | 0.34 | 0.01 | 0.16 | 0.08 | 0.08 | 0.18 | 0.08 | 0.01 | 1.3 | 2.63 | 100 |
| DSL48+Baron | N/A | 0.02 | 0.4 | N/A | N/A | 1.4 | N/A | N/A | 0.25 | 0.23 | 18 |

Table 3.1: Computational requirements in seconds for the parametric LP and MILP. No distinction is done for CPU times less than 0.01s.

3.6 Conclusions

Parametric optimization considers the solution of mathematical programs that depend on parameters for all parameter values within a prespecified range. Algorithms identify qualitatively invariant solutions and their optimality range. Parametric optimization is useful as a tool for resource allocation.

We consider two classes of parametric mixed-integer linear programs for which no algorithms exist in the open literature, namely the general case, where a parameter affects the cost-vector, right-hand side vectors and matrix, and the multiparametric cost vector case. Based on the work by Pertsinidis et al. [224, 225] we formulate a single level optimization problem that identifies the optimality region. In general, this formulation is a mixed-integer nonlinear program with nonconvex functions, but under certain restrictions an exact linear reformulation is possible.

In the cost vector case, the feasible set is parameter independent and the parameters affect the objective affinely. We present the extension of a literature algorithm from the case of one parameter [101, 162, 164] to the multiparametric case. The optimality region of a given solution is a convex polyhedron in the parameter space. Our algorithm identifies the vertices of these polyhedra and calls MILP solvers for these parameter points. The computational requirement to identify the polyhedra vertices is generally small compared with the MILP calls. The number of MILP calls needed is at least equal to the number of vertices, and at most equal to the number of optimal solutions plus the number of vertices. The actual number needed strongly depends on the problem structure, but is at least exponential in the number of parameters. We therefore also propose an alternative, based on the single level formulation for the optimality regions. The number of optimization problems to be solved is equal to the number of optimal solutions. This alternative is most promising for the case that the parameters only affect the cost coefficients of the integer variables, resulting in a (mixed-integer) linear formulation. Finally we describe an algorithm based on branch-and-bound on the integer variables and the solution of parametric linear programs at each node.

In the general case, in which the parameter also affects the matrix, we first consider parametric linear programs and formulate an alternative to the parametric simplex algorithm by Dinkelbach [91]. The algorithm by Dinkelbach works with operations on rational functions, and due to error propagation is not practical for large size problems. Our alternative requires less consecutive operations and as such is more robust. We also propose an alternative based on the numerical solution of parametric systems of equations via continuation. For the mixed-integer linear programs we propose two algorithms. One algorithm is based on branch-and-bound on the integer variables and the other is based on the identification of optimality regions via mixed-integer nonlinear programs. We present a number of test problems including two case studies from man-portable power generation and numerical results from implementation of our algorithms. As expected, algorithms based on operations with rational functions are only applicable to problems of small size, approximately 10 variables. On the other hand continuation methods scale favorably to bigger problems. Also due to the special structure, the MINLP formulation of the optimality region is tractable for mid-size problems, with up to a few hundred variables.

3.7 Future Work

3.7.1 Algorithmic Improvement

In the branch-and-bound algorithms for the multiparametric cost vector case (Algorithm 3.4) and single parametric general case (Algorithm 3.7) upper bounds are only obtained when the LP relaxations gave an integer feasible solution. Another possibility to obtain upper bounds is to solve the parametric MILP at a fixed parameter value fix the integer variables and then solve the resulting parametric LP. This is essentially a combination of the branch-and-bound approach with the algorithms based on optimality regions, and is likely to show improved performance over both algorithms.

In the optimality region based algorithm for parametric MILP (Algorithm 3.6)

the solution of the MINLP is by far the most time consuming step, especially for increasing problem size. General-purpose solvers, can only partially exploit the problem structure and therefore specialized algorithms are expected to outperform these. A simple possibility to improve the computational performance is to perform sampling of the parameter range before calling the MINLP, i.e., choose some parameter points, solve the original MILP and check if the optimal objective value is better than the assumed optimal solution; if it is the parameter range for the MINLP can be reduced.

Using DSL48 as a continuation code is not the optimal choice. The first reason is related with the step size selection. The integrator used calculates the step size by controlling the truncation and the approximation error associated with the extrapolating polynomials [54]. For the continuation problems formulated there is no truncation error so that the step size calculation in DSL48 is not optimal. The second reason is that the state events correspond to the state variables plus a constant. Therefore introducing separate discontinuity functions is not the most efficient solution. Furthermore, the corrector used in DSL48 is based on the Newton method, whereas the solution of a linear system suffices. Preliminary numerical results show that on a Pentium IV at 3.0GHz for a problem with approximately 1000 variables DSL48 requires approximately 0.5s until the first event is hit while a call to MA48 (for fixed parameter value) only requires approximately 1.8ms. These considerations suggest that a specialized continuation algorithm would show a significant computational improvement. Note also, that the structure of the linear system does not change, so that analysis of the sparsity pattern only needs to be done once. Algorithm 3.8 sketches such an algorithm. Finally, it may be advantageous to use interpolating and extrapolating rational functions instead of polynomials.

Algorithm 3.8 (Specialized Continuation).

1. Initialization

Set $event_flag = 0$.

Set $p = p_0$

Analyze and store sparsity pattern.

Solve linear system.

2. **REPEAT**

- (a) Pick extrapolating function. (**Predictor**)
 - (b) Calculate step size dp .
 - (c) $step_flag = 0$.
 - (d) **REPEAT**
 - i. $p = p + dp$.
 - ii. Solve linear system for p . (**Corrector**)
 - iii. **IF** tolerances are met **THEN**
 - $step_flag = 1$.
 - Calculate interpolating function.
 - Apply event detection phase [221].
IF event is detected **THEN**
 - $event_flag = 1$
 - Apply consistent event phase [221].
 - ELSE**
 - $dp = dp/2$.
- UNTIL** $step_flag \neq 0$
- UNTIL** $event_flag \neq 0$

A problem with the branch-and-bound algorithms is that they cannot take advantage of advanced features of MILP solvers, such as the introduction of cuts. The development of cuts that are valid in the parametric case are expected to significantly improve the convergence of branch-and-bound.

3.7.2 Extension to Nonlinear Cost Vector Case

In Section 3.4 we considered parametric MILPs, for which the feasible region is parameter independent and the cost vector is an affine function of the parameters. It

is interesting to consider whether the theoretical properties and the algorithms presented can be readily extended to a MILP with nonlinear parameter dependence of the cost vector or to parametric mixed-integer nonlinear programs (MINLP) where the parameters only affect the objective function. Unfortunately, this is not possible in general, because the optimality regions of qualitatively invariant solutions are not convex.

Parametric MILP with Nonlinear Dependence of Cost Vector

Recall that in the cost vector case of parametric MILPs, the parameter range can be divided into a finite collection of (convex) polyhedra, for each of which a constant solution is optimal. This property is used in our algorithms, that rely in identifying the vertices of these polyhedra. If the dependence of the cost vector on the parameter is nonlinear, the optimality region of a constant solution is not necessarily convex, as Example 3.2 shows. Note that in Example 3.2 a concave dependence on the parameter is assumed, but a similar behavior can be constructed with nonlinear convex dependence on the parameter.

Example 3.2 (Parametric MILP with Nonlinear Parameter Dependence).

As an example consider a simple parametric linear program with nonlinear dependence on the parameter

$$\begin{aligned} \min_{\mathbf{x}} & (-0.09 + p - p^2) x_2 \\ \text{s.t. } & x_1 + x_2 = 1 \\ & \mathbf{x} \in [0, 1]^2. \end{aligned}$$

The optimality regions of the two vertices are

$$\begin{aligned} (1, 0) & \text{ is optimal for } p \in [0.1, 0.9] \\ (0, 1) & \text{ is optimal for } p \in [0, 0.1] \cup [0.9, 1]. \end{aligned}$$

The optimality region of $\mathbf{x} = (0, 1)$ is not convex.

Parametric MINLP

Unlike the MILP case, in mixed-integer nonlinear programs, in general there do not exist parameter sets with a nonempty interior for which a constant solution is optimal. It is conceivable to define qualitatively invariant solutions, e.g., a fixed integer realization along with some active constraint set, but in general, the optimality regions of such qualitatively invariant solutions are not convex, as Example 3.3 shows.

Example 3.3 (Parametric MINLP with Affine Parameter Dependence).

Consider the following MINLP with linear dependence of the objective function on the parameter

$$\begin{aligned} \min_{x,y} \quad & -x^2 p + x^4 + 0.01 y - 0.2 y p \\ \text{s.t.} \quad & x \leq 1 - y \\ & x \in [0, 1] \\ & y \in \{0, 1\}. \end{aligned}$$

Note that for all $p \in [0, 1]$ the constraints are linear and the objective function convex, and the above program is characterized as a mixed-integer convex program (MICP).

For $y = 1$ we obtain $x = 0$ and an objective function $f(p) = 0.01 - 0.2p$. For $y = 0$ the constraint is redundant. The stationarity condition $\frac{\partial(-x^2 p + x^4)}{\partial x} = 0$, gives a candidate optimal point x as a function of p as $\frac{\sqrt{2p}}{2}$. By convexity the only other candidates are the variable bounds $x = 0$ and $x = 1$. Therefore the optimal x as a function of p is $\frac{\sqrt{2p}}{2}$ with an objective function of $-\frac{p^2}{4}$. The optimality regions of the two integer realizations are thus

$$y = 0 \text{ is optimal for } p \in [0, 0.0536] \cup [0.7464, 1]$$

$$y = 1 \text{ is optimal for } p \in [0.0536, 0.7464]$$

and the optimality region of $y = 0$ is not convex.

3.7.3 Extension to General Nonlinear Case

In this section we briefly discuss the possible extension of our proposed methods for the general parametric MILP to nonlinear programs. Consider a general mixed-integer nonlinear program with parameter dependent functions

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y}, p) \\ \text{s.t. } \mathbf{g}(\mathbf{x}, \mathbf{y}, p) &\leq \mathbf{b}(p) \\ \mathbf{x} &\in X \subset \mathbb{R}^{n_x} \\ \mathbf{y} &\in \{0, 1\}^{n_y} \end{aligned} \tag{3.27}$$

to be solved for $p \in [0, 1]$. One possibility is to mimic the solution of MINLPs and solve parametric subproblems (over and under-estimating), e.g., parametric convex program, parametric LP, parametric MILP. Another possibility is to follow the continuation algorithm, thus taking advantage of state-of-the-art MINLP solvers.

Necessary Conditions

Suppose that we have solved (3.27) for $p = p_s$. Consider the NLP that results from fixing the integer variables to their optimal value $\bar{\mathbf{y}}$. Some of the constraints are active and some are inactive. Suppose also that some constraint qualification holds, and the KKT conditions are necessary for a local minimum. These necessary conditions define a system of equations that can be solved with a continuation approach, for increasing values of the parameter until one of the inactive primal constraints $g_i(\mathbf{x}, \bar{\mathbf{y}}, p) \leq b_i(p)$ or one of the dual inequalities (nonnegative KKT multiplier) is violated. As a result we obtain the parametric dependence of a candidate optimal solution $\bar{\mathbf{x}}(p)$, a candidate optimal objective function $\bar{f}(p)$ as well as a parameter range $[p_s, p_t]$ for which the candidate solution is feasible and satisfies the necessary optimality conditions.

Sufficient Conditions

The optimality region formulations did not assume linearity and (3.4) can be directly applied by adding the parameter to the optimization variables and including the cost constraint

$$\begin{aligned}
 & \min_{\mathbf{x}, \mathbf{y}, p} p \\
 & \text{s.t. } f(\mathbf{x}, \mathbf{y}, p) \leq \bar{f}(p) - \varepsilon \\
 & \quad \mathbf{g}(\mathbf{x}, \mathbf{y}, p) \leq \mathbf{b}(p) \\
 & \quad \mathbf{x} \in X \subset \mathbb{R}^{n_x} \\
 & \quad \mathbf{y} \in \{0, 1\}^{n_y} \\
 & \quad p \in P \subset \mathbb{R}.
 \end{aligned} \tag{3.28}$$

Similarly to the MILP case, (3.28) has in general more nonconvex nonlinear terms than the original problem (3.27).

Illustrative Example

As a preliminary test of extending our approach to nonlinear programs, we consider the chemical equilibrium of ideal gases for varying temperature. Note that chemical equilibrium of ideal gases is a convex nonlinear program with linear constraints and a strictly convex objective function [257]. The nonstoichiometric formulation is

$$\begin{aligned}
 & \min_{\mathbf{x}} \sum_{i=1}^n x_i \left(G_i^0(T) + RT \ln \frac{x_i}{x_n} \right) \\
 & \text{s.t. } \sum_{i=1}^n \alpha_{i,j} x_i = \sum_{i=1}^{n-1} \alpha_{i,j} b_i \quad j = 1, \dots, n_e \\
 & \quad x_{n+1} = \sum_{i=1}^n x_i \\
 & \quad \mathbf{x} \in \mathbb{R}^{n+1}, \quad \mathbf{x} \geq 0,
 \end{aligned} \tag{3.29}$$

where for $i = 1, \dots, n$ x_i denotes the molnumber of species i , the coefficients $\alpha_{i,j}$ correspond to the number of atoms of element j in species i and b_i is the initial molnumber of species i . Because of the logarithmic term, for implementation a nonzero lower bound needs to be imposed on the molnumbers $x_i \geq \varepsilon > 0$.

A system of 13 species and a 5-th order polynomial for the heat capacity is used. The Gibbs free energy G_i^0 consists of polynomial and logarithmic terms in the temperature. The temperature is considered as the parameter in the range 400 – 1000K. Note that the parameter only affects (nonlinearly) the (nonlinear) objective function. Discretization with a step size of 5K using SNOPT in GAMS takes approximately 5 seconds on a Dual Athlon at 1.63GHz. The continuation approach is tested using the process simulator ABACUSS II [273, 272]. There are convergence issues with the consistent initialization, i.e., the solution of the KKT conditions for $T = 400\text{K}$. In general, the solution provided by the NLP solver can be used as an initial guess and the convergence issues would be eliminated. Another numerical difficulty is that integration fails when the integration step becomes relatively large (approximately 8K). A maximal integration step cannot be specified in ABACUSS and we therefore integrated in steps which made the computational requirements significant, in the order of 10s. Since the problem is convex the KKT conditions are sufficient for optimality. Nevertheless we fitted a fifth-order polynomial to the objective function and tested (3.28) with BARON in GAMS with different tolerances ε corresponding to feasible or infeasible problems. All problems were solved successfully in approximately one second, but no guarantees were given for optimality or infeasibility because of the loose bounds on the variables provided. These preliminary numerical experiments suggest that the approach could indeed be extended to nonlinear programs.

3.7.4 Extension to General Multiparametric MILP

The extension of our algorithms to the multiparametric case is not straightforward. A major complication is that the optimality regions are arbitrary shaped. Another complication is that some of the approaches presented, such as predictor-corrector continuation, do not readily extend to the multiparametric case. We first present

examples that illustrate the shapes of optimality region and then discuss some possibilities of extending the subproblems.

Shape of Optimality Regions

Example 3.4 (Optimality Region: Box inside Box).

Consider the following parametric LP, where the parameters $\mathbf{p} \in [0, 1]^2$ affect the matrix entries

$$\begin{aligned}
 & \min_{\mathbf{x}} x_1 \\
 & \text{s.t. } (-1/4 + p_1)x_2 + x_1 = 1 \\
 & \quad (-1/4 + p_2)x_3 + x_1 = 1 \\
 & \quad (3/4 - p_1)x_4 + x_1 = 1 \\
 & \quad (3/4 - p_2)x_5 + x_1 = 1 \\
 & \quad \mathbf{x} \in \mathbb{R}^5, \quad \mathbf{x} \geq 0.
 \end{aligned} \tag{3.30}$$

There are two qualitatively different solutions, one with x_1 as a nonbasic variable and one with x_1 as a basic variable. In the former case we have $x_1 = 0$ and as a consequence

$$x_2 = \frac{1}{(-1/4 + p_1)}, \quad x_3 = \frac{1}{(-1/4 + p_2)}, \quad x_4 = \frac{1}{(3/4 - p_1)}, \quad x_5 = \frac{1}{(3/4 - p_2)}$$

this solution is feasible and optimal in $p_1, p_2 \in (1/4, 3/4)$, which is a convex region. In the latter case one of $\{x_2, x_3, x_4, x_5\}$ is nonbasic, equal to 0. From the corresponding equation we obtain $x_1 = 1$, and consequently $x_2 = x_3 = x_4 = x_5 = 0$. This solution is feasible over the whole parameter space and optimal in $[0, 1]^2 - (1/4, 3/4)^2$, which is not convex.

Example 3.5 (Nonconvex Optimality Regions with Curved Separation Lines).

Consider the following parametric LP, where the parameters $\mathbf{p} \in [0, 1]^2$ affect the

matrix entries

$$\begin{aligned}
& \min_{\mathbf{x}} x_1 \\
& \text{s.t.} \quad p_2 x_1 - x_2 = 1 \\
& -p_1 x_1 + x_3 + x_4 = 0 \\
& \quad \quad \quad p_1 x_3 = 1 \\
& \mathbf{x} \in \mathbb{R}^4, \quad \mathbf{x} \geq 0.
\end{aligned} \tag{3.31}$$

Because x_3 is the only variable in the third constraint, it is basic. For the special case $p_1 = 0$ we obtain a singular matrix and (3.31) is infeasible. For $p_2 = 0$ we obtain $x_2 = -1$ from the first constraint, which again renders the program infeasible. Therefore for $\mathbf{p} \in [0, 1]^2 : p_1 = 0 \vee p_2 = 0$ the system is infeasible. For $\mathbf{p} > \mathbf{0}$ there are three possible bases. The first possible basis contains x_2, x_3, x_4 ; setting $x_1 = 0$ we obtain

$$x_1 = 0, \quad x_2 = -1, \quad x_3 = 1/p_1, \quad x_4 = -1/p_1$$

and therefore this basis is infeasible in the whole parameter range. In the second possible basis, the basic variables are x_1, x_3, x_4 . Setting $x_2 = 0$ gives

$$x_1 = 1/p_2, \quad x_2 = 0, \quad x_3 = 1/p_1, \quad x_4 = \frac{p_1^2 - p_2}{p_1 p_2}.$$

This basis is feasible and optimal for $\mathbf{p} \in (0, 1]^2 : p_1^2 \geq p_2$, which is a nonconvex region with a curved separation line, one open and two closed faces. The third possible basis x_1, x_2, x_3 gives

$$x_1 = \frac{1}{p_1^2}, \quad x_2 = \frac{p_2 - p_1^2}{p_1^2}, \quad x_3 = \frac{1}{p_1}, \quad x_4 = 0.$$

This basis is feasible and optimal for $\mathbf{p} \in (0, 1]^2 : p_1^2 \leq p_2$, which is convex with a curved separation line, two open and two closed faces.

Checking Optimality Regions

Obtaining the parametric dependence of a solution in the multiparametric case is significantly more difficult than for a single parameter. The approach based on LU-factorization with rational operations works in principle for the multiparametric case, but has the practical limitations discussed earlier. The continuation approach on the other hand cannot be readily extended to the multiparametric case. It can be used to verify feasibility and LP optimality along parameter directions, e.g., along the p_1 axis.

A possibility is to check the feasibility and optimality range without explicitly obtaining the functional dependence. In general this leads to two-stage optimization problems which are very expensive to solve. Under Assumption 3.6, a single level formulation is possible. This can be done by considering the maximization of the bound violation of the basic variables \mathbf{x}_B subject to the system $\mathbf{B}(\mathbf{p})\mathbf{x}_B = \mathbf{b}(\mathbf{p})$ and leads to a nonsmooth nonconvex nonlinear problem

$$\begin{aligned} \max_{\mathbf{x}_B, \mathbf{p}, \mathbf{u}} \quad & \sum_{i=1}^{m_1+m_2} \max(u_i, v_i, 0) \\ \mathbf{B}(\mathbf{p})\mathbf{x}_B - \mathbf{b}(\mathbf{p}) \quad &= \mathbf{0} \\ u_i \leq \quad & -x_{B,i} \\ v_i \leq \quad & x_{B,i} - x_{B,i}^{UP} \\ \mathbf{p} \in \quad & P'. \end{aligned}$$

For \mathbf{x}_B the bounds should be set looser than the actual bounds $[\mathbf{0}, \mathbf{x}_B^{UP}]$. In some cases the special structure of this problem will make it tractable, but in general it is a very expensive program. Based on these considerations it is questionable if the general case can be treated efficiently with a generic algorithm. Heuristic based algorithms tailored to the systems are more likely to be successful.

Chapter 4

Bilevel Programming

Acknowledgments. *The development of the algorithm benefited from the collaboration with Panayiotis Lemonidis. In particular the elaborate branching in $2n_x + 1$ nodes is more due to Panayiotis Lemonidis than the author.*

4.1 Introduction and Literature Review

Bilevel programs are programs where an *outer program* is constrained by an embedded *inner program*. Here, inequality constrained nonlinear bilevel programs of the form

$$\begin{aligned} f^* &= \min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y}) \\ \text{s.t. } & \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \mathbf{y} \in \arg \min_{\mathbf{z}} h(\mathbf{x}, \mathbf{z}) \\ & \text{s.t. } \mathbf{p}(\mathbf{x}, \mathbf{z}) \leq \mathbf{0} \\ & \mathbf{q}(\mathbf{z}) \leq \mathbf{0} \\ & \mathbf{x} \in X \subset \mathbb{R}^{n_x}, \quad \mathbf{y}, \mathbf{z} \in Y \subset \mathbb{R}^{n_y}, \end{aligned} \tag{4.1}$$

are considered without any convexity assumptions. The co-operative (optimistic) formulation [85] is assumed, where if for a given \mathbf{x} the inner program has multiple optimal solutions \mathbf{y} , the outer optimizer can choose among them. As has been pro-

posed in the past, e.g., [68, 23], dummy variables (\mathbf{z} instead of \mathbf{y}) are used in the inner program since this clarifies some issues and facilitates discussion. We focus on nonconvex inner programs, i.e., the case when some of the functions $\mathbf{p}(\mathbf{x}, \mathbf{z})$, $\mathbf{q}(\mathbf{z})$, $h(\mathbf{x}, \mathbf{z})$ are not partially convex with respect to \mathbf{z} for all possible values of \mathbf{x} .

There are many applications of bilevel programs as well as theoretical and algorithmic contributions in the literature and the reader is directed to [53, 203, 274, 37, 85, 86, 253] for a thorough review of applications and algorithms; in this short literature review we focus on contributions relevant to nonconvex inner programs. Most of the literature is devoted to special cases of (4.1), in particular linear functions, e.g., [46]. Typically, the inner program is assumed convex satisfying a constraint qualification and is replaced by the equivalent KKT first-order optimality conditions; the resulting single level mathematical program with equilibrium constraints (MPEC) violates the Mangasarian-Fromovitz constraint qualification due to the complementarity slackness constraints [262]. Fortuny-Amat and McCarl [113] reformulate the complementarity slackness conditions using integer variables. Stein and Still [262] consider generalized semi-infinite programs (GSIP) and use a regularization of the complementarity slackness constraints. We will discuss how this strategy could be adapted to general bilevel programs (4.1). Very few proposals have been made for bilevel programs with nonconvex functions and to our best knowledge, no valid algorithm has been proposed to solve bilevel programs to guaranteed global optimality when nonconvexity is present in the inner program. Clark and Westerberg [69] introduced the notion of a local solution where the inner and outer programs are solved to local optimality and a semi-local solution is obtained [37, p. 341]. Gümüs and Floudas [133] proposed a B&B procedure to obtain the global solution of bilevel programs; we show here that this procedure is not valid in general when the inner program (4.2) is nonconvex. Bard [36] considered a simpler formulation without outer constraints and with a unique minimum for the inner problem, and proposed an algorithm based on a grid search between a lower and an upper bound of the optimal objective value, without a guarantee of convergence in the general case. Falk and Liu [104] proposed a bundle method which obtains local solutions to the inner and outer programs. Algorithms

that guarantee convergence to the global solution have been proposed for related programs under nonconvexity, such as min-max programs [105, 286], semi-infinite programs (SIP) [48, 45], and generalized semi-infinite programs [180].

We first discuss what is a reasonable expectation for the solution of nonconvex bilevel programs based on state-of-the-art notions in global optimization. To analyze some of the consequences of nonconvexity in the inner program we first discuss two equivalent reformulations of (4.1) as a simpler bilevel program and as a GSIP. We then discuss how to treat the inner variables in a B&B framework and identify issues with literature proposals regarding lower and upper bounds. Finally for KKT based approaches we discuss the need for bounds on the KKT multipliers.

We then propose a bounding algorithm for the global solution of (4.1) allowing nonconvex functions in both the inner and outer programs. The algorithm is based on a series of single-level optimization formulations. Equality constraints in the outer program would not change anything significant in the development of the algorithm and are only omitted for simplicity. The same holds for equality constraints in the inner program that do not depend on the outer variables \mathbf{x} . On the other hand, the presence of \mathbf{x} -dependent equality constraints in the inner program would require significant changes to the algorithm presented.

4.2 Definitions

Definition 4.1 (Inner Program). For a fixed \mathbf{x} we denote:

$$\begin{aligned}
 & \min_{\mathbf{z}} h(\mathbf{x}, \mathbf{z}) \\
 & \text{s.t. } \mathbf{p}(\mathbf{x}, \mathbf{z}) \leq \mathbf{0} \\
 & \quad \mathbf{q}(\mathbf{z}) \leq \mathbf{0} \\
 & \quad \mathbf{z} \in Y,
 \end{aligned} \tag{4.2}$$

the *inner program*.

Definition 4.2 (Parametric Optimal Solution Function). The parametric optimal

solution value of (4.2) as a function of the outer variables is denoted $\bar{h}(\mathbf{x})$ and the set of optimal points $H(\mathbf{x}) \subset Y$.

Definition 4.3 (ε -Optimality). A pair $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is called ε -optimal if it satisfies the constraints of the inner and outer programs, ε_h -optimality in the inner program, and ε_f -optimality in the outer program, i.e.:

$$\begin{aligned}
\mathbf{g}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) &\leq \mathbf{0} \\
\mathbf{p}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) &\leq \mathbf{0} \\
\mathbf{q}(\bar{\mathbf{y}}) &\leq \mathbf{0} \\
h(\bar{\mathbf{x}}, \bar{\mathbf{y}}) &\leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_h \\
f(\bar{\mathbf{x}}, \bar{\mathbf{y}}) &\leq f^* + \varepsilon_f.
\end{aligned} \tag{4.3}$$

Remark 4.1. In Section 4.4 we discuss that for bilevel programs (4.1) with nonconvex inner programs, it is only plausible to expect a finitely terminating algorithm to provide a guarantee for ε -optimality.

Definition 4.4 (\mathbf{x} Feasible in the Outer Program). The subset of X which is admissible in the outer program is denoted:

$$X_{outer} = \{\mathbf{x} \in X : \exists \bar{\mathbf{y}} \in Y : \mathbf{g}(\mathbf{x}, \bar{\mathbf{y}}) \leq \mathbf{0}\}.$$

Definition 4.5 (\mathbf{x} Feasible in the Inner Program). The subset of X which is admissible in the inner program is denoted:

$$X_{inner} = \{\mathbf{x} \in X : \exists \bar{\mathbf{y}} \in Y : \mathbf{p}(\mathbf{x}, \bar{\mathbf{y}}) \leq \mathbf{0}, \mathbf{q}(\bar{\mathbf{y}}) \leq \mathbf{0}\}.$$

Definition 4.6 (Level Sets). For a given $\bar{f} \in \mathbb{R}$ define the (potentially nonconvex) level sets

$$Q_l(\bar{f}) = \{\mathbf{x} \in X, \mathbf{y} \in Y : \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0}, \mathbf{p}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0}, \mathbf{q}(\mathbf{y}) \leq \mathbf{0}, f(\mathbf{x}, \mathbf{y}) \leq \bar{f}\}$$

and the projection to the X space

$$X_l(\bar{f}) = \{\mathbf{x} \in X : \exists \mathbf{y} \in Y : (\mathbf{x}, \mathbf{y}) \in Q_l(\bar{f})\}.$$

Definition 4.7 (Partition). A partition of a set $X^i \subset X$ is a finite collection of subsets, $X^l, X^{l+1}, \dots, X^{l+m}$ such that

$$X^i = X^l \cup X^{l+1} \cup \dots \cup X^{l+m} \quad \text{and} \quad \text{int}(X^{l_1}) \cap \text{int}(X^{l_2}) = \emptyset, \quad \forall l_1 \neq l_2,$$

compare also [154]. The definition for $X^i \times Y^i$ is analogous.

4.3 Reformulations

One of the major difficulties associated with the solution of (4.1) is that the two optimization programs communicate with each other through a solution *set* $H(\mathbf{x})$, possibly infinite. To facilitate the following discussion, two equivalent optimization formulations are discussed that communicate with each other through a solution *value*. These formulations have been proposed in the past by other authors [84, 85, 36] and are also similar to the difference function discussed by Amouzegar [23], where convexity is assumed and the variables \mathbf{y} do not participate in the outer constraints.

By the introduction of an additional variable h^* and the use of the dummy variables \mathbf{z} , the bilevel program (4.1) is equivalent [84, 85] to

$$\begin{aligned} & \min_{\mathbf{x}, \mathbf{y}, w} f(\mathbf{x}, \mathbf{y}) \\ & \text{s.t. } \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{p}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{q}(\mathbf{y}) \leq \mathbf{0} \\ & \quad h(\mathbf{x}, \mathbf{y}) - h^* \leq 0 \\ & \mathbf{x} \in X, \quad \mathbf{y} \in Y, \quad h^* \in \mathbb{R} \\ & \quad h^* = \min_{\mathbf{z}} h(\mathbf{x}, \mathbf{z}) \\ & \quad \text{s.t. } \mathbf{p}(\mathbf{x}, \mathbf{z}) \leq \mathbf{0} \\ & \quad \quad \mathbf{q}(\mathbf{z}) \leq \mathbf{0} \\ & \quad \quad \mathbf{z} \in Y. \end{aligned} \tag{4.4}$$

What is achieved with the introduction of the extra variable h^* is that the outer

and inner program are coupled by a much simpler requirement, namely the optimal solution *value* of an optimization program (\min) as opposed to the *set* of optimal solutions ($\arg \min$).

In principle (4.4) could be solved using multi-parametric optimization as a sub-problem. As a first step a global multi-parametric optimization algorithm would be used for the solution of the inner program for all $\mathbf{x} \in X$ dividing X into regions where the minimum \bar{h} is a known (smooth) function. The second step would be a global solution of the outer program for each of the optimality regions. This procedure is not suggested as a solution strategy since it would be overly computationally intensive. Note that application of this solution strategy to the original bilevel program (4.1) would require a parametric optimization algorithm that furnishes *all* optimal solutions as a function of the parameter \mathbf{x} , and to our best knowledge this is not possible at present.

The simplified bilevel program (4.4) allows the following observation: a relaxation of the inner program results in a restriction of the overall program and vice-versa. This observation also holds for other programs with a bilevel interpretation such as GSIPs, see [181], but does not hold for the original bilevel program (4.1) because any alteration of the inner program can result in an alteration of its set of optimal solutions and as such to the formulation of an unrelated optimization problem. We will show though that since relaxing the inner program in (4.4) results in an underestimation of h^* , the constraint $h(\mathbf{x}, \mathbf{y}) - h^* \leq 0$ becomes infeasible. Indeed, consider any $\bar{\mathbf{x}}$ and the corresponding optimal objective value of the inner program (4.2) h^* . By the definition of optimality we have $h(\bar{\mathbf{x}}, \mathbf{y}) \geq h^*$ for all \mathbf{y} feasible in the inner program, i.e., $\forall \mathbf{y} \in Y$, s.t. $\mathbf{p}(\bar{\mathbf{x}}, \mathbf{y}) \leq \mathbf{0}$. As a consequence no \mathbf{y} exists that can satisfy all the constraints of (4.4) exactly if h^* is underestimated. If ε_h optimality of the inner program is acceptable and the magnitude of the underestimation is less than ε_h it is possible to obtain (approximately) feasible points of (4.4).

The bilevel programs (4.1) and (4.4) are also equivalent to the following GSIP [36]

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y}) \\
& \text{s.t. } \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\
& \mathbf{p}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\
& \mathbf{x} \in X, \quad \mathbf{y} \in Y \\
& h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{z}), \quad \forall \mathbf{z} \in \bar{Y}(\mathbf{x}) \\
& \bar{Y}(\mathbf{x}) = \{\mathbf{z} \in Y : \mathbf{p}(\mathbf{x}, \mathbf{z}) \leq \mathbf{0}, \mathbf{q}(\mathbf{z}) \leq \mathbf{0}\}.
\end{aligned} \tag{4.5}$$

Note that this reformulation does not contradict the observation by Stein and Still [261] that under certain assumptions GSIP problems can be viewed as a special case of bilevel programs. Stein and Still compare a GSIP with variables \mathbf{x} to a bilevel program with variables \mathbf{x} and \mathbf{y} whereas the bilevel considered here (4.1) and the GSIP (4.5) have the same set of variables \mathbf{x}, \mathbf{y} . Moreover, the GSIP (4.5) does not necessarily satisfy the assumptions made by Stein and Still [261]. Note also that the GSIP (4.5) does not contain any Slater points as defined in Lemonidis and Barton [181] unless ε_h optimality of the inner program is allowed.

In either of the two reformulated programs our proposal of ε_h optimality can be interpreted as ε_h feasibility of the coupling constraint. As mentioned above, NLP solvers typically allow such a violation of ordinary nonlinear constraints, and this motivates again the proposal of ε -optimality (4.3).

4.4 Optimality Requirement

An interesting question is how exactly to interpret the requirement

$$\mathbf{y} \in \arg \min_{\mathbf{z} \in Y, \mathbf{p}(\mathbf{x}, \mathbf{z}) \leq \mathbf{0}, \mathbf{q}(\mathbf{z}) \leq \mathbf{0}} h(\mathbf{x}, \mathbf{z})$$

which by definition means that any feasible \mathbf{y} is a global minimum of the inner program for a fixed value of \mathbf{x} . Clark and Westerberg [69] proposed the notion of local solutions to the bilevel program (4.1) requiring that a furnished solution pair $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ satisfies local optimality in the inner program. This is in general a very strong relaxation. Bard [37, p. 341] gives formal definitions of so called semi-local solutions and describes also algorithms furnishing such points.

Note though that in single-level optimization finitely terminating algorithms guaranteeing global optimality exist only for special cases, e.g., linear programs. For nonconvex nonlinear programs (NLP)

$$\begin{aligned}
 f^* &= \min_{\mathbf{x}} f(\mathbf{x}) \\
 \text{s.t. } & \mathbf{g}(\mathbf{x}) \leq \mathbf{0} \\
 & \mathbf{x} \in X
 \end{aligned} \tag{4.6}$$

state-of-the-art algorithms in general only terminate finitely with an ε_f -optimal solution. That is, given any $\varepsilon_f > 0$ a feasible point $\bar{\mathbf{x}}$ is furnished and a guarantee that its objective value is not more than ε_f larger than the optimal objective value

$$f(\bar{\mathbf{x}}) \leq f^* + \varepsilon_f.$$

Moreover, to our best knowledge, no algorithm can provide guarantees for the distance of the points furnished from optimal solutions ($\|\bar{\mathbf{x}} - \mathbf{x}^*\|$, where \mathbf{x}^* is some optimal solution), which would allow a direct estimate of the requirement $\mathbf{y} \in \arg \min$. Finally global and local solvers for (4.6) typically allow an ε_g violation of the constraints [247].

We therefore propose that for bilevel programs (4.1) with nonconvex inner programs, it is only plausible to expect a finitely terminating algorithm to provide ε -optimal points, according to Definition (4.3). Note that as a consequence of ε_h -optimality in the inner program, $f^* \leq f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ need not hold (as Example 4.1 shows).

Example 4.2 illustrates that the same behavior can be observed in nonconvex (single level) NLPs when constraint violation is allowed.

Example 4.1 (Consequences of ε -Optimality in the Inner Program). Consider the bilevel program

$$\begin{aligned} \min_y y \\ \text{s.t. } y \in \arg \min_z -z^2 \\ y, z \in [-1 + \delta, 1], \end{aligned} \tag{4.7}$$

for some small $\delta > 0$. The inner program gives $y = 1$ and therefore the unique optimal solution is $y^* = 1$ with an optimal objective value of 1. Assume that the optimality tolerance of the inner program ε_h is such that $\varepsilon_h \geq 2\delta - \delta^2$. Based on definition (4.3) all $y \in [-1 + \delta, -\sqrt{1 - \varepsilon_h}]$ and $y \in [\sqrt{1 - \varepsilon_h}, 1]$ are admissible and as a consequence an objective value of $-1 + \delta$ can be obtained for the bilevel program (4.7).

Example 4.2 (Consequences of ε -feasibility in Nonconvex Nonlinear Programs). Consider the nonconvex nonlinear program

$$\begin{aligned} \min_x x \\ 1 - x^2 \leq 0 \\ x \in [-1 + \delta, 2], \end{aligned} \tag{4.8}$$

for some small $\delta > 0$. There are infinitely many feasible points $x \in [1, 2]$ and the problem satisfies the linear/concave constraint qualification [41, p. 322]. The unique optimal solution is $x^* = 1$ with an optimal objective value of 1. Assume that the feasibility tolerance of the nonlinear constraint ε_g is such that $\varepsilon_g \geq 2\delta - \delta^2$. All $x \in [-1 + \delta, -\sqrt{1 - \varepsilon_g}]$ and $x \in [\sqrt{1 - \varepsilon_g}, 2]$ are admissible and as a consequence an objective value of $-1 + \delta$ can be obtained for (4.8).

4.5 Consequences of Nonconvexity in the Inner Program

4.5.1 Branching on the \mathbf{y} Variables

As mentioned in the introduction, the main focus of this thesis is on the consequences of nonconvex inner programs for B&B procedures. In this section we briefly review some features of this algorithm class and discuss complications for the inner variables. For more details on B&B procedures the reader is referred to the literature, e.g., [154, 266]. For discussion purposes consider again the inequality constrained nonconvex nonlinear program (NLP) (4.6). The goal of B&B is to bracket the optimal solution value f^* between an upper bound UBD , typically the objective value at a feasible point, and a lower bound LBD , typically the optimal objective value of a relaxed program. To that end the host set X is partitioned into a finite collection of subsets $X_i \subset X$, such that $X = \cup_i X_i$. For each subset i a lower bound LBD_i and an upper bound UBD_i are established. The partitions are successively refined and typically this is done in an exhaustive matter [154], i.e., the diameter of each subset tends to zero. At each iteration the bounds are updated $LBD = \min_i LBD_i$ and $UBD = \min_i UBD_i$. Therefore UBD is non-increasing and LBD non-decreasing. The algorithm terminates when the difference $UBD - LBD$ is lower than a pre-specified tolerance.

For bilevel programs the inner program can have multiple optimal solutions for some values of \mathbf{x} and therefore a complication is whether or not to branch on the variables \mathbf{y} . Gümüs and Floudas [133] do not specify in their proposal if they do, and all their examples are solved at the root node. Amouzegar [23] for a special convex case proposes to only branch on the variables \mathbf{x} .

The dilemma is that on one hand \mathbf{y} are optimization variables in the outer program which suggests that one should branch on them but on the other hand the requirement of optimality in the inner program requires consideration of the whole set Y without branching. One extreme possibility is to branch on the variables $\mathbf{x} \in X$ and $\mathbf{y} \in Y$ without distinction. Example 4.3 shows that some points $(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \in X_i \times Y_i$ could be feasible for node i but not for the original program (4.1) and therefore $UBD = \min_i UBD_i$ cannot be used. The other extreme possibility is to consider for a given node X_i a partition of Y to subsets Y_j and keeping the worst subset. This procedure is not correct either as Example 4.4 shows. The introduction of dummy variables \mathbf{z} for the inner program allows the distinction of \mathbf{y} between the inner and the outer programs. In Section 4.6 we propose an algorithm where optional branching is performed on the \mathbf{x} and \mathbf{y} variables, but not on the \mathbf{z} variables. An alternative would be to employ subdivision similar to ideas presented for Semi-Infinite Programs [44, 45].

Example 4.3 (The host set of the inner program cannot be partitioned). Consider the simple linear bilevel program

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } -y \leq 0 \\
 & y \in \arg \min_z z \\
 & y, z \in [-1, 1].
 \end{aligned} \tag{4.9}$$

reset The inner program gives $y = -1$ which is infeasible in the outer program by the constraint $y \geq 0$. If we partition the host set $Y = [-1, 1]$ by bisection ($Y_1 = [-1, 0]$

and $Y_2 = [0, 1]$) we obtain two bilevel programs. The first one

$$\begin{aligned} \min_y y \\ \text{s.t. } -y \leq 0 \\ y \in \arg \min_z z \\ y, z \in [-1, 0] \end{aligned}$$

is also infeasible by the same argument. The second

$$\begin{aligned} \min_y y \\ \text{s.t. } -y \leq 0 \\ y \in \arg \min_z z \\ y, z \in [0, 1] \end{aligned}$$

gives $y = 0$. This point is not feasible in the original bilevel program (4.9). Taking the optimal solution of this node as an upper bound would give $UBD = 0$, which is clearly false.

Example 4.4 (The host set Y in the outer program cannot be partitioned keeping the worst subset). Consider the simple linear bilevel program

$$\begin{aligned} \min_y y \\ \text{s.t. } y \in \arg \min_z -z^2 \\ y, z \in [-1, 1]. \end{aligned} \tag{4.10}$$

The inner program gives $y = \pm 1$ and therefore the unique optimal solution is $y = -1$ with an optimal objective value of -1 . If we partition the host set $Y = [-1, 1]$ by

bisection ($Y_1 = [-1, 0]$ and $Y_2 = [0, 1]$) we obtain two bilevel programs. The first one

$$\begin{aligned} \min_y y \\ \text{s.t. } y \in \arg \min_z -z^2 \\ y, z \in [-1, 0] \end{aligned}$$

gives the unique feasible point $y = -1$ with an objective value of -1 . The second

$$\begin{aligned} \min_y y \\ \text{s.t. } y \in \arg \min_z -z^2 \\ y, z \in [0, 1] \end{aligned}$$

gives the unique feasible point $y = 1$ with an objective value of 1 . Keeping the worse of the two bounds would result in an increase of the upper bound.

4.5.2 Upper Bounding Procedure

Obtaining upper bounds for bilevel programs with a nonconvex inner program is very difficult due to the constraint “ \mathbf{y} is a global minimum of the inner program” and no obvious restriction of this constraint is available. Applying this definition to check the feasibility of a pair of candidate solutions $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ results in a global optimization problem for which in general state-of-the-art algorithms only furnish ε_f -optimal solutions at finite termination.

Only recently cheap upper bounds have been developed for some related problems such as SIPs [44, 45] and GSIPs [181] under nonconvexity. In these proposals the upper bounds are associated only with a feasible point $\bar{\mathbf{x}}$ and no identification of the inner variables is needed. Cheap upper bounds are conceivable for a general bilevel program but to our best knowledge no valid proposal exists for the case of nonconvex inner programs. On termination, algorithms for bilevel programs (4.1)

have to furnish a feasible pair $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, i.e., values must be given for the inner variables. If the upper bounds obtained in a B&B tree are associated with such feasible points, a global solution to a nonconvex optimization program has to be identified for each upper bound. This leads to an exponential procedure nested inside an exponential procedure and is employed in the algorithm proposed in Section 4.6. To avoid the nested exponential procedure a formulation of upper bounds without the identification of feasible points is required.

Solving a MPEC Locally Does Not Produce Valid Upper Bounds

Gümüs and Floudas [133] propose to obtain an upper bound by replacing the inner program by the corresponding KKT conditions, under some constraint qualification, and then solving the resulting program to local optimality. In this section we show that this procedure does not in general generate valid upper bounds. In the case of a nonconvex inner program, the KKT conditions are not sufficient for a global minimum and replacing the inner (nonconvex) program by the (necessary only) KKT conditions amounts to a relaxation of the bilevel program. Solving a (nonconvex) relaxed program to local optimality provides no guarantees that the point found is feasible to the original bilevel program; moreover the objective value does not necessarily provide an upper bound on the original bilevel program. This behavior is shown in Example 4.5. Although this example has a special structure, it shows a general property. Moreover in the proposal by Gümüs and Floudas [133] the complementarity constraint is replaced by the big-M reformulation and new variables are introduced; some issues associated with this are discussed in Section 4.5.4.

Example 4.5 (Counterexample for upper bound based on local solution of MPEC). For simplicity a bilevel program with a single variable y and box constraints is considered.

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y \in \arg \min_z -z^2 \\
 & y, z \in [-0.5, 1].
 \end{aligned} \tag{4.11}$$

By inspection the solution to the bilevel program (4.11) is $y = 1$ with an optimal objective value 1. The inner program has a concave objective function and linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187]. The KKT conditions for the inner program are given by

$$\begin{aligned}
-0.5 - y &\leq 0 \\
y - 1 &\leq 0 \\
-2y - \lambda_1 + \lambda_2 &= 0 \\
\lambda_1(-y - 0.5) &= 0 \\
\lambda_2(y - 1) &= 0 \\
\lambda_1 &\geq 0 \\
\lambda_2 &\geq 0.
\end{aligned} \tag{4.12}$$

System (4.12) has three solutions. The first solution is $y = -0.5$, $\lambda_1 = 1$, $\lambda_2 = 0$, a suboptimal local minimum of the inner program ($h = -0.25$). The second solution is $y = 0$, $\lambda_1 = 0$, $\lambda_2 = 0$, the global maximum of the inner program ($h = 0$). Finally the third solution is $y = 1$, $\lambda_1 = 0$, $\lambda_2 = 2$, the global minimum of the inner program ($h = -1$). The one-level optimization program proposed by Gümüs and Floudas [133] is therefore equivalent to

$$\begin{aligned}
&\min_y y \\
&\text{s.t. } y \in \{-0.5, 0, 1\}
\end{aligned} \tag{4.13}$$

and has three feasible points. Since the feasible set is discrete, all points are local minima and a local solver can converge to any of the points corresponding to the solutions of the system (4.12). The first solution $y = -0.5$, with an objective value $f = -0.5$ and the second solution $y = 0$, with an objective value $f = 0$ are both infeasible for (4.11). Moreover they clearly do not provide valid upper bounds on the

original program (4.11). The third solution $y = 1$ with an objective value of $f = 1$, which has the worst objective value for (4.13), is the true solution to the original program (4.11). Note that solving (4.13) to global optimality would provide a valid lower bound as described in the next section.

4.5.3 Lower Bounding Procedure

Lower bounds are typically obtained with an underestimation of the objective function and/or a relaxation of the feasible set [154] and a global solution of the resulting program. While global solution is a necessary requirement to obtain a valid lower bound, typically the constructed programs are convex and therefore global optimality can be guaranteed with local solvers.

Relaxation of the Inner Program Does Not Provide a Valid Lower Bound

Gümüs and Floudas [133] propose to obtain a lower bound to the bilevel program (4.1) by constructing a convex relaxation of the inner program (4.2) and replacing the convex inner program by its KKT conditions. Consider for a fixed $\bar{\mathbf{x}}$ a convex relaxation of (4.2) and the corresponding set of optimal solutions $H^c(\bar{\mathbf{x}})$

$$\begin{aligned}
 H^c(\bar{\mathbf{x}}) &= \arg \min_{\mathbf{z}} h^c(\bar{\mathbf{x}}, \mathbf{z}) \\
 \text{s.t. } &\mathbf{p}^c(\bar{\mathbf{x}}, \mathbf{z}) \leq \mathbf{0} \\
 &\mathbf{q}^c(\mathbf{z}) \leq \mathbf{0} \\
 &\mathbf{z} \in Y,
 \end{aligned} \tag{4.14}$$

where $h^c(\bar{\mathbf{x}}, \cdot)$, \mathbf{q}^c and $\mathbf{p}^c(\bar{\mathbf{x}}, \cdot)$ are convex underestimating functions of $h(\bar{\mathbf{x}}, \cdot)$, \mathbf{q} and $\mathbf{p}(\bar{\mathbf{x}}, \cdot)$ on Y for fixed $\bar{\mathbf{x}}$. This procedure results in an underestimation of the optimal solution value of the inner program but typically not in a relaxation of the feasible set of the bilevel program (4.1). The desired set inclusion property for the two sets of optimal solutions

$$H(\bar{\mathbf{x}}) \subset H^c(\bar{\mathbf{x}}) \tag{4.15}$$

does not hold in general. It does hold when the inner constraints $\mathbf{p}(\bar{\mathbf{x}}, \cdot)$ are convex and $h^c(\bar{\mathbf{x}}, \cdot)$ is the convex envelope of $h(\bar{\mathbf{x}}, \cdot)$. Constructing the convex envelope is in general not practical and Gümüs and Floudas propose to use α -BB relaxations [15]. Example 4.6 shows that convex underestimating functions based on this method do not guarantee the desired set inclusion property even in the case of linear constraints. In the case of nonconvex constraints one might postulate that the set inclusion property (4.15) holds if the feasible set is replaced by its convex hull and the objective function by the convex envelope. This is not true, as shown in Example 4.7. Both examples show that the proposal by Gümüs and Floudas does not provide valid lower bounds for bilevel programs involving a nonconvex inner program. Although the examples have a special structure, they show a general property. Moreover in the proposal by Gümüs and Floudas [133] the complementarity constraint is replaced by the big-M reformulation and new variables are introduced; issues associated with this are discussed in Section 4.5.4.

Example 4.6 (Counterexample for lower bound based on relaxation of inner program: box constraints). For simplicity a bilevel program with a single variable y and box constraints is considered

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y \in \arg \min_z z^3 \\
 & y, z \in [-1, 1].
 \end{aligned} \tag{4.16}$$

The inner objective function is monotone and the unique minimum of the inner program is $y^* = -1$. The bilevel program is therefore equivalent to

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y = -1 \\
 & y \in [-1, 1]
 \end{aligned}$$

with the unique feasible point $y^* = -1$ and an optimal objective value of -1 . Taking the α -BB relaxation for $\alpha = 3$ (the minimal possible) of the inner objective function, the relaxed inner program is

$$\begin{aligned} \min_z & z^3 + 3(1-z)(-1-z) \\ \text{s.t. } & z \in [-1, 1], \end{aligned}$$

which has the unique solution $y^* = 0$, see also Figure 4-1. Clearly the desired set inclusion property (4.15) is violated. Using the solution to the convex relaxation of the inner program, the bilevel program now is equivalent to

$$\begin{aligned} \min_y & y \\ \text{s.t. } & y = 0 \\ & y \in [-1, 1] \end{aligned}$$

and a false lower bound of 0 is obtained. Note that since the objective function is a monomial, its convex envelope is known [182]

$$h^c(z) = \begin{cases} \frac{3}{4}z - \frac{1}{4} & \text{if } z \leq \frac{1}{2} \\ z^3 & \text{otherwise.} \end{cases}$$

By replacing the inner objective with the convex envelope the minimum of the relaxed inner program would be attained at $y^* = -1$ and the set inclusion property would hold.

Example 4.7 (Counterexample for lower bound based on relaxation of inner program: nonconvex constraints). For simplicity a bilevel program with a single variable y convex objective functions and a single nonconvex constraint in the inner program is

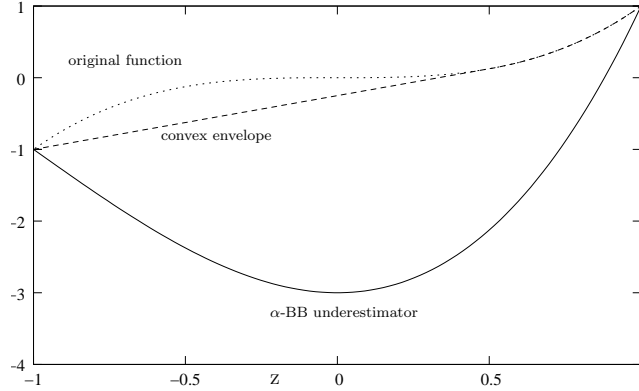


Figure 4-1: Inner level objective function, its convex envelope and its α -BB underestimator for example (4.6)

considered.

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y \in \arg \min_z z^2 \\
 & \quad \text{s.t. } 1 - z^2 \leq 0 \\
 & \quad y, z \in [-10, 10].
 \end{aligned} \tag{4.17}$$

The two optimal solutions of the inner program are $y^* = \pm 1$, so that the bilevel program becomes

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y = \pm 1 \\
 & \quad y \in [-10, 10]
 \end{aligned}$$

with the unique optimal point $y^* = -1$ and an optimal objective value of -1 . Taking the convex hull of the feasible set in the inner program, the relaxed inner program

$$\begin{aligned}
 & \min_z z^2 \\
 & \text{s.t. } z \in [-10, 10]
 \end{aligned}$$

has the unique solution $y^* = 0$. Clearly the desired set inclusion property (4.15) is violated. Using the solution to the convex relaxation of the inner program, the bilevel program becomes

$$\begin{aligned} \min_y & y \\ \text{s.t.} & y = 0 \\ & y \in [-10, 10]. \end{aligned}$$

and a false lower bound of 0 is obtained. Note that using any valid underestimator for the function $1 - z^2$ would result in the same behavior as taking the convex hull.

Lower Bounds Based on Relaxation of Optimality Constraint

A conceptually simple way of obtaining lower bounds to the bilevel programs is to relax the constraint “ \mathbf{y} is a global minimum of the inner program” with the constraint “ \mathbf{y} is feasible in the inner program” and to solve the resulting program globally. Assuming some constraint qualification, a tighter bound can be obtained by the constraint “ \mathbf{y} is a KKT point of the inner program”. In Section 4.6 we analyze how either relaxation can lead to convergent lower bounds. As mentioned in the introduction, Stein and Still [262] propose to solve GSIPs with a convex inner program by solving regularized MPECs. They note that for nonconvex inner programs this strategy would result to a relaxation of the GSIP. The complementarity slackness condition $\mu_i g_i(\mathbf{x}, \mathbf{z}) = 0$ is essentially replaced by $\mu_i g_i(\mathbf{x}, \mathbf{z}) = -\tau^2$, where τ is a regularization parameter. A sequence $\tau \rightarrow 0$ of regularized programs is solved; each of the programs is a relaxation due to the special structure of GSIPs. For a general bilevel program the regularization has to be slightly altered as can be seen in Example 4.8. The complication is that the KKT points are not feasible points in the regularized program and only for $\tau \rightarrow 0$ would a solution to the MPEC program provide a lower bound to the original bilevel program. A possibility to use the regularization approach for lower bounds would be to use an inequality constraint $-\tau^2 \leq \mu_i g_i(\mathbf{x}, \mathbf{z})$ as in [260]. To obtain the global solution with a branch-and-bound algorithm an

additional complication is upper bounds for the KKT multipliers, see Section 4.5.4.

Example 4.8 (Example for complication in lower bound based on regularized MPEC). Consider a linear bilevel program with a single variable y and for simplicity take \mathbb{R} as the host set.

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y \in \arg \min_z z \\
 & \quad \text{s.t. } -z \leq 0 \\
 & y, z \in \mathbb{R}.
 \end{aligned} \tag{4.18}$$

The inner program has the unique solution $y = 0$, and thus there is only one feasible point in the bilevel program with an objective value of 0. Note that all the functions are affine, and the inner program satisfies a constraint qualification. The equivalent MPEC is

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } 1 - \mu = 0 \\
 & \quad \mu \geq 0 \\
 & \quad -y \leq 0 \\
 & \quad \mu(-y) = 0 \\
 & \quad y \in \mathbb{R}.
 \end{aligned}$$

From the stationarity condition $1 - \mu = 0$ it follows $\mu = 1$ and therefore from the complementarity slackness $y = 0$. As explained by Stein and Still [262] the regularized

MPEC is equivalent to the solution of

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } 1 - \mu = 0 \\
 & \quad \mu \geq 0 \\
 & \quad -y \leq 0 \\
 & \quad \mu(-y) = -\tau^2 \\
 & \quad y \in \mathbb{R},
 \end{aligned} \tag{4.19}$$

which gives $\mu = 1$, $y = \tau^2$ and an objective value of τ^2 , which clearly is not a lower bound to 0. Replacing the requirement $\mu y = 0$ by $-\tau^2 \leq \mu(-y)$ or equivalently $y \leq \tau^2/\mu$ in (4.19) would give a correct lower bound of 0.

4.5.4 Complication in KKT Approaches: Multiplier Bounds

A complication with bounds based on KKT conditions are that extra variables μ (KKT multipliers) are introduced (one for each constraint) which are bounded only below (by zero). The big-M reformulation of the complementarity slackness condition needs explicit bounds for both the constraints g_i and KKT multipliers μ_i . Fortuny-Amat and McCarl [113] first proposed the big M-reformulation but do not specify how big M should be. Gümüs and Floudas [133] use the upper bound on slack variables as an upper bound for the KKT multipliers which is not correct in general as Example 4.9 shows. Note also that Gümüs and Floudas [133] do not specify how to obtain bounds on the slack variables of the constraints but this can be easily done, e.g., by computing an interval extension of the inequality constraints. Moreover, for a valid lower bound a further relaxation or a global solution of the relaxations constructed is needed and typically all variables need to be bounded [266, 247]. The regularization approach as in Stein and Still [262] does not need bounds for the regularization but if the resulting program is to be solved to global optimality bounds on the KKT multipliers are again needed. For programs with a special structure upper bounds may be known

a priori or it may be easy to estimate those. Examples are the feasibility test and flexibility index problems [132] where all KKT multipliers are bounded above by 1. Alternatively, specialized algorithms need to be employed that do not require bounds on the KKT multipliers.

Example 4.9 (Example with arbitrarily large KKT multipliers). Let us now consider a simple example with a badly scaled constraint that leads to arbitrarily large multipliers for the optimal KKT point

$$\begin{aligned} \min_z \quad & -z \\ & \delta(z^2 - 1) \leq 0 \\ & z \in [-2, 2], \end{aligned} \tag{4.20}$$

where $\delta > 0$. The only KKT point and optimal solution point is $z = 1$ and the KKT multiplier associated with the constraint $\delta(z^2 - 1) \leq 0$ is $\mu = \frac{1}{2\delta}$. The slack variable associated with the constraint can at most take the value $-\delta$. As $\delta \rightarrow 0$, the gradient of the constraint at the optimal solution $\frac{d(\delta(z^2-1))}{dz} = 2\delta z = 2\delta \rightarrow 0$ and μ becomes arbitrarily large. For $\delta = 0$ the Slater constraint qualification is violated.

4.6 Algorithmic Development

We propose a bounding algorithm for the global solution of (4.1) allowing nonconvex functions in both the inner and outer programs. Equality constraints in the outer program would not change anything significant in the development of the algorithm and are only omitted for simplicity. The same holds for equality constraints in the inner program that do not depend on the outer variables \mathbf{x} . On the other hand, the presence of \mathbf{x} -dependent equality constraints in the inner program would require changes to the algorithm presented.

In Section 4.6.1 we outline the assumptions necessary for finite termination of our algorithm. These assumptions also guarantee the existence of a minimum of (4.1). In Section 4.6.2 we present a lower bounding procedure based on the solution of

an optimization problem where the constraints of the inner and outer program are augmented by a parametric bound on the optimal solution value of the inner program as a function of the outer variables. In Section 4.6.3 we present the upper bounding procedure, which is based on probing the solution obtained by the lower bounding procedure. In Section 4.6.4 we describe the algorithmic framework and prove its finite convergence to an ε -optimal solution. As in branch-and-bound algorithms (B&B) for single-level programs, the optimal objective value is bracketed between a lower and an upper bound, but (explicit) branching of the variables is not required for convergence. The basic strategy of the algorithm is similar to the algorithm by Blankenship and Falk [48] for semi-infinite programs, in that the lower bounding problems become successively tighter, until the upper bounding problem is guaranteed to generate a feasible point. Branching may accelerate convergence and we propose three branching heuristics. In Section 4.6.6 we describe a basic numerical implementation of the algorithm and present results from its application to literature and original problems. Finally, we discuss the performance of the algorithm and propose improvements of the computational performance.

4.6.1 Assumptions

Assumption 4.1 (Host Sets). The host sets $X \subset \mathbb{R}^{n_x}$, $Y \subset \mathbb{R}^{n_y}$ are Cartesian products of (compact) intervals, i.e., for all variables explicit bounds are known ($x_j \in [x_j^{LO}, x_j^{UP}]$ for $j = 1, \dots, n_x$ and $y_j \in [y_j^{LO}, y_j^{UP}]$ for $j = 1, \dots, n_y$).

Remark 4.2. Considering arbitrary bounded polyhedra as host sets would not essentially alter the algorithm and the restriction to boxes is done for the sake of simplicity. With mild restrictions on branching, for each node $X^i \times Y^i$ we have $x_j \in [x_j^{i,LO}, x_j^{i,UP}]$ and $y_j \in [y_j^{i,LO}, y_j^{i,UP}]$.

Assumption 4.2 (Basic Properties of Functions). The functions $f : X \times Y \rightarrow \mathbb{R}$, $\mathbf{g} : X \times Y \rightarrow \mathbb{R}^{n_g}$, $h : X \times Y \rightarrow \mathbb{R}$, and $\mathbf{p} : X \times Y \rightarrow \mathbb{R}^{n_p}$ are continuous on $X \times Y$. Similarly, $\mathbf{q} : Y \rightarrow \mathbb{R}^{n_q}$ is continuous on Y .

Remark 4.3. By the continuity of the constraints and the compact host sets it directly

follows that X_{inner} , X_{outer} and $X_{inner} \cap X_{outer}$ are closed and therefore compact. Moreover, $Q_l(\bar{f})$ is compact for any \bar{f} and therefore also $X_l(\bar{f})$ is compact. Finally, for all $\bar{\mathbf{x}} \in X_{inner}$ the minimum of the inner program exists.

Assumption 4.3 (Inner Problem). There exists some $\tilde{\varepsilon}_f > 0$ such that for each point $\bar{\mathbf{x}} \in X_{outer} \cap X_{inner}$ at least one of the following two conditions holds:

1. For any $\varepsilon_{h1} > 0$ there exists a point $\tilde{\mathbf{z}} \in Y$ such that

$$\mathbf{p}(\bar{\mathbf{x}}, \tilde{\mathbf{z}}) < \mathbf{0}, \quad \mathbf{q}(\tilde{\mathbf{z}}) \leq \mathbf{0}, \quad h(\bar{\mathbf{x}}, \tilde{\mathbf{z}}) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h1}. \quad (4.21)$$

2. The outer objective value is $\tilde{\varepsilon}_f$ worse than the optimal objective value f^*

$$f(\bar{\mathbf{x}}, \bar{\mathbf{y}}) > f^* + \tilde{\varepsilon}_f, \quad \forall \bar{\mathbf{y}} \in Y : \mathbf{p}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \leq \mathbf{0}, \quad \mathbf{q}(\bar{\mathbf{y}}) \leq \mathbf{0}, \quad \mathbf{g}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \leq \mathbf{0} \quad (4.22)$$

or equivalently $\bar{\mathbf{x}} \notin X_l(f^* + \tilde{\varepsilon}_f)$.

Remark 4.4. By convention, for infeasible problems the optimal objective value is taken as infinity ($f^* = +\infty$). Therefore for infeasible problems condition (4.21) must hold for all $\bar{\mathbf{x}} \in X_{outer} \cap X_{inner}$.

Remark 4.5. Note that conditions (4.21) and (4.22) can both hold for some points.

Remark 4.6. Condition (4.21) of Assumption 4.3 allows the construction of parametric upper bounds for the parametric optimal solution function of the inner program, thus guaranteeing convergence of the branch-and-bound procedure. If this assumption was required for all $\mathbf{x} \in X_{outer} \cap X_{inner}$, continuity of the constraints \mathbf{p} would give $X_{outer} \cap X_{inner} = X_{outer}$, or the inner program would be feasible for all \mathbf{x} that are admissible in the outer program. It has been argued that this should always be the case [99, 37], but we do not make this restriction. Note that points $\mathbf{x} \notin X_{inner}$ are considered infeasible in the bilevel program.

Remark 4.7. Using the continuity of the inner objective function h , a sufficient condition for (4.21) is that for each $\bar{\mathbf{x}} \in X_{outer} \cap X_{inner}$ and for each solution point of the

inner problem $\mathbf{z}^* \in H(\bar{\mathbf{x}})$ and for each $\varepsilon_z > 0$ there exists a point $\tilde{\mathbf{z}} \in Y$, such that

$$\mathbf{p}(\bar{\mathbf{x}}, \tilde{\mathbf{z}}) < \mathbf{0}, \quad \mathbf{q}(\tilde{\mathbf{z}}) \leq \mathbf{0}, \quad \|\tilde{\mathbf{z}} - \mathbf{z}^*\| < \varepsilon_z. \quad (4.23)$$

Remark 4.8. In the case of differentiability of the inner problem, condition (4.23) and therefore also (4.21) can be derived from the Mangasarian-Fromowitz constraint qualification (MFCQ) [41, p. 323] for the inner program.

Assumption 4.4 (Assumptions for the KKT-based Lower Bound). Construction of lower bounds based on the KKT necessary conditions requires for all $\bar{\mathbf{x}} \in X_{outer} \cap X_{inner}$ the further assumptions of (i) differentiability of $h(\bar{\mathbf{x}}, \cdot)$ and $\mathbf{p}(\bar{\mathbf{x}}, \cdot)$ on some open set embedding Y (with respect to the inner variables \mathbf{z}), (ii) a constraint qualification for the inner program, and (iii) a-priori known upper bounds for the KKT multipliers.

Assumption 4.5 (Existence of Global NLP and MINLP Algorithms). For any $\varepsilon_{NLP} > 0$ there exist algorithms that can solve nonconvex nonlinear programs (NLP) and mixed-integer nonlinear programs (MINLP) involving a finite number of inequality constraints to ε_{NLP} -optimality and the functions in (4.1) satisfy their requirements, e.g., continuous second derivatives. On finite termination these algorithms provide a lower bound to the optimal solution value and a feasible point with an objective function value that is not more than ε_{NLP} larger than the lower bound.

Remark 4.9. All the formulated subproblems are inequality constrained with the exception of the stationarity constraint of the KKT-based lower bounds. The complementarity conditions are reformulated to inequalities involving binary variables. Since we do not use the KKT conditions for convergence, an approximate solution, i.e., a relaxation, of the stationarity condition, suffices for a lower bound. Typical NLP/MINLP solvers, e.g., [247] satisfy inequalities only within a (nonzero) tolerance. To account for this, only slight modifications would be needed for the results presented here.

4.6.2 Lower Bounding Procedure

In single-level optimization, lower bounds are typically obtained by the solution of a convex relaxation. As discussed in Section 4.5.3, a valid relaxation of the constraint “ \mathbf{y} is a global minimum of the inner program” is the constraint “ \mathbf{y} is feasible in the inner program” [274]. This idea is related to the upper bounding procedure in Gümüs and Floudas [133] which is valid for bilevel programs with a convex inner program [197] and also to remarks by Stein and Still [262] for generalized semi-infinite programs (GSIP).

It can be easily verified that the above requirement alone does not give a convergent lower bound, see, e.g., Example C.1. To achieve convergence, parametric upper bounds for the optimal solution function of the inner program are included in the lower bounding problem.

Consider subsets of the original host sets $X^i \subset X$ and $Y^i \subset Y$. Let K be an index set for a finite collection of pairs (\mathbf{y}^k, V^k) composed of points $\mathbf{y}^k \in Y$ and sets $V^k \subset X$, such that for each \mathbf{y}^k the inner constraints are satisfied for all $\mathbf{x} \in V^k$, i.e.,

$$\begin{aligned} \mathbf{q}(\mathbf{y}^k) &\leq \mathbf{0} \\ \mathbf{p}(\bar{\mathbf{x}}, \mathbf{y}^k) &\leq \mathbf{0}, \quad \forall \bar{\mathbf{x}} \in V^k. \end{aligned} \tag{4.24}$$

Then, the program

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} & f(\mathbf{x}, \mathbf{y}) \\ \text{s.t.} & \quad \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{p}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{q}(\mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{x} \in V^k \Rightarrow h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{y}^k), \quad \forall k \in K \\ & \quad \mathbf{x} \in X^i, \quad \mathbf{y} \in Y^i \end{aligned} \tag{4.25}$$

provides a relaxation of (4.1) (with \mathbf{x}, \mathbf{y} restricted to the host sets X^i, Y^i). Indeed,

consider a point $(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \in X^i \times Y^i$ which is feasible in (4.1). We directly obtain $\mathbf{g}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \leq \mathbf{0}$, $\mathbf{p}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) \leq \mathbf{0}$, and $\mathbf{q}(\bar{\mathbf{y}}) \leq \mathbf{0}$. Furthermore, since $\bar{\mathbf{y}}$ is a global minimum of the inner program for $\bar{\mathbf{x}}$ we obtain together with (4.24)

$$h(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = \bar{h}(\bar{\mathbf{x}}) \leq h(\bar{\mathbf{x}}, \mathbf{y}^k), \quad \forall k \in K : \bar{\mathbf{x}} \in V^k$$

which proves that $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ is feasible in (4.25). Therefore, a valid lower bound can be obtained from the global solution value of (4.25). Obviously, in the formulation of the lower bounding problem, only sets V^k that intersect with X^i need be considered. Also, when $X^i \subset V^k$ no logical constraint is needed and $h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{y}^k)$ can be directly used. Note that the use of logical constraints is well established, see, e.g., [281, 208, 42]. In Section 4.6.6 we describe a simple implementation of these constraints.

If Assumption 4.4 is satisfied, a tighter lower bound can be obtained by further requiring that \mathbf{y} satisfies the KKT necessary conditions for the inner program and adding the KKT multipliers to the set of variables

$$\begin{aligned} & \min_{\mathbf{x}, \mathbf{y}, \boldsymbol{\mu}} f(\mathbf{x}, \mathbf{y}) \\ & \text{s.t.} \quad \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{p}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{q}(\mathbf{y}) \leq \mathbf{0} \\ & \quad \mathbf{x} \in V^k \Rightarrow h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{y}^k), \quad \forall k \in K \\ & \quad \nabla_{\mathbf{y}} h(\mathbf{x}, \mathbf{y}) + \boldsymbol{\mu}^T \nabla_{\mathbf{y}} \tilde{\mathbf{p}}(\mathbf{x}, \mathbf{y}) = \mathbf{0} \\ & \quad \mu_j \tilde{p}_j(\mathbf{x}, \mathbf{y}) = 0, \quad j = 1, \dots, n_p + n_q + 2n_y \\ & \quad \mu_j \in [0, \mu_j^{max}], \quad j = 1, \dots, n_p + n_q + 2n_y \\ & \quad \mathbf{x} \in X^i, \quad \mathbf{y} \in Y^i, \end{aligned} \tag{4.26}$$

where for simplicity the constraints (\mathbf{p} and \mathbf{q}) of the inner program have been combined and augmented (to $\tilde{\mathbf{p}}$) to include the box constraints $\mathbf{y} \in Y$ (the complete host

set), i.e.:

$$\tilde{p}_i(\mathbf{x}, \mathbf{y}) = \begin{cases} p_j(\mathbf{x}, \mathbf{y}), & j = 1, \dots, n_p \\ q_j(\mathbf{y}), & j = n_p + 1, \dots, n_p + n_q \\ y_{j-n_p-n_q} - y_{j-n_p-n_q}^{UP}, & j = n_p + n_q + 1, \dots, n_p + n_q + n_y \\ -y_{j-n_p-n_q-n_y} + y_{j-n_p-n_q-n_y}^{LO}, & j = n_p + n_q + n_y + 1, \dots, n_p + n_q + 2n_y. \end{cases}$$

Note that state-of-the-art solvers in general provide only ε_{NLP} -estimates to the solution of either (4.25) or (4.26). To obtain a valid lower bound, the final lower bound provided by the solver has to be used as the lower bound. On the other hand, the ε_{NLP} -optimal point furnished is used for the subsequent steps of the algorithm.

Remark 4.10. The addition of the KKT necessary conditions does not guarantee convergence without the parametric upper bounds for the inner program. For instance consider Example C.4

$$\begin{aligned} \min_y & y \\ \text{s.t. } & y \in \arg \min_z -z^2 \\ & y, z \in [-0.5, 1]. \end{aligned}$$

The inner problem has three KKT points, $y \in \{-0.5, 0, 1\}$ but only $y = 1$ is the global minimum and therefore the only feasible point in the bilevel program. The optimal solution value of the bilevel program is 1 and therefore the lowest upper bound obtainable is $UBD = 1$. The KKT based lower bound (4.26) without the logical constraints, gives -0.5 for $Y = [-0.5, 1]$ (at the root node). If branching is performed, only nodes $Y^i \subset (0, 1]$ give a lower bound $\geq 1 = UBD$. Nodes containing -0.5 give a lower bound of -0.5 and nodes not containing -0.5 but containing 0 give a lower bound $LBD^i \leq 0$ and cannot be fathomed by value dominance. Since this example does not contain variables \mathbf{x} , fathoming can be performed based on a comparison of the objective value of the inner program. For general bilevel programs

this comparison needs to be done for fixed values of \mathbf{x} , or parametrically in \mathbf{x} , which motivates the logical constraints.

Remark 4.11. Assuming that finite upper and lower bounds are available, namely *UBD* and *LBD* respectively, these can be augmented to problems (4.25) or (4.26) as a constraint $LBD \leq f(\mathbf{x}, \mathbf{y}) \leq UBD$ with the aim of accelerating convergence. The lower bound inherited by the parent node can be used as a lower bound *LBD*. The current incumbent can be used for *UBD*; nodes with a lower bound that does not satisfy this inequality are fathomed anyway by value dominance.

Remark 4.12. An alternative to the global solution of problems (4.25) or (4.26) is to further relax these problems using convex relaxation methods, e.g., [266], and solve the resulting convex programs with a convex solver. In this case, also a feasible point of (4.25) or (4.26) should be obtained and used in the subsequent steps of the algorithm. This can also be achieved by solving (4.25) or (4.26) with a global solver and a loose tolerance ε_{NLP} .

In the remainder of this section a three-step procedure is described to obtain points \mathbf{y}^k and sets V^k for (4.24). The first step is to fix the variables \mathbf{x} to the values of the optimal solution $\bar{\mathbf{x}}$ obtained by the lower bounding problem (4.25) or (4.26) and to solve the inner problem globally

$$\begin{aligned} h^* &= \min_{\mathbf{z}} h(\bar{\mathbf{x}}, \mathbf{z}) \\ \text{s.t. } & \mathbf{p}(\bar{\mathbf{x}}, \mathbf{z}) \leq \mathbf{0} \\ & \mathbf{q}(\mathbf{z}) \leq \mathbf{0} \\ & \mathbf{z} \in Y, \end{aligned} \tag{4.27}$$

for the entire host set (Y as opposed to Y^i). The results of this step are also used for the upper bounding procedure, see Section 4.6.3. Feasibility of (4.27) is guaranteed by the solution of (4.25) or (4.26). Similarly to the solution of (4.25) or (4.26), the final lower bound from the solver needs to be used for h^* .

The second step is to pick $\varepsilon_{h2} > 0$ and to find a point \mathbf{y}^k such that $\mathbf{p}(\bar{\mathbf{x}}, \mathbf{y}^k) < \mathbf{0}$,

$\mathbf{q}(\mathbf{y}^k) \leq \mathbf{0}$ and $h(\bar{\mathbf{x}}, \mathbf{y}^k) \leq h^* + \varepsilon_{h2}$, e.g., by solution of the optimization problem

$$\begin{aligned}
& \min_{\mathbf{z}, u} u \\
& \text{s.t. } h(\bar{\mathbf{x}}, \mathbf{z}) \leq h^* + \varepsilon_{h2} \\
& p_i(\bar{\mathbf{x}}, \mathbf{z}) \leq u, \quad i = 1, \dots, n_p \\
& \mathbf{q}(\mathbf{z}) \leq \mathbf{0} \\
& \mathbf{z} \in Y, \quad u \leq 0.
\end{aligned} \tag{4.28}$$

This problem is feasible by the solution of (4.27). Provided that condition (4.21) of Assumption 4.3 is satisfied, the optimal solution value of (4.28) is negative and \mathbf{y}^k satisfies the required properties. To accelerate convergence, the solution of the inner problem (4.27) can be used as an initial guess. Finite convergence of the algorithm is guaranteed for sufficiently small ε_{h2} , see Section 4.6.5.

Remark 4.13. With the further assumption of MFCQ in the inner program, it would be possible to obtain a point \mathbf{y}^k by considering the solution of (4.27) and taking a small step in the descent direction of the constraints p_i of the inner program which at the optimal solution of (4.27) are active, i.e., equal to zero.

The third step is to identify a set V^k , that satisfies (4.24) and contains $\bar{\mathbf{x}}$ in its interior, or its boundary coincides with the boundary of X^i . This problem has been considered by Oluwole et al. [218] in the context of kinetic model reduction and their methodology can be directly used here. Successively smaller boxes V^k are guessed as shown in Subroutine 4.1. For a given box, (4.24) can in principle be checked by globally solving the nonsmooth nonconvex nonlinear optimization problem

$$u = \max_{\mathbf{x} \in V^k} \max_i p_i(\mathbf{x}, \mathbf{y}^k).$$

If $u \leq 0$, (4.24) is satisfied. Solving the above optimization problem is expensive and we therefore employ interval analysis to overestimate u . A consequence of this overestimation is that we do not obtain the largest possible V^k . Note that for an efficient implementation the details of this procedure are important and should be

tuned for the instance considered. For instance, it may be advantageous to guess V^k relative to the original host set X and not X^i .

Subroutine 4.1 (Calculating V^k).

Given a point $\bar{\mathbf{x}}$ and the bounds of node i on the \mathbf{x} variables $\mathbf{x}^{i,LO}$ and $\mathbf{x}^{i,UP}$ we want to calculate valid bounds for the box $V^k = [\mathbf{v}^{k,LO}, \mathbf{v}^{k,UP}]$. For simplicity, successively smaller boxes are guessed by scaling the node by $d \in (0, 1]$.

1. Set $d = 1$.

2. **LOOP**

(a) **FOR** $j = 1, \dots, n_x$ **DO**

- **IF** $\bar{x}_j - \frac{d}{2}(x_j^{i,UP} - x_j^{i,LO}) < x_j^{i,LO}$ **THEN**
 - Set $v_j^{k,LO} = x_j^{i,LO}$.
 - Set $v_j^{k,UP} = x_j^{i,LO} + d(x_j^{i,UP} - x_j^{i,LO})$.
- **ELSE IF** $\bar{x}_j + \frac{d}{2}(x_j^{i,UP} - x_j^{i,LO}) > x_j^{i,UP}$ **THEN**
 - Set $v_j^{k,LO} = x_j^{i,UP} - d(x_j^{i,UP} - x_j^{i,LO})$.
 - Set $v_j^{k,UP} = x_j^{i,UP}$.
- **ELSE**
 - Set $v_j^{k,LO} = \bar{x}_j - \frac{d}{2}(x_j^{i,UP} - x_j^{i,LO})$.
 - Set $v_j^{k,UP} = \bar{x}_j + \frac{d}{2}(x_j^{i,UP} - x_j^{i,LO})$.

END

(b) Check (4.24) by evaluating the interval extension of $\mathbf{p}(\cdot, \mathbf{y}^k)$ on V^k .

IF (4.24) is satisfied **THEN** terminate **ELSE** Reduce d **END**.

END

Note that the computational requirement for this subroutine is typically insignificant compared to the lower bounding problems.

At this point a brief explanation of interval analysis is warranted. For a thorough analysis, the reader is referred to the literature, e.g., [204, 19]. Since V^k is a Cartesian

product of intervals and the constraints of the inner problem \mathbf{p} are continuous, the image of each real valued function $p_i(\cdot, \mathbf{y}^k) : V^k \rightarrow \mathbb{R}$ (for fixed \mathbf{y}^k) is an interval $[p_i^l, p_i^u]$. An interval valued function $G(V^k)$ which satisfies

$$[p_i^l, p_i^u] \subset G(V^k) = [p_i^L, p_i^U]$$

is referred to as an *inclusion function* for $p_i(\cdot, \mathbf{y}^k)$ on V^k . An obvious requirement on the inclusion function is convergence to the true image $[p_i^l, p_i^u]$ as $\|\mathbf{v}^{k,UP} - \mathbf{v}^{k,LO}\|$ is reduced. The natural interval extension is an example of such an inclusion function. It is derived by replacing its variable x_j by the corresponding interval $[v_j^{k,LO}, v_j^{k,UP}]$ and evaluating the resulting expression using the rules of interval arithmetic [204]. The functions are decomposed into a sequence of compositions of elementary operations (e.g., multiplication, addition) and intrinsic functions, such as monomials or the exponential function. For each of the intrinsic functions and elementary operations, rules are available to construct the natural interval extension. For instance, in the addition of two intervals the lower bound is given by addition of the two lower bounds and the upper bound by addition of the two upper bounds. In general, natural interval extensions lead to an overestimation of $[p_i^l, p_i^u]$, but in special cases, such as monomials, an exact calculation is obtained. Tighter inclusion functions can be calculated using Taylor model inclusions [19]. Note also that interval analysis methods can be automated, see e.g., [269, 271].

4.6.3 Upper Bounding Procedure

As discussed in Section 4.5.2, currently no method exists that provides valid, convergent upper bounds for bilevel programs with nonconvex inner programs without the generation of feasible points. In this section we propose an upper bounding procedure by probing the feasibility of a candidate solution $\bar{\mathbf{x}}$ and $\mathbf{y} \in Y^i \subset Y$. Blankenship and Falk [48] used an analogous idea for SIP.

Given a candidate $\bar{\mathbf{x}}$, the first step is to solve the nonconvex inner program (4.27) globally and obtain an optimal solution $\bar{\mathbf{y}}$ and an optimal solution value h^* . For

an arbitrary point $\bar{\mathbf{x}}$, this program may be infeasible, in which case no solution to the bilevel program exists for $\mathbf{x} = \bar{\mathbf{x}}$ and no upper bound can be obtained. In our algorithm we only consider candidates generated by the solution of the lower bounding problem for which the feasibility of (4.27) is guaranteed. Given the solution h^* the outer problem is solved for the fixed $\bar{\mathbf{x}}$

$$\begin{aligned}
& \min_{\mathbf{y}} f(\bar{\mathbf{x}}, \mathbf{y}) \\
& \text{s.t. } \mathbf{g}(\bar{\mathbf{x}}, \mathbf{y}) \leq \mathbf{0} \\
& \quad \mathbf{p}(\bar{\mathbf{x}}, \mathbf{y}) \leq \mathbf{0} \\
& \quad \mathbf{q}(\mathbf{y}) \leq \mathbf{0} \\
& \quad h(\bar{\mathbf{x}}, \mathbf{y}) \leq h^* + \varepsilon_h \\
& \quad LBD^i \leq f(\bar{\mathbf{x}}, \mathbf{y}) \\
& \quad \mathbf{y} \in Y^i,
\end{aligned} \tag{4.29}$$

allowing an ε_h -violation of the inner level objective. This step is performed, because, due to potential non-uniqueness of the solutions of the inner program, a valid upper bound may be obtained even if the solution to (4.27) does not satisfy the outer constraints. If (4.29) is infeasible then no solution exists for $\mathbf{x} = \bar{\mathbf{x}}$; otherwise an upper bound is obtained. The inequality $LBD^i \leq f(\bar{\mathbf{x}}, \mathbf{y})$ is added to accelerate convergence of (4.29) and to alleviate partially the consequences of allowing ε_h -optimality in the inner program.

Remark 4.14. If the solution to (4.27) is feasible in the outer program, an upper bound is obtained without solving (4.29), but (4.29) in general gives a better upper bound. Similarly, an upper bound can be obtained through any feasible point of (4.29), e.g., a local solution. For reasons of simplicity, we will assume that (4.29) is always solved to global optimality. If the optimal solution point of (4.27) satisfies the outer constraints, it can be used as an initial guess for (4.29).

Remark 4.15. Several algorithmic heuristics to avoid unnecessary solutions of (4.29) are conceivable, but instead of directly using the heuristics, we will assume that the

NLP solver used can take advantage of these.

4.6.4 Algorithm Statement

The basic strategy of our algorithm is similar to the algorithm by Blankenship and Falk [48] for semi-infinite programs. The addition of parametric upper bounds on the optimal solution value of the inner program via the pairs (\mathbf{y}^k, V^k) makes the lower bounding problems successively tighter. Without branching, finite termination is essentially achieved because either the sets V^k cover $X_{inner} \cap X_{outer}$ and infeasibility is proved, or a lower bounding problem furnishes a point inside an existing set V^k and close to a previously generated point and a ε -optimal point is obtained. The generation of parametric upper bounds is possible for those points $\bar{\mathbf{x}}$ that satisfy condition (4.21) of Assumption 4.3. If the lower bounding procedure furnishes a point $\bar{\mathbf{x}}$ that does not satisfy (4.21), the corresponding node can be fathomed because $\bar{\mathbf{x}}$ satisfies (4.22).

We describe the algorithm in a branch-and-bound framework. Branching is not necessary for convergence, but is interesting as a heuristic for accelerating the convergence and has the advantage that it allows more flexibility, such as the local solution of certain subproblems. For finite termination without additional assumptions on the problem instance, some restrictions on the branching and/or node selection heuristics are required. In Theorem 4.1, we prove finite termination for the cases of no branching, as well as best-bound and breadth-first node selection heuristics. It is possible to show finite termination for other heuristics, but this is outside the scope of this thesis.

Input to the algorithm are the optimality tolerances ε_f and ε_h , satisfying the assumptions of Theorem 4.1.

Algorithm 4.1 (Main Algorithm).

1. **(Initialization)**

Set $LBD = -\infty$, $UBD = +\infty$, $k = 1$ and $l = 1$.

Set $K = \emptyset$ and $N = \{X \times Y\}$.

2. **(Termination Test)**

Delete from N all nodes $X^i \times Y^i$ with $LBD^i \geq UBD - \varepsilon_f$ (**Fathoming by value dominance**).

Set $LBD = \min_{X^i \times Y^i \in N} LBD^i$.

If $N = \emptyset$ terminate.

3. **(Node Selection)**

Select a node $X^i \times Y^i$ from N according to some node selection heuristic.

4. **(Lower Bounding)**

Solve (4.25) or (4.26) globally.

IF feasible **THEN**

- Set LBD^i to the optimal objective value (final lower bound).
- Set $\bar{\mathbf{x}}$ equal to the solution point (ε_{NLP} -optimal point).

ELSE (Fathoming by Infeasibility)

- Delete node $X^i \times Y^i$ from N and goto step 2.

END

5. **(Fathoming by Value Dominance)**

IF $LBD^i \geq UBD - \varepsilon_f$ **THEN** delete node $X^i \times Y^i$ from N and goto step 2.

6. **(Inner Problem)**

Solve NLP (4.27) globally for $\mathbf{x} = \bar{\mathbf{x}}$. (Recall that feasibility of this program is guaranteed.)

Set h^* equal to the optimal objective value (final lower bound).

7. **(Population of Parametric Upper Bounds to Inner Problem)**

Solve (4.28). (Recall that feasibility of this program is guaranteed.)

IF $u^* < 0$ **THEN**

- Set \mathbf{y}^k equal to the solution point.

- Obtain an appropriate set V^k .
- Insert k to K .
- Set $k = k + 1$.

ELSE

- Delete node $X^i \times Y^i$ from N and goto step 2.

END

8. **(Upper Bounding)**

Solve NLP (4.29) for $\mathbf{x} = \bar{\mathbf{x}}$ with h^* as the upper bound for $h(\bar{\mathbf{x}}, \mathbf{y})$ and (if feasible) obtain an ε_{NLP} -optimal point $\bar{\mathbf{y}}$.

IF feasible and $f(\bar{\mathbf{x}}, \bar{\mathbf{y}}) < UBD$ **THEN** set $UBD = f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and $(\mathbf{x}^*, \mathbf{y}^*) = (\bar{\mathbf{x}}, \bar{\mathbf{y}})$.

9. **(Optional Branching Step)**

Delete node $X^i \times Y^i$ from N .

Partition the set $X^i \times Y^i$ into m nodes $X^l \times Y^l, X^{l+1} \times Y^{l+1}, \dots, X^{l+m} \times Y^{l+m}$ according to some branching heuristic.

Set $LBD^l = LBD^{l+1} = \dots = LBD^{l+m} = LBD^i$.

Add the new nodes to N .

Set $l = l + m$.

10. **(Loop)**

Goto step 2.

At this point a justification of the potential fathoming at Step 7 is provided. If the lower bounding problem furnishes points $\tilde{\mathbf{x}}$ that do not satisfy (4.21), by Assumption 4.3, we have $\tilde{\mathbf{x}} \notin X_l(f^* + \varepsilon_f)$ and therefore the lower bound (with or without the KKT heuristic) is higher than the optimal solution value: $LBD^i \geq f^* + \varepsilon_f$. The solution of the lower bounding problem is found within tolerance ε_{NLP} and therefore, as long as $\varepsilon_{NLP} < \varepsilon_f$ such points can only be visited if branching is performed and only in nodes that do not contain points $\tilde{\mathbf{x}} \in X_l(f^*)$.

A direct consequence of the validity of the lower and upper bounding procedures is that on termination of the algorithm, if $UBD = +\infty$, the instance is infeasible. Otherwise, UBD is an ε_f -estimate of the optimal solution value ($UBD \leq f^* + \varepsilon_f$) and $(\mathbf{x}^*, \mathbf{y}^*)$ is an ε -optimal point (see Definition 4.3) at which UBD is attained. Note that depending on the problem instance it may be beneficial to perform Step 8 directly after Step 6 and only perform Step 7 if $LBD^i < UBD^i - \varepsilon_f$.

Note that Algorithm 4.1 can be applied to bilevel programs irrespectively of convexity properties. For bilevel programs with an inner program that is convex on Y for each fixed $\bar{\mathbf{x}}$ and that satisfies Assumption 4.4, application of the KKT-based lower bounding problem (4.26) leads to convergence at the root node. On the other hand, if the simpler lower bounding problem (4.25) is used on such programs, the convexity is not exploited.

4.6.5 Convergence Proof

Lemma 4.1 (Continuity of Optimal Solution Function of Inner Problem). *The optimal objective function $\bar{h} : X \rightarrow \mathbb{R}$ of the inner problem is continuous for all $\mathbf{x} \in X_{inner} \cap X_{outer}$ satisfying (4.21).*

Proof. Consider any fixed $\bar{\mathbf{x}} \in X_{inner} \cap X_{outer}$. By (4.21) for any $\varepsilon_{h1} > 0$, there exists $\tilde{\mathbf{y}} \in Y$ such that

$$\mathbf{p}(\bar{\mathbf{x}}, \tilde{\mathbf{y}}) < \mathbf{0}, \quad \mathbf{q}(\tilde{\mathbf{y}}) \leq \mathbf{0} \tag{4.30}$$

$$h(\bar{\mathbf{x}}, \tilde{\mathbf{y}}) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h1}. \tag{4.31}$$

By continuity of the inner objective $h(\cdot, \tilde{\mathbf{y}})$ on X , for any $\varepsilon_{h3} > 0$ there exists $\delta_1 > 0$ such that

$$h(\mathbf{x}, \tilde{\mathbf{y}}) < h(\bar{\mathbf{x}}, \tilde{\mathbf{y}}) + \varepsilon_{h3}, \quad \forall \mathbf{x} \in X : \|\bar{\mathbf{x}} - \mathbf{x}\| < \delta_1. \tag{4.32}$$

Combining inequalities (4.31) and (4.32) we obtain

$$h(\mathbf{x}, \tilde{\mathbf{y}}) < \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h1} + \varepsilon_{h3}, \quad \forall \mathbf{x} \in X : \|\bar{\mathbf{x}} - \mathbf{x}\| < \delta_1. \tag{4.33}$$

By (4.30) and continuity of $\mathbf{p}(\cdot, \tilde{\mathbf{y}})$, there exists $\delta_2 > 0$ such that

$$\mathbf{p}(\mathbf{x}, \tilde{\mathbf{y}}) \leq \mathbf{0}, \quad \forall \mathbf{x} \in X : \|\bar{\mathbf{x}} - \mathbf{x}\| < \delta_2.$$

Together with $\mathbf{q}(\tilde{\mathbf{y}}) \leq \mathbf{0}$, $\tilde{\mathbf{y}}$ is feasible in the inner program for all $\mathbf{x} \in X : \|\bar{\mathbf{x}} - \mathbf{x}\| < \delta_2$.

By the definition of $\bar{h}(\mathbf{x})$ we therefore have

$$\bar{h}(\mathbf{x}) \leq h(\mathbf{x}, \tilde{\mathbf{y}}), \quad \forall \mathbf{x} \in X : \|\bar{\mathbf{x}} - \mathbf{x}\| < \delta_2.$$

With (4.33) we obtain

$$\bar{h}(\mathbf{x}) < \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h1} + \varepsilon_{h3}, \quad \forall \mathbf{x} \in X : \|\bar{\mathbf{x}} - \mathbf{x}\| < \min(\delta_1, \delta_2)$$

which proves that \bar{h} is upper semi-continuous at $\bar{\mathbf{x}}$.¹

By Theorem 4.2.1 in Bank et al. [35] for all $\bar{\mathbf{x}} \in X_{inner} \cap X_{outer}$ the optimal objective function \bar{h} of the inner problem is lower semi-continuous. \square

Lemma 4.2 (Minimum of Bilevel Program Exists). *Under Assumptions 4.1, 4.2 and 4.3, either (4.1) is infeasible or the minimum of (4.1) exists.*

Proof. Assume for now that f^* denotes the infimum of (4.1) without asserting that the minimum is attained. By Definition 4.6 of the level sets, the bilevel program (4.1) is equivalent to

$$\begin{aligned} f^* &= \inf_{\mathbf{x}, \mathbf{y}} f(\mathbf{x}, \mathbf{y}) \\ \text{s.t. } & h(\mathbf{x}, \mathbf{y}) \leq \bar{h}(\mathbf{x}) \\ & (\mathbf{x}, \mathbf{y}) \in Q_l(f^*). \end{aligned} \tag{4.34}$$

Since the level set $Q_l(f^*)$ is compact, so is the feasible set of (4.34). Noting that for all $(\mathbf{x}, \mathbf{y}) \in Q_l(f^*)$ we have $\mathbf{x} \in X_l(f^*)$ and condition (4.21) of Assumption 4.3 is satisfied. Therefore, by Lemma 4.1 \bar{h} is continuous on the feasible set of (4.34).

¹Compare also Dempe [85, p. 65].

Therefore either (4.34) is infeasible or its minimum is attained. As a consequence either (4.1) is infeasible or the minimum of (4.1) exists.² \square

Lemma 4.3 (Sets V^k have Nonempty Interior). *For any (arbitrary but fixed) $\varepsilon_{h2} > 0$ there exists $\delta_1 > 0$ such that for any $\bar{f} \leq f^* + \tilde{\varepsilon}_f$ and for each point $\bar{\mathbf{x}} \in X_l(\bar{f})$ the points $\mathbf{y}^k(\bar{\mathbf{x}})$ generated in Step 7 of Algorithm 4.1 satisfy*

$$\mathbf{p}(\mathbf{x}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \mathbf{0}, \quad \mathbf{q}(\mathbf{y}^k(\bar{\mathbf{x}})) \leq \mathbf{0}, \quad h(\bar{\mathbf{x}}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h2}, \quad \forall \mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \delta_1.$$

Note that δ_1 is independent of $\bar{\mathbf{x}}$.

Proof. Since $\bar{f} \leq f^* + \tilde{\varepsilon}_f$, all points $\bar{\mathbf{x}} \in X_l(\bar{f})$ satisfy (4.21), $X_l(\bar{f})$ is compact and by Lemma 4.1 the optimal objective function of the inner problem $\bar{h} : X \rightarrow \mathbb{R}$ is continuous at all $\bar{\mathbf{x}} \in X_l(\bar{f})$.

Let $\bar{u}(\mathbf{x})$ denote the parametric optimal solution value of (4.28). By the continuity of the functions and the compactness of $X_l(\bar{f})$, \bar{u} is continuous and its maximum over $\bar{\mathbf{x}} \in X_l(\bar{f})$ is attained. Since $\varepsilon_{h2} > 0$, by (4.21) \bar{u} is strictly negative on $X_l(\bar{f})$. Therefore, there exists $\tilde{u} < 0$ such that for all $\bar{\mathbf{x}} \in X_l(\bar{f})$

$$p_i(\bar{\mathbf{x}}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \tilde{u} < 0, \quad i = 1, \dots, n_p, \quad \mathbf{q}(\mathbf{y}^k(\bar{\mathbf{x}})) \leq \mathbf{0}, \quad h(\bar{\mathbf{x}}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h2}.$$

Since $\mathbf{p}(\cdot, \mathbf{y})$ is continuous, and $X_l(\bar{f})$ is compact, $\mathbf{p}(\cdot, \mathbf{y})$ is uniformly continuous on $X_l(\bar{f})$ [244]. Therefore there exists $\delta_1 > 0$ (independent of $\bar{\mathbf{x}}$) such that for any $\bar{\mathbf{x}} \in X_l(\bar{f})$

$$\mathbf{p}(\mathbf{x}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \mathbf{0}, \quad \mathbf{q}(\mathbf{y}^k(\bar{\mathbf{x}})) \leq \mathbf{0}, \quad h(\bar{\mathbf{x}}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h2}, \quad \forall \mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \delta_1.$$

\square

Remark 4.16. A direct consequence of Lemma 4.3 is that there exists $d_1 > 0$, such

²Compare also Dempe [84].

that all sets V^k obtainable in Step 7 of Algorithm 4.1 satisfy

$$\min_j \{v_j^{k,UP} - v_j^{k,LO}\} \geq d_1.$$

Our proposal of interval analysis underestimates the size of these sets, but it has been shown [181] that natural interval extensions converge uniformly and therefore there exists $d_2 > 0$, such that for all $\bar{\mathbf{x}} \in X_l(\bar{f})$ the sets obtained satisfy

$$\min_j \{v_j^{k,UP} - v_j^{k,LO}\} \geq d_2.$$

Lemma 4.4. *Let $X_t \subset X$ be compact, and $\delta > 0$. Consider any infinite sequence of points $\mathbf{x}^i \in X_t$. There exists a finite index $I > 0$, such that*

$$\|\mathbf{x}^I - \mathbf{x}^i\| \leq \delta, \quad \text{for some } i < I.$$

Proof. Consider any infinite sequence $\mathbf{x}^i \in X_t$. Since X_t is compact it is also bounded and therefore the sequence is also bounded. Therefore there exists a point $\bar{\mathbf{x}}$ and a subsequence \mathbf{x}^{i_k} that converges to $\bar{\mathbf{x}}$ (see, e.g., Theorem 3.6 in [244]), i.e., there exists a finite $K > 0$, such that

$$\|\mathbf{x}^{i_k} - \bar{\mathbf{x}}\| \leq \delta/2, \quad \forall k \geq K.$$

Therefore

$$\|\mathbf{x}^{i_{K+1}} - \mathbf{x}^{i_K}\| \leq \|\mathbf{x}^{i_{K+1}} - \bar{\mathbf{x}}\| + \|\mathbf{x}^{i_K} - \bar{\mathbf{x}}\| \leq \delta/2 + \delta/2 \leq \delta.$$

For $I = i_{K+1}$ and $i = i_K$ we have the desired result. □

Theorem 4.1 (Finite Termination). *If the tolerance of the optimization subproblems*

ε_{NLP} and ε_{h2} in (4.28) satisfy

$$0 < \varepsilon_{NLP} \leq \min(\varepsilon_f/2, \varepsilon_h, \tilde{\varepsilon}_f)$$

$$0 < \varepsilon_{h2} < \varepsilon_h - \varepsilon_{NLP}$$

and either of the following three algorithmic heuristics are employed

1. *No branching is performed.*
2. *The branching heuristic is exhaustive [153] in X and the breadth-first node selection heuristic is used.*
3. *The branching heuristic is exhaustive in X and the best-bound node selection heuristic is used.*

then Algorithm 4.1 terminates finitely.

Since the proof of Theorem 4.1 is lengthy, we first present an outline of the proof. In the case of no branching, and since the lower bounding problem visits only points $\bar{\mathbf{x}} \in X_{outer} \cap X_{inner}$, it will not visit points $\tilde{\mathbf{x}} \notin X_l(f^* + \tilde{\varepsilon}_f)$. Therefore, by Assumption 4.3 at the points visited by the lower bounding problem it is possible to construct the parametric upper bounds to the optimal solution value of the inner program via the pairs (\mathbf{y}^k, V^k) . The corresponding logical constraints augmented to the lower bounding problem successively tighten the lower bounding problem to the extent that it will either become infeasible or furnish a point inside an existing V^k which is also a ε -optimal point. In the case of branching and since the branching heuristic is exhaustive in X , the nodes visited eventually become smaller than the smallest possible V^k and are fathomed.

Proof.

Let $\bar{f} = f^* + \tilde{\varepsilon}_f$.

1. *No branching*

Consider first the case that no branching is performed and note that in this case

only points $\bar{\mathbf{x}} \in X_l(f^* + \varepsilon_{NLP}) \subset X_l(\bar{f})$ are furnished by the lower bounding problem.

Points $\mathbf{x} \in X_l(\bar{f})$ satisfy condition (4.21) of Assumption 4.3 and this allows the generation of logical constraints via the pairs (\mathbf{y}^k, V^k) . Let $\bar{\mathbf{x}} \in X_l(\bar{f})$ be furnished by the lower bounding problem. We will show that if in a subsequent iteration the lower bounding problem furnishes a point $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ with $\hat{\mathbf{x}}$ sufficiently close to $\bar{\mathbf{x}}$, it will be ε -optimal. By Lemma 4.3 there exists $\delta_1 > 0$ such that for each point $\bar{\mathbf{x}} \in X_l(\bar{f})$ the points \mathbf{y}^k generated in Step 7 of Algorithm 4.1 satisfy

$$\mathbf{p}(\mathbf{x}, \mathbf{y}^k) \leq \mathbf{0}, \quad \mathbf{q}(\mathbf{y}^k(\bar{\mathbf{x}})) \leq \mathbf{0}, \quad h(\bar{\mathbf{x}}, \mathbf{y}^k(\bar{\mathbf{x}})) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h2}, \quad \forall \mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \delta_1. \quad (4.35)$$

Recall also

$$h(\bar{\mathbf{x}}, \mathbf{y}^k) \leq \bar{h}(\bar{\mathbf{x}}) + \varepsilon_{h2}.$$

By assumption $\varepsilon_h - \varepsilon_{NLP} - \varepsilon_{h2} > 0$. By continuity of \bar{h} at $\bar{\mathbf{x}}$ there exists $\delta_2 > 0$ such that

$$\bar{h}(\bar{\mathbf{x}}) \leq \bar{h}(\mathbf{x}) + (\varepsilon_h - \varepsilon_{h2} - \varepsilon_{NLP})/2, \quad \forall \mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \delta_2.$$

By continuity of $h(\cdot, \mathbf{y}^k)$ on X , there exists $\delta_3 > 0$ such that

$$h(\mathbf{x}, \mathbf{y}^k) \leq h(\bar{\mathbf{x}}, \mathbf{y}^k) + (\varepsilon_h - \varepsilon_{h2} - \varepsilon_{NLP})/2, \quad \forall \mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \delta_3.$$

Combining these last three inequalities gives

$$h(\mathbf{x}, \mathbf{y}^k) \leq \bar{h}(\mathbf{x}) + \varepsilon_h - \varepsilon_{NLP}, \quad \forall \mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \min(\delta_2, \delta_3).$$

Therefore, together with (4.35), \mathbf{y}^k is ε_h -optimal in the inner problem for all $\mathbf{x} \in X : \|\mathbf{x} - \bar{\mathbf{x}}\| < \delta$, where $\delta = \min(\delta_1, \delta_2, \delta_3) > 0$. Note that these $(\mathbf{x}, \mathbf{y}^k)$ are not necessarily feasible with respect to the outer constraints, and therefore termination does not occur immediately.

Since $X_l(\bar{f})$ is compact and $\delta > 0$, by Lemma 4.4 after a finite number of iter-

ations either the lower bounding problem becomes infeasible, in which case the algorithm terminates, or the lower bounding problem furnishes a point $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$, with $\hat{\mathbf{x}}$ sufficiently close to $\bar{\mathbf{x}}$, i.e., $\|\hat{\mathbf{x}} - \bar{\mathbf{x}}\| < \delta$. By construction of the lower bounding problem, this $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ satisfies the inner and outer constraints and also $h(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \leq h(\hat{\mathbf{x}}, \mathbf{y}^k)$ (by the logical constraint) and as a consequence

$$h(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \leq \bar{h}(\hat{\mathbf{x}}) + \varepsilon_h - \varepsilon_{NLP}$$

or $\hat{\mathbf{y}}$ is ε_h -optimal in the inner problem for $\hat{\mathbf{x}}$.³ The lower bound LBD^i obtained satisfies $LBD^i \geq f(\hat{\mathbf{x}}, \hat{\mathbf{y}}) - \varepsilon_{NLP}$. The point $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ is feasible in the bilevel program and therefore the upper bounding problem (step 8) furnishes an upper bound UBD^i , satisfying $UBD^i \leq f(\hat{\mathbf{x}}, \hat{\mathbf{y}}) + \varepsilon_{NLP}$. Noting now that the optimization problems are solved with tolerance $\varepsilon_{NLP} < 2\varepsilon_f$ it follows $UBD^i - LBD^i \leq 2\varepsilon_{NLP} \leq \varepsilon_f$ and the algorithm terminates.

2. Breadth-first node selection heuristic

Let δ have the same meaning as in the case without branching. Using the breadth-first node selection heuristic, and since the branching is exhaustive in X , for any $d > 0$, after a finite number of iterations for all nodes i and all variables x_j we have

$$x_j^{i,UP} - x_j^{i,LO} < d, \quad \forall j = 1, \dots, n_x, \quad \forall i \in N.$$

Note that nodes i with $X^i \cap X_{inner} \cap X_{outer} = \emptyset$ are fathomed by infeasibility. Without loss of generality we ignore such nodes in the following. At every level, one or more nodes satisfy $X_l(f^*) \cap X^i \neq \emptyset$. For all these nodes the lower bounding problem generates points $\bar{\mathbf{x}} \in X_l(f^* + \varepsilon_{NLP}) \subset X_l(\bar{f})$. Once d is sufficiently small such that

$$\|\mathbf{x} - \bar{\mathbf{x}}\| < \delta, \quad \forall \mathbf{x} \in X^i$$

³The ε_{NLP} tolerance is included here, because the global solution of the inner problem only gives a ε_{NLP} -estimate of $\bar{h}(\bar{\mathbf{x}})$.

the lower bounding problem of all children nodes j is either infeasible or furnishes $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ with $\hat{\mathbf{x}}$ sufficiently close to $\bar{\mathbf{x}}$, leading to an upper bound UBD^j such that $UBD^j - LBD^j \leq 2\varepsilon_{NLP} \leq \varepsilon_f$. In either case all children nodes of i are fathomed. If the problem is infeasible, all nodes satisfy $X_l(f^*) \cap X^i \neq \emptyset$ and are fathomed. Otherwise, at least one node contains an optimal solution and furnishes $UBD^j \leq f^* + \varepsilon_f = \bar{f}$. With this incumbent, all nodes X^i , for which $X_l(f^*) \cap X^i = \emptyset$ are fathomed when they are visited.

3. Best-bound node selection heuristic

Let δ have the same meaning as in the case without branching. At each iteration, a node with the best lower bound is chosen. Therefore only nodes i with an inherited lower bound $LBD^i \leq f^*$ can be chosen.

Consider any infinite nested sequence of nodes. The nodes in this sequence satisfy $X_l(f^*) \cap X^i \neq \emptyset$, and therefore the lower bounding problem generates points $\bar{\mathbf{x}} \in X_l(f^* + \varepsilon_{NLP}) \subset X_l(\bar{f})$. Since branching is exhaustive in X , for any $d > 0$, after a finite number of iterations for all variables x_j we have

$$x_j^{i,UP} - x_j^{i,LO} < d, \quad \forall j = 1, \dots, n_x.$$

Similarly to the best-bound heuristic this leads to either fathoming by infeasibility or generation of an upper bound such that $UBD^i - LBD^i \leq \varepsilon_f$ and the node is fathomed. Since $LBD^i \leq f^*$, this upper bound is sufficient to fathom all nodes by value dominance.

□

Remark 4.17. Note the requirement that the branching procedure is exhaustive allows the branching to be performed only every finite number of iterations. Moreover, the proof can be extended to the case that branching is performed only a finite number of iterations and the resulting nodes are visited without further branching.

Remark 4.18. For a finite number of iterations, an arbitrarily large ε_{h2} can be used for (4.28). In the worse case, this will create a finite number of redundant logical

constraints, but may accelerate convergence, by obtaining larger sets V^k at step 7.

Remark 4.19. Since we have proved that after a finite number of iterations the lower bounding problem furnishes a point that satisfies ε -optimality, the solution of the upper bounding problem is not required for finite convergence. On the other hand, the upper bounding procedure may accelerate convergence, and we therefore consider its solution at every iteration. Note that $LBD \leq f^*$ is always guaranteed since the lower bounding problem is a valid relaxation.

Branching Heuristics

While a great number of heuristics are conceivable, we consider three of particular interest. The simplest possibility is the extreme of no branching which results in a very simple implementation of the algorithm. In this case the addition of the logical constraints makes the lower bounding problems successively tighter and, in general, more expensive to solve.

Another simple choice is to perform branch-and-bound without any distinction between \mathbf{x} and \mathbf{y} by bisection on the variable with the current largest range. This procedure corresponds to a common branching heuristic in global single-level optimization. Convergence is achieved by a combination of the node shrinking and the addition of parametric upper bounds to the inner problem.

Remark 4.20. For the breadth-first selection heuristic and branching by bisection on the variable with the current largest range, finite convergence can be guaranteed for $\tilde{\varepsilon}_f = 0$ in condition (4.22).

A more elaborate and specialized branching heuristic is to branch only on the \mathbf{x} variables by partitioning the node into a set of nodes in such a way that one of the children nodes corresponds to the set V^k . The advantage of this branching heuristic is that it avoids the use of logical constraints. Given the parent node bounds on the variables $\mathbf{x}^{i,LO}$ and $\mathbf{x}^{i,UP}$ and the box bounds $\mathbf{v}^{k,LO}$ and $\mathbf{v}^{k,UP}$ a simple procedure for this partitioning is described in Subroutine 4.2.

Subroutine 4.2 (Branching into $2n_x + 1$ Nodes).

Two temporary vectors $\tilde{\mathbf{x}}^{LO}$ and $\tilde{\mathbf{x}}^{UP}$ and two temporary variables t_1 and t_2 are used.

1. **(Initialization)**

Set $\tilde{\mathbf{x}}^{LO} = \mathbf{v}^{k,LO}$ and $\tilde{\mathbf{x}}^{UP} = \mathbf{v}^{k,UP}$.

2. **(Node corresponding to V^k box)**

Create node with $\mathbf{x} \in [\tilde{\mathbf{x}}^{LO}, \tilde{\mathbf{x}}^{UP}]$.

3. **(Up to two nodes per variable)**

FOR $j = 1, \dots, n_x$ **DO**

- Set $t_1 = \tilde{x}_j^{LO}$ and $t_2 = \tilde{x}_j^{UP}$
- **IF** $t_1 > x_j^{i,LO}$
 - $\tilde{x}_j^{LO} = x_j^{i,LO}$ and $\tilde{x}_j^{UP} = t_1$.
 - Create node with $\mathbf{x} \in [\tilde{\mathbf{x}}^{LO}, \tilde{\mathbf{x}}^{UP}]$.
- **IF** $t_2 < x_j^{i,UP}$
 - $\tilde{x}_j^{LO} = t_2$ and $\tilde{x}_j^{UP} = x_j^{i,UP}$.
 - Create node with $\mathbf{x} \in [\tilde{\mathbf{x}}^{LO}, \tilde{\mathbf{x}}^{UP}]$.
- Set $\tilde{x}_j^{LO} = x_j^{i,LO}$ and $\tilde{x}_j^{UP} = x_j^{i,UP}$.

END FOR

By construction, the created nodes are a partition of the parent node and a maximum of $2n_x + 1$ children nodes are generated.

Remark 4.21. When the set V^k is equal to X^i , e.g., when the inner constraints do not depend on \mathbf{x} , this procedure re-creates the node X^i which is equivalent to the heuristic of no branching.

4.6.6 Implementation and Numerical Results

Implementation

The algorithm was implemented in C++ and tested on a 64-bit Xeon processor 3.2GHz running Linux 2.6.13. The best-bound heuristic occurred at the node selection step, and among nodes with the same lower bound, the one that entered the set of active nodes (N) first was always used. As is typical with optimization codes, both an absolute and relative termination criterion was used and termination occurred if either of the criteria was satisfied.

The resulting nonconvex NLP and MINLPs were all solved globally with BARON version 7.4 [247] using GAMS version 22.0 [56] through system calls. Note that strictly speaking BARON does not exactly satisfy the assumption about solvers, since the final lower bound furnished may not be strictly below the optimal objective function value (it can be slightly above); since tight tolerances were used for the subproblems, this consideration does not have practical implications. The occurrence of third order monomials, e.g., x^3 caused very slow convergence of the formulated (MI)NLPs in some of the case studies. For consistency purposes, we therefore systematically encoded third order monomials as a product of a square and a linear term, e.g., $x^2 x$, and fourth order monomials as the product of two squares, e.g., $x^2 x^2$.

The logical constraints in the lower bounding problem and the complementarity conditions of the KKT-based lower bounding problem were implemented using the big-M formulation [123, 113].

Given a node X^i and a box V^k we first check if their intersection is empty. If it is ($X^i \cap V^k = \emptyset$) the constraint does not need to be introduced. Also if $X^i \subset V^k$ we directly introduce the constraint

$$h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{y}^k).$$

Otherwise up to two binary variables and constraints are introduced for each component of \mathbf{x} , as described in Subroutine 4.3. Therefore up to $2n_x + 1$ binary variables

are required to formulate a logical constraint.

Subroutine 4.3 (Implementation of Logical Constraints).

- Set $l = 0$

- **FOR** $j = 1, \dots, n_x$ **DO**

- **IF** $v_j^{k,LO} > x_j^{i,LO}$ **THEN**

- * Set $l = l + 1$ and introduce a binary variable $w_l \in \{0, 1\}$ representing

- $x_j > v_j^{k,LO}$

- * Introduce a constraint

$$w_l \geq \frac{x_j - v_j^{k,LO}}{x_j^{i,UP} - x_j^{i,LO}}.$$

- **IF** $v_j^{k,UP} < x_j^{i,UP}$ **THEN**

- * Set $l = l + 1$ and introduce a binary variable $w_l \in \{0, 1\}$ representing

- $x_j < v_j^{k,UP}$

- * Introduce a constraint

$$w_l \geq \frac{v_j^{k,UP} - x_j}{x_j^{i,UP} - x_j^{i,LO}}.$$

- Introduce the logical constraint as

$$h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{y}^k) + \sum_{i=1}^l (1 - w_i) (h^{max} - h(\mathbf{x}, \mathbf{y}^k)),$$

where $h^{max} \geq h(\mathbf{x}, \mathbf{y})$ for all $(\mathbf{x}, \mathbf{y}) \in X^i \times Y^i$. h^{max} can be easily estimated using interval analysis, see, e.g., [204]. Note that unless $w_i = 1$, for all $i = 1, \dots, l$ this constraint is redundant.

Note that the constraint $\mathbf{x} \in V^k \Rightarrow h(\mathbf{x}, \mathbf{y}) \leq h(\mathbf{x}, \mathbf{y}^k)$ is only introduced for the interior of V^k .

For the KKT-based lower bounds the complementarity condition $\mu_i \tilde{p}_i(\mathbf{x}, \mathbf{y})$ is reformulated with the help of a binary variable $w_l \in \{0, 1\}$ indicating if the constraint

is active or not

$$\begin{aligned}\mu_i &\leq M_i y_l \\ -\tilde{p}_i(\mathbf{x}, \mathbf{y}) &\leq P_i(1 - w_l),\end{aligned}\tag{4.36}$$

noting that $\tilde{p}_i(\mathbf{x}, \mathbf{y}) \leq 0$ and $\mu_i \geq 0$. By Assumption 4.4 an upper bound M_i for the KKT multipliers is known a priori. Bounds on the constraints P_i can be easily estimated by interval extension, see e.g., [204].

Test Set

To test our algorithm we used literature examples collected in [246, 133] and also created a number of test problems with mostly nonconvex inner problems. All the problem formulations are collected in Section C of the appendix, including an analysis of the feasible sets and optimal solutions as well as justifications for the values used for the KKT multipliers.

Table 4.1 contains a summary of the problem properties. The first column is the label of the example. The second through sixth columns (n_x, n_y, n_g, n_p, n_q) contain the number of \mathbf{x} variables, \mathbf{y} variables, constraints in the outer problem, constraints in the inner problem that depend on the outer variables, and constraints in the inner problem (excluding box constraints) that do not depend on the outer variables. The seventh through tenth columns (f, g, h, p) contain the functional form of the outer objective, the outer constraints, the inner objective and the inner constraints: A stands for affine linear, C stands for convex nonlinear, N stands for nonconvex nonlinear, and P stands for pseudoconvex; for the outer functions the characterization is joint in \mathbf{x} and \mathbf{y} while for the inner functions the characterization is only for the \mathbf{z} -dependence, e.g., convex means partially convex on Y . The eleventh column (h) indicates whether or not the inner objective depends on the outer variables; F stands for false (no dependence), T for true, and a dash is used for the problems without \mathbf{x} variables ($n_x = 0$). Finally, the last two columns contain the optimal solution value

and the set of optimal solutions as obtained by an analysis of the problems. Problems C.24 and C.28 have not yet been analyzed completely and the best available solution value, along with a (presumably) optimal solution is given. To emphasize that these problems have not been analyzed, a question mark is set next to the best available solution value.

Note that some of the problems do not contain any outer variables ($n_x = 0$). These problems can be easily solved by solving the inner problem and then solving an augmented outer program. They have been included because, despite their simplicity, they reveal problems with certain approaches.

Illustrative Examples

To illustrate how the algorithm works consider first Example C.10 without branching.

- Set $LBD = -\infty$ and $UBD = \infty$.
- Solve the lower bounding problem

$$\min_{x \in [0.1, 1], y \in [-1, 1]} y$$

and obtain $\bar{x} = 0.1$, $\bar{y} = -1$, $LBD = -1$.

- Solve the inner problem for $\bar{x} = 0.1$

$$\min_{z \in [-1, 1]} 0.1 \left(16z^4 + 2z^3 - 8z^2 - \frac{3}{2}z + \frac{1}{2} \right)$$

and obtain $\bar{y} = 0.5$, $h^* = -0.1$.

- $n_p = 0 \Rightarrow y^1 = 0.5$, $V^1 = [0.1, 1]$.

- Solve the upper bounding problem

$$\begin{aligned} & \min_{y \in [-1, 1]} y \\ & \text{s.t. } 0.1 \left(16z^4 + 2z^3 - 8z^2 - \frac{3}{2}z + \frac{1}{2} \right) \leq -0.1 \end{aligned}$$

and obtain $\bar{x} = 0.1$, $\bar{y} = 0.5$, $f(\bar{x}, \bar{y}) = 0.5$.

Set $UBD = 0.5$, $(x^*, y^*) = (\bar{x}, \bar{y})$.

- $UBD - LBD = 1.5$. Do not terminate.
- Solve the lower bounding problem

$$\begin{aligned} & \min_{x \in [0.1, 1], y \in [-1, 1]} y \\ & \text{s.t. } x \left(16y^4 + 2y^3 - 8y^2 - \frac{3}{2}y + \frac{1}{2} \right) \leq -0.1x \end{aligned}$$

and obtain $\bar{x} = 0.1$, $\bar{y} = 0.5$, $LBD = 0.5$.

$UBD - LBD = 0$. Terminate.

Consider now Example C.30 from [133]. Note that this example violates assumption 4.3. In deviation from Algorithm 4.1, we solve the upper bounding problem before fathoming of the node.

- Set $LBD = -\infty$ and $UBD = \infty$.
- Solve the lower bounding problem

$$\begin{aligned} & \min_{x, \mathbf{y}} x \\ & \text{s.t. } \frac{0.0332333}{z_2} + 0.1z_1 - 1 \leq 0 \\ & 4\frac{x}{z_2} + 2\frac{x^{-0.71}}{z_2} + 0.0332333x^{-1.3} - 1 \leq 0 \\ & x \in [0.1, 10] \quad \mathbf{y}, \mathbf{z} \in [0.1, 10]^2 \end{aligned}$$

and obtain $\bar{x} = 0.1936160927$, $\bar{y}_1 = 7.257797625$, $\bar{y}_2 = 10$, $LBD = 0.1936160927$.

- Solve the inner problem for $\bar{x} = 0.1936160927$

$$\begin{aligned} \min_{\mathbf{z}} & -z_1 + 0.5864z_1^{0.67} \\ \text{s.t.} & \frac{0.0332333}{z_2} + 0.1z_1 - 1 \leq 0 \\ & 4\frac{x}{z_2} + 2\frac{x^{-0.71}}{z_2} + 0.0332333x^{-1.3} - 1 \leq 0 \\ & x \in [0.1, 10] \quad \mathbf{y}, \mathbf{z} \in [0.1, 10]^2 \end{aligned}$$

and obtain $h^* = -7.230078387$, $\bar{y}_1 = 9.9667667$, $\bar{y}_1 = 10$.

- Condition (4.21) is violated. This instance does not satisfy Assumption 4.3 and Algorithm 4.1 would fail. With a slight change it converges at the root node.
- Solve the upper bounding problem

$$\begin{aligned} \min_{\mathbf{y}} & 0.1936160927 \\ \text{s.t.} & y_1 + 0.5864y_1^{0.67} \leq -7.230078387 \\ & \mathbf{y} \in [0.1, 10]^2 \end{aligned}$$

and obtain $\bar{x} = 0.1936160927$, $\bar{y}_1 = 9.9667667$, $\bar{y}_2 = 10$.

Set $UBD = 0.1936160927$, $x^* = 0.1936160927$, $y_1^* = 9.9667667$, $y_2^* = 10$.

Numerical Results

Tables 4.2, 4.3 and 4.4 contain the numerical results with the three branching heuristics presented. The branching on the \mathbf{x} variables in $2n_x + 1$ nodes is only applied to the test problems for which the resulting algorithm is different than the extreme of no branching. The optimality and feasibility tolerances for BARON were set to $\varepsilon_{NLP} = 10^{-6}$ for all problems. For most problems the optimality tolerance for the inner problem was set to $\varepsilon_h = 10^{-5}$ and the absolute and relative termination criteria to $\varepsilon_f = 10^{-4}$. For some of the literature problems, when the KKT heuristic is not used for the lower bound, the computational requirement is quite high, and the tolerances

were set according to a tradeoff between accuracy and computational time. For these problems the solution is repeated with the KKT-based lower bounds and the default tolerances. Note that for all problems the tolerances used satisfy the assumptions in Lemma 4.1. Example C.30 violates assumption 4.3, but solving the upper bounding problem at the root node gives the same value as the lower bounding problem and the algorithm converges at the root node.

The first column (Label) has the label of the problem, while the second (f^*) and third ($\mathbf{x}^*, \mathbf{y}^*$) the optimal objective value and set of optimal solutions respectively, obtained by analysis. The fourth column (μ^{max}) contains the maximal values for KKT multipliers used, and the fifth (ε_h) the optimality tolerance for the inner program. For some problems a sequence of decreasing ε_{h2} to obtain the points y^k was used in step 7; the starting value is given in the sixth column (ε_{h2}^0). This value was decreased by a factor of 1.1 at each iteration down to $0.8\varepsilon_h$. For the rest of the problems $\varepsilon_{h2} = 0.8\varepsilon_h$ was used for all iterations. The seventh column (ε_f) contains the termination tolerance (absolute and relative termination criteria were set equal). To guess the boxes V^k , a decreasing sequence was used; each time the interval diameter was set to one and decreased by a factor of 0.9 until the interval extensions showed feasibility. Natural interval extensions were used.

The eighth through fourteenth columns contain the results obtained with the use of simple lower bounds while the fifteenth through twenty-first column shows the results obtained with the use of the KKT heuristic for the lower bounds; \bar{f} shows the optimal objective value obtained; $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ shows the optimal solution obtained; UBD shows the node at which the optimal solution was first obtained; #UBD shows the number of upper bounding calls; #LBD shows the number of lower bounding calls; the first time columns show the sum of CPU time reported by GAMS and spent in the main program, while the second time columns show the time obtained by the timing function. Note that there is a significant difference between these two times requirements, presumably due to the system calls and processing time for GAMS. Because some CPU times are very small, we present an average of 10 runs.

Table 4.1: Summary of problem properties

| Label | n_x | n_y | n_g | n_p | n_q | f | g | h | p | h | f^* | $\mathbf{x}^*, \mathbf{y}^*$ |
|-----------|-------|-------|-------|-------|-------|-----|-----|-----|-----|-----|----------|------------------------------|
| mb_0_1_01 | 0 | 1 | 0 | 0 | 0 | A | - | A | - | - | 1 | 1 |
| mb_0_1_02 | 0 | 1 | 1 | 0 | 0 | A | A | A | - | - | ∞ | none |
| mb_0_1_03 | 0 | 1 | 0 | 0 | 1 | A | - | C | N | - | -1 | -1 |
| mb_0_1_04 | 0 | 1 | 0 | 0 | 0 | A | - | N | - | - | 1 | 1 |
| mb_0_1_05 | 0 | 1 | 0 | 0 | 0 | A | - | N | - | - | 0.5 | 0.5 |
| mb_0_1_06 | 0 | 1 | 0 | 0 | 0 | A | - | N | - | - | -1 | -1 |
| mb_0_1_07 | 0 | 1 | 1 | 0 | 0 | A | A | A | - | - | ∞ | none |
| mb_1_1_01 | 1 | 1 | 2 | 0 | 0 | C | A | A | - | T | 0 | -0.567,0 |
| mb_1_1_02 | 1 | 1 | 1 | 0 | 0 | A | A | N | - | F | -1 | -1,-1 |
| mb_1_1_03 | 1 | 1 | 0 | 0 | 0 | A | - | N | - | T | 0.5 | [0.1,1],0.5 |
| mb_1_1_04 | 1 | 1 | 0 | 0 | 0 | A | - | N | - | T | -0.8 | 0,-0.8 |
| mb_1_1_05 | 1 | 1 | 0 | 0 | 0 | N | - | N | - | T | 0 | 0,0 |
| mb_1_1_06 | 1 | 1 | 0 | 0 | 0 | A | - | N | - | T | -1 | 0,1 |
| mb_1_1_07 | 1 | 1 | 0 | 0 | 0 | C | - | N | - | T | 0.25 | 0.25,0.5 |
| mb_1_1_08 | 1 | 1 | 0 | 0 | 0 | A | - | N | - | T | 0 | -1,1 |
| mb_1_1_09 | 1 | 1 | 0 | 0 | 0 | A | - | N | - | T | -2 | -1,0 and -0.5,-1 |
| mb_1_1_10 | 1 | 1 | 0 | 0 | 0 | C | - | N | - | T | 0.1875 | -0.25, \pm 0.5 |
| mb_1_1_11 | 1 | 1 | 0 | 0 | 0 | N | - | N | - | T | 0.25 | 0.5,0 |
| mb_1_1_12 | 1 | 1 | 0 | 0 | 0 | N | - | N | - | T | -0.258 | 0.18858,-0.76759 |
| mb_1_1_13 | 1 | 1 | 0 | 0 | 0 | C | - | N | - | T | 0.3125 | 0.5,0.5 |
| mb_1_1_14 | 1 | 1 | 0 | 0 | 0 | C | - | N | - | T | 0.2095 | -0.5544,0.4554 |
| mb_1_1_15 | 1 | 1 | 0 | 1 | 0 | C | - | N | N | T | 0.2095 | -0.5544,0.4554 |
| mb_1_1_16 | 1 | 1 | 1 | 1 | 0 | C | N | C | C | F | 0.1756 | -0.4191,-1 |
| mb_2_3_01 | 2 | 3 | 2 | 1 | 2 | N | N | N | N | T | -1 ? | -1,-1,-1,1,1 |
| mb_2_3_02 | 2 | 3 | 3 | 0 | 0 | N | N | N | - | T | -2.3535 | -1,-1,1,1,-0.707 |
| gf_1 | 1 | 1 | 1 | 1 | 0 | C | A | C | A | T | 2250 | 11.25,5 |
| gf_2 | 1 | 2 | 0 | 2 | 0 | N | A | A | C | F | 1 | 1,0,1 |
| gf_3 | 2 | 3 | 0 | 2 | 1 | A | - | P | A | T | -29.2 ? | 0,0.9,0,0.6,0.4 |
| gf_4 | 1 | 1 | 3 | 0 | 0 | C | A | C | - | F | 9 | 3,5 |
| gf_5 | 1 | 1 | 0 | 1 | 1 | A | - | N | N | F | 0.193616 | 0.1936,9.9667,10 |
| sc_1 | 1 | 2 | 0 | 3 | 0 | A | - | A | A | F | -13 | 5,4,2 |
| sc_2 | 1 | 1 | 0 | 3 | 0 | C | - | C | A | F | 5 | 1,3 |

Table 4.2: Numerical results without branching

| Label | f^* | $\mathbf{x}^*, \mathbf{y}^*$ | μ^{max} | ϵ_h | ϵ_{h2}^0 | ϵ_f | f | $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ | UBD | #UBD | #LBD | time | f | $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ | UBD | #UBD | #LBD | time |
|-----------|----------|------------------------------|-------------|--------------|-------------------|--------------|----------|--------------------------------------|-----|------|------|-----------|----------|--------------------------------------|-----|------|-----------|-----------|
| mb_0_1_01 | 1 | 1 | 2 | 10^{-5} | NA | 10^{-4} | 1.00 | 1.00 | 1 | 1 | 2 | 0.06 0.29 | 1 | 1 | 1 | 1 | 0.05 0.23 | |
| mb_0_1_02 | ∞ | - | 2 | 10^{-5} | NA | 10^{-4} | ∞ | | 0 | 1 | 2 | 0.04 0.29 | ∞ | | 0 | 0 | 1 | 0.00 0.08 |
| mb_0_1_03 | -1 | -1 | 2 | 10^{-5} | NA | 10^{-4} | -1.00 | -1.00 | 1 | 1 | 2 | 0.06 0.30 | -1.00 | -1.00 | 1 | 1 | 1 | 0.07 0.26 |
| mb_0_1_04 | 1 | -1 | 2 | 10^{-5} | NA | 10^{-4} | 1.00 | 1.00 | 1 | 1 | 2 | 0.06 0.30 | 1.00 | 1.00 | 1 | 1 | 2 | 0.05 0.29 |
| mb_0_1_05 | 0.5 | 0.5 | 2 | 10^{-5} | NA | 10^{-4} | 0.500 | 0.500 | 1 | 1 | 2 | 0.13 0.37 | 0.500 | 0.500 | 1 | 1 | 2 | 0.08 0.32 |
| mb_0_1_06 | -1 | -1 | 10 | 10^{-5} | NA | 10^{-4} | -1 | -1 | 1 | 1 | 1 | 0.04 0.23 | -1 | -1 | 1 | 1 | 1 | 0.05 0.23 |
| mb_0_1_07 | ∞ | - | 2 | 10^{-5} | NA | 10^{-4} | ∞ | | 0 | 1 | 2 | 0.03 0.29 | ∞ | | 0 | 0 | 1 | 0.00 0.08 |
| mb_1_1_01 | 0 | -0.567,0 | 5 | 10^{-5} | NA | 10^{-4} | 0 | -0.567,0 | 2 | 2 | 2 | 0.08 0.43 | 0 | -0.567,0 | 1 | 1 | 1 | 0.05 0.23 |
| mb_1_1_02 | -1 | -1,-1 | 10 | 10^{-5} | NA | 10^{-4} | -1 | -1 -1 | 1 | 1 | 1 | 0.04 0.22 | -1 | -1 -1 | 1 | 1 | 1 | 0.14 0.33 |
| mb_1_1_03 | 0.5 | [0,1],0.5 | 2 | 10^{-5} | NA | 10^{-4} | 0.498 | 0.1, 0.498 | 1 | 1 | 2 | 2.0 2.2 | 0.498 | 0.100,0.498 | 1 | 1 | 2 | 0.12 0.36 |
| mb_1_1_04 | -0.8 | 0,-0.8 | 100 | 10^{-5} | NA | 10^{-4} | -0.8 | 0 -0.8 | 1 | 1 | 1 | 0.04 0.22 | -0.8 | 0 -0.8 | 1 | 1 | 1 | 0.05 0.22 |
| mb_1_1_05 | 0 | 0,0 | 2 | 10^{-5} | NA | 10^{-4} | -0.004 | -0.004,0.00 | 9 | 9 | 9 | 4.4 6.0 | -0.004 | 0.004,0.00 | 9 | 9 | 9 | 0.87 2.5 |
| mb_1_1_06 | -1 | 0,1 | 2 | 10^{-5} | NA | 10^{-4} | -1.006 | -0.006,1 | 2 | 2 | 2 | 0.12 0.47 | -1.006 | -0.006,1 | 2 | 2 | 2 | 0.13 0.53 |
| mb_1_1_07 | 0.25 | 0.25,0.5 | 10 | 10^{-5} | NA | 10^{-4} | 0.246 | 0.250,0.496 | 1 | 3 | 4 | 0.25 0.84 | 0.250 | 0.25,0.500 | 2 | 2 | 2 | 0.13 0.49 |
| mb_1_1_08 | 0 | -1,1 | 2 | 10^{-5} | NA | 10^{-4} | 0.00 | -1,1.00 | 1 | 1 | 2 | 0.08 0.31 | 0.00 | -1,1.00 | 1 | 1 | 2 | 0.09 0.33 |
| mb_1_1_09 | -2 | (-1,0),(-0.5,-1) | 2 | 10^{-5} | NA | 10^{-4} | -2.005 | -1,-0.0045 | 1 | 1 | 2 | 0.15 0.39 | -2.0045 | -1,-0.0045 | 1 | 1 | 2 | 0.06 0.30 |
| mb_1_1_10 | 0.1875 | -0.25, \pm 0.5 | 2 | 10^{-5} | NA | 10^{-4} | 0.185 | -0.250,-0.494 | 2 | 18 | 19 | 2.8 6.1 | 0.1845 | -0.250,-0.494 | 2 | 3 | 4 | 0.27 0.87 |
| mb_1_1_11 | 0.25 | 0.5,0 | 2 | 10^{-5} | NA | 10^{-4} | 0.250 | 0.5,0.00 | 2 | 2 | 2 | 0.10 0.45 | 0.2500 | 0.500,0.00 | 2 | 2 | 2 | 0.11 0.46 |
| mb_1_1_12 | -0.258 | 0.189,0.768 | 2 | 10^{-5} | NA | 10^{-4} | -0.260 | 0.1874,0.438 | 10 | 12 | 13 | 1.7 3.9 | -0.258 | 0.189,0.434 | 2 | 2 | 2 | 0.15 0.52 |
| mb_1_1_13 | 0.3125 | 0.5,0.5 | 2 | 10^{-5} | NA | 10^{-4} | 0.310 | 0.501,0.497 | 3 | 3 | 4 | 0.26 0.84 | 0.313 | 0.500,0.500 | 2 | 2 | 2 | 0.67 1.0 |
| mb_1_1_14 | 0.2095 | -0.554,0.454 | 10 | 10^{-5} | NA | 10^{-4} | 0.200 | -0.500,0.447 | 1 | 2 | 3 | 0.31 0.73 | 0.202 | -0.500,0.450 | 2 | 2 | 2 | 0.23 0.59 |
| mb_1_1_15 | 0.2095 | -0.554,0.454 | 10 | 10^{-5} | NA | 10^{-4} | 0.207 | -0.567,0.454 | 5 | 5 | 5 | 1.1 2.3 | 0.2095 | -0.554,0.455 | 1 | 1 | 1 | 0.28 0.53 |
| mb_1_1_16 | 0.1756 | -0.4191,-1 | NA | 10^{-5} | NA | 10^{-4} | -0.4191 | -1.000 | 2 | 2 | 3 | 0.08 0.60 | NA | NA | NA | NA | NA | NA |
| mb_2_3_01 | -1? | | 20 | 10^{-5} | NA | 10^{-4} | -1.00 | 1,-1,-0.00,-1,-1 | 1 | 1 | 2 | 0.38 0.69 | -1 | -1,-1,-1,1,1 | 1 | 1 | 1 | 0.25 0.51 |
| mb_2_3_02 | -2.3535 | -1,-1,1,1,-0.707 | 10 | 10^{-5} | NA | 10^{-4} | -2.35 | -1,-1,1,1.00,-0.707 | 2 | 2 | 3 | 0.17 0.59 | -2.35 | -1,-1,1,1.00,-0.707 | 1 | 1 | 2 | 0.16 0.40 |
| gf_1 | 2250 | 11.25,5 | 10^4 | 0.001 | 500 | 0.1 | 2250 | 11.24,5.028 | 42 | 42 | 43 | 86 97 | 2250 | 11.25,5.00 | 1 | 1 | 1 | 0.32 0.56 |
| gf_2 | 1 | 1,0,1 | 100 | 10^{-5} | 0.1 | 0.15 | 1.00 | 1,-0.003,1.00 | 1 | 20 | 21 | 2.2 7.6 | 1.00 | 1,0,1.00 | 1 | 1 | 1 | 0.11 0.35 |
| gf_3 | -29.2? | 0,0.9,0,0.6,0.4 | 10^3 | 0.1 | 1 | 0.5 | -30.3 | 0,0.38,0.515,0.768,0.042 | 15 | 15 | 16 | 20 24 | -29.2 | 0,0.9,0,0.6,0.4 | 1 | 1 | 1 | 0.56 0.80 |
| gf_4 | 9 | 3,5 | NA | 10^{-5} | NA | 10^{-4} | 9.00 | 3,5.00 | 2 | 2 | 2 | 0.10 0.46 | 9 | 3,5.00 | 1 | 1 | 1 | 0.05 0.23 |
| gf_5 | 0.1936 | 0.1936,9.9667,10 | 10 | 10^{-5} | NA | 10^{-4} | 0.194 | 0.194,9.97,10 | 1 | 1 | 1 | 0.07 0.32 | 0.194 | 0.194,9.97,10 | 1 | 1 | 1 | 0.08 0.33 |
| sc_1 | -13 | 5,4,2 | 10 | 10^{-5} | NA | 10^{-4} | -13 | 5 4 2 | 1 | 1 | 1 | 0.05 0.29 | -13 | 5,4,2 | 1 | 1 | 1 | 0.06 0.30 |
| sc_2 | 5 | 1,3 | 10 | 10^{-5} | 2 | 0.01 | 5.00 | 0.994,3.00 | 40 | 52 | 53 | 17 31 | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.13 0.37 |
| | | | | 10^{-5} | NA | 10^{-4} | | | | | | | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.13 0.37 |

Table 4.4: Numerical results with special branching on the \mathbf{x} variables

| Label | f^* | $\mathbf{x}^*, \mathbf{y}^*$ | μ^{max} | ε_h | ε_{h2}^0 | ε_f | f | $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ | UBD | #UBD | #LBD | time | f | $\bar{\mathbf{x}}, \bar{\mathbf{y}}$ | UBD | #UBD | #LBD | time |
|-----------|--------|------------------------------|-------------|-----------------|----------------------|-----------------|---------|--------------------------------------|-----|------|------|-----------|--------|--------------------------------------|-----|------|------|-----------|
| mb_1_1_15 | 0.2095 | -0.554,0.454 | 10 | 10^{-5} | NA | 10^{-4} | 0.2069 | -0.567,0.454 | 5 | 5 | 5 | 1.1 2.3 | 0.2095 | -0.554,0.455 | 1 | 1 | 1 | 0.28 0.52 |
| mb_2_3_01 | -1? | | 20 | 10^{-5} | NA | 10^{-4} | -1.00 | 1,-1,-0.003,-1,-1 | 1 | 1 | 2 | 0.38 0.68 | -1 | -1,-1,-1,1,1 | 1 | 1 | 1 | 0.26 0.50 |
| mb_1_1_16 | 0.1756 | -0.4191,-1 | NA | 10^{-5} | NA | 10^{-4} | -0.4191 | -1.000 | 2 | 2 | 3 | 0.08 0.70 | NA | NA | NA | NA | NA | NA |
| gf_1 | 2250 | 11.25,5 | 10^4 | 0.001 | 500 | 0.1 | 2250 | 11.18,5.30 | 19 | 41 | 49 | 3.4 14 | 2250 | 11.25, 5.00 | 1 | 1 | 1 | 0.32 0.56 |
| gf_2 | 1 | 1,0,1 | 100 | 10^{-5} | 0.1 | 0.15 | 0.997 | 1,-0.003,1.00 | 1 | 17 | 23 | 1.2 5.6 | 2250 | 11.25,5.00 | 1 | 1 | 1 | 0.32 0.56 |
| gf_3 | -29.2? | 0,0.9,0,0.6,0.4 | 10^3 | 0.1 | 1 | 0.5 | -29.9 | 0,0.27,0.554,0.777,0 | 4 | 19 | 42 | 6.9 15 | 1.00 | 1,0,1.00 | 1 | 1 | 1 | 0.11 0.35 |
| gf_5 | 0.1936 | 0.1936,9.9667,10 | 10 | 10^{-5} | NA | 10^{-4} | 0.194 | 0.194, 9.97,10 | 1 | 1 | 1 | 0.07 0.32 | 1.00 | 1,0,1.00 | 1 | 1 | 1 | 0.10 0.35 |
| sc_1 | -13 | 5,4,2 | 10 | 10^{-5} | NA | 10^{-4} | -13 | 5, 4, 2 | 1 | 1 | 1 | 0.05 0.32 | -29.2 | 0, 0.9, 0, 0.6, 0.4 | 1 | 1 | 1 | 0.56 0.81 |
| sc_2 | 5 | 1,3 | 10 | 10^{-5} | 2 | 0.01 | 5.00 | 1.00,3.00 | 32 | 65 | 125 | 4.0 24.7 | -29.2 | 0, 0.9, 0, 0.6, 0.4 | 1 | 1 | 1 | 0.56 0.85 |
| | | | 10 | 10^{-5} | NA | 10^{-4} | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.09 0.33 | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.09 0.33 |
| | | | 10 | 10^{-5} | NA | 10^{-4} | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.05 0.30 | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.05 0.30 |
| | | | 10 | 10^{-5} | NA | 10^{-4} | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.13 0.37 | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.13 0.37 |
| | | | 10 | 10^{-5} | NA | 10^{-4} | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.13 0.37 | 5.00 | 1.00,3.00 | 1 | 1 | 1 | 0.13 0.37 |

Conclusions from Numerical Experiments

All the test problems were solved without significant numerical difficulties. For sufficiently small tolerances the solution furnished approaches the optimal solution. As in single-level optimization appropriate choice of tolerances is necessary for computational efficiency and accuracy of solutions. Moreover, the computational requirements are greatly dependent on the problem structure.

The KKT-based heuristic for the lower bound, when applicable, greatly reduces the number of iterations needed, and moreover since it gives a tighter bound, the upper bounding procedure is less likely to produce points far away from (absolutely) feasible points. On the other hand, the cost per iteration is higher because of the complementarity conditions. With the use of the KKT-based heuristic for the lower bound, the computational requirement for all problems and all three heuristics was less than 1 second.

As expected, when no branching is performed, the number of iterations is smaller, but the cost per iteration is higher. For the test set considered, there is no clear advantage of one branching heuristic over the other, which indicates that the optimal branching heuristic depends on the problem. As a general guideline for problems that are solved in few iterations no branching is advantageous. For larger problems, the elaborate branching to $2n_x + 1$ nodes is expected to outperform the other heuristics because no logical constraints are needed. Finally, for any given problem, typically either the elaborate branching or no branching will outperform the regular branching.

4.7 Conclusions and Future Work

An algorithm for the global solution of bilevel programs involving nonconvex functions was presented. The novelty is that nonconvexity in the inner program is permitted and a guaranteed global solution is obtained. Finite termination of the algorithm to an ε -optimal solution is proved. An implementation is described and tested on a number of original and literature test problems.

Several alterations to the algorithm and implementation are conceivable. An

interesting alternative to the MINLP formulation of the KKT-based lower bounds is an MPEC formulation [262, 260]. Also, different global MINLP algorithms could be applied, such as outer approximation [169]. Furthermore, in the case of regular branching, the introduction of the logical constraints to the lower bounding problems could be deferred until the node size is such that the parametric upper bound to the inner problem is valid for the entire node ($X^i \subset V^k$). Preliminary experimentation shows that this is not advantageous.

There are many alternatives to the implementation of logical constraints described in Subroutine 4.3. For instance, the number of binary variables used can be reduced to at most n_x for each logical constraint, by the use of nonconvex nonlinear constraints. For instance

$$w_j \geq 1 - \frac{\left(2x_j - v_j^{k,LO} - v_j^{k,UP}\right)^2}{\left(v_j^{k,UP} - v_j^{k,LO}\right)^2}$$

enforces $w_j = 1$ if $x_j \in [v_j^{k,LO}, v_j^{k,UP}]$. Alternatively, instead of introducing the constraint for the entire box V^k , it may be advantageous to introduce it for an inscribed ellipsoid using a single binary variable

$$w \geq 1 - \sum_{j=1}^{n_x} \frac{\left(2x_j - v_j^{k,LO} - v_j^{k,UP}\right)^2}{\left(v_j^{k,UP} - v_j^{k,LO}\right)^2}.$$

Finally there are alternatives to the big-M formulation such as the convex hull formulation, see e.g., [130].

The algorithm was presented in a branch-and-bound framework. A more general alternative is to embed it in a generalized branch-and-cut framework, such as the one described in [171] for nonconvex MINLPs. Also, other branching heuristics could be introduced, such as branching only on a subset of the variables (the complicating variables in some sense) or branching on the inner variables for the lower bounding problem (using Y^i) but not for the upper bounding problem (keeping Y). In single-level programs algorithms incorporating domain reduction show significant performance enhancements over pure branch-and-bound [266]. It would be interesting

to also consider domain reduction for bilevel programs.

The algorithm presented here relies on the global solution of the subproblems, resulting in a nested exponential procedure, and it would be interesting to, at least partially, eliminate this nested procedure. Currently, no alternative other than global solution of the inner problem exists to obtain or confirm an upper bound, but an obvious possibility is to solve the upper bounding problem only periodically. On the other hand, the lower bounding problem could be solved locally to obtain a candidate $\bar{\mathbf{x}}$ and a convex relaxation of the lower bounding problem (4.25) could be solved to obtain the lower bound. Preliminary experimentation with these ideas showed slower convergence, but for problems of large size they may be beneficial. Also, to obtain points \mathbf{y}^k , the solution of simpler programs than (4.28) is conceivable.

The ideas presented here could be extended to address some related programs such as flexibility problems [131], semi-infinite programs, and bilevel programs involving binary variables. To that extent the algorithmic ideas presented here could be combined with different approaches, such as methods based on interval-extensions [181, 44, 45]. Finally, the extension to equality-constrained inner programs is of interest.

Chapter 5

Conclusions and Future Work

As mentioned in the introduction, this thesis consists of essentially three parts. In each of these parts conclusions are presented and scope for future work in the respective areas is identified. The purpose of this chapter is to provide an overall conclusion and some possibilities of further connections between the three parts.

In the first part an integrated design methodology for portable power generation based on fuel cell systems is proposed. This methodology is based on a decomposition into three levels of modeling detail, namely system-level models for process synthesis, intermediate fidelity models for optimization of component sizes and operation, and computational fluid dynamics detailed models for geometric design. This methodology can be extended to devices based on other power generation mechanisms, such as oxidation of a fuel in conjunction with a thermophotovoltaic element. Moreover it is interesting to consider the extension to more general microchemical systems. Finally, considering other chemical products that are processes is interesting.

In the second part algorithms for parametric mixed-integer programming are developed. Mathematical programs often involve unknown parameters and the task of parametric optimization is, in principle, to solve the mathematical program for all possible values of these unknown parameters. In this thesis algorithms for the general case, where a parameter affects the cost-vector, right-hand side vectors and matrix, and the multiparametric cost vector case are proposed, implemented and tested. To our best knowledge, for either problem no algorithms exist in the open literature.

Extensions of interest are the multiparametric general case and the nonlinear case.

Finally, the co-operative formulation of bilevel programs with nonconvex functions is considered. Most literature contributions for bilevel programs consider convex inner programs and the algorithms available are only applicable to bilevel programs with convex inner programs, or furnish a semi-local solution, i.e., a local solution in the inner and outer programs. We first identify consequences of nonconvexity in the inner program and then propose an algorithm based on a series of global single-level optimization problems.

The main motivation for the development of parametric optimization algorithms within this thesis is the allocation of R&D resources in micropower, in particular the identification of the components in micropower generation devices that are the most important to develop. Due to limitations in the simulation and optimization tools used, two instances for the models are implemented, one for simulation and one for parametric optimization. To take full advantage of the algorithms developed it is of paramount importance to overcome these limitations and use the same modeling framework. This would facilitate case studies and provide insight into the most important components. Moreover the extension to case studies with more than one unknown parameter would be very interesting. As described in Chapter 3 the development of a general-purpose algorithm for multiparametric optimization is very difficult, but the development of heuristic-based algorithms for the special case of two parameters in the resource allocation formulation seems tractable. A potential basis for this algorithm are the formulations for the subproblems of identifying a candidate solution and probing the feasibility and optimality region.

Bilevel programs are considered within this thesis, mainly due to the inherent and well-known relation of bilevel programming and parametric optimization. Regardless of that, bilevel programs are useful design tools and our algorithm could thus be used in conjunction with our design methodology for micropower generation devices. One possibility is to consider some components that are limited by chemical or phase equilibrium considerations. Another interesting application of two-stage formulations is to consider the flexibility and robustness of designs.

Appendix A

Modeling Details

A.1 Appendix: Symbols Used

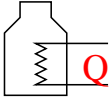
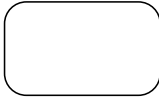
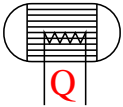





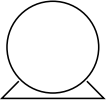

| | | | |
|---|----------------------|--|-----------------|
|  | Burner |  | Feed |
|  | Reactor |  | Design decision |
|  | Component separation |  | Compressor |
|  | Stream splitter |  | Fuel cell |
|  | Pump |  | Flash |

Figure A-1: Explanation of the symbols used.

A.2 Appendix: Nomenclature

Table A.1: Nomenclature

| Abbrev. | Property | Unit |
|------------------|---|---------------------------------|
| A | Surface | m^2 |
| d | Cartridge inner diameter | m |
| e_{grav}^{bat} | Gravimetric energy density of auxiliary battery | Wh/kg |
| e_{vol}^{bat} | Volumetric energy density of auxiliary battery | Wh/l |
| F | Faraday constant | C/mol |
| G_i° | Molar Gibbs free energy of pure component i at reference pressure | J/mol |
| $\Delta_r G$ | Gibbs free energy of reaction r at system pressure | J/mol |
| $H_{i,j}$ | Molar enthalpy of component i in stream j | J/mol |
| k_i^{th} | Heat conductivity of species i | $\frac{W}{mK}$ |
| k_i^{trns} | Transport coefficient for component i | $\frac{mol}{s} / \frac{mol}{s}$ |
| K_C | Compressor parameter | J/mol/K |
| K_P | Pump parameter | J/l |
| M | Mass | kg |
| MW_i | Molar Weight of component i | kg/mol [234] |
| m_i | Mass of Component i | kg |
| n_i | Mol number of component i | mol |
| n_{i0} | Initial mol number of component i | mol |
| $N_{i,j}$ | Molar flow of component i of stream j | mol/s |
| $N_{i,diff}$ | Diffusion Rate of Component i | mol/s |
| P | Pressure | bar |
| P_{ref} | Reference pressure | bar |
| PW | Power | W |
| Q_{loss} | Heat losses | W |
| Q_p | Heat load of unit p | W |
| $Q_{i,in}$ | Heat Exchange into Unit i | W |
| $Q_{i,out}$ | Heat Exchange out of Unit i | W |
| $Q_{distant}$ | Heat exchange from distant units | W |
| R | Gas constant | J/mol/K |
| t | Cartridge wall thickness | m |
| T | Temperature | K |
| T_{amb} | Ambient Temperature | K |
| u | Velocity | m/s |
| U | Voltage | V |
| U_{loss} | Overall heat loss coefficient | W/m ² /K |
| V | Volume | m ³ |
| w_i | Weight Fraction of Component i | kg/kg |
| α | Splitratio | $\frac{mol}{s} / \frac{mol}{s}$ |
| ϵ | Product of emissivity and view factor | $\frac{W}{m^2} / \frac{W}{m^2}$ |
| η_{diff} | Diffusion Efficiency | $\frac{mol}{s} / \frac{mol}{s}$ |
| η_{ads} | Adsorption Efficiency | $\frac{mol}{s} / \frac{mol}{s}$ |
| η_M | Membrane efficiency | $\frac{mol}{s} / \frac{mol}{s}$ |
| η_{FC} | Fuel cell efficiency | W/W |
| ζ_r | Conversion of reaction r | mol/mol |
| $\nu_{i,j}$ | Stoichiometric Coefficient of Component i in Reaction j | |
| ξ_r | Extent of reaction r | mol/s |
| ρ_i | Density of Component i | kg/m ³ |
| σ_A | Maximal allowable tensile stress | Pa |
| τ | Residence time | s |
| $\tau_{mission}$ | Mission duration (time between refueling) | s |
| $\tau_{startup}$ | Process startup time | s |
| Φ | Air ratio | $\frac{mol}{s} / \frac{mol}{s}$ |
| χ_{temp} | Temperature Loss Factor | K/K |
| χ_{heat} | Heat Exchange Loss Factor | W/W |
| Ψ | Water Excess Coefficient | $\frac{mol}{s} / \frac{mol}{s}$ |

A.3 Appendix: Physical Properties

Ideal gas phase and ideal solution are assumed. The reference for enthalpy and entropy is taken at $T_{ref} = 298$ K, $P_{ref} = 1$ bar and elements in their standard states. Two different models are being used, one with mean heat capacity and a simplified vapor liquid equilibrium (VLE) and one with polynomial heat capacity and a more elaborate VLE.

The entropy of formation S_i^f is calculated as

$$S_i^f = \frac{H_i^f - G_i^f}{T_{ref}}.$$

If mean heat capacities are used the molar enthalpy and entropy in the gas phase are calculated as

$$\begin{aligned} H_i^g(T) &= H_i^f + C_{PM,i} (T - T_{ref}) \\ S_i^g(T) &= S_i^f + C_{PM,i} \ln \left(\frac{T}{T_{ref}} \right). \end{aligned}$$

Alternatively a 4th-order polynomial with two temperature intervals $[T_{min}, T_{int}]$, $[T_{int}, T_{max}]$ is used for the heat capacity

$$\begin{aligned} H_i^g(T) &= H_i^f + C_{PA,i} (T - T_{ref}) + C_{PB,i} \frac{(T^2 - T_{ref}^2)}{2} + C_{PC,i} \frac{(T^3 - T_{ref}^3)}{3} \\ &+ C_{PD,i} \frac{(T^4 - T_{ref}^4)}{4} + C_{PE,i} \frac{(T^5 - T_{ref}^5)}{5} \\ S_i^g(T) &= S_i^f + C_{PA,i} \ln \left(\frac{T}{T_{ref}} \right) + C_{PB,i} (T - T_{ref}) + C_{PC,i} \frac{(T^2 - T_{ref}^2)}{2} \\ &+ C_{PD,i} \frac{(T^3 - T_{ref}^3)}{3} + C_{PE,i} \frac{(T^4 - T_{ref}^4)}{4} \end{aligned}$$

The Gibbs free energy of the pure components in the gas phase is then calculated as

$$G_i^g = H_i^g - T S_i^g.$$

The vaporization enthalpy for the condensible components H₂O, CH₃OH is assumed

linear in the temperature. This approach, instead of using an exact calculation of the vaporization enthalpy, e.g., by the Antoine equation, has the advantage of much easier convergence and is a valid approximation for atmospheric pressure, since vaporization takes place in a very narrow temperature range.

$$\Delta_{vap}H_i(T) = \Delta_{vap}H_i(T_{lo}) + \frac{T - T_{lo}}{T_{up} - T_{lo}} (\Delta_{vap}H_i(T_{up}) - \Delta_{vap}H_i(T_{lo}))$$

For the other components the enthalpy of vaporization is irrelevant and was arbitrarily set to zero.

There is a choice between simplified and more elaborate vapor liquid equilibrium (VLE). In the simplified VLE if the temperature is lower than the boiling temperature of the component the component is assumed liquid, otherwise gaseous. In order to avoid a discontinuity a transition region of 0.1K is used.

$$\begin{aligned} T_{boil,i} < T &\Rightarrow H_i(T) = H_i^g(T) \\ T_{boil,i} - 0.1 \leq T \leq T_{boil,i} &\Rightarrow H_i(T) = H_i^g(T) - \Delta_{vap}H_i(T) \frac{T_{boil,i} - T}{0.1} \\ T < T_{boil,i} - 0.1 &\Rightarrow H_i(T) = H_i^g(T) - \Delta_{vap}H_i(T). \end{aligned}$$

For the more elaborate VLE T-P flash calculations are considered according to [172, p. 75]. Because of the assumption of ideal solution and ideal gas phase the equilibrium condition for the 2-phase region can be represented as $Py_i = P_{sat,i}(T) x_i$ and the T-P flash calculations contain only one unknown, namely the vapor fraction. In the case of recycle streams, mixing of streams is needed and H-P flashes as described in [172, p. 80] are used. The wide boiling scheme is used where the vapor fraction is calculated in an inner loop similar to the T-P flash and an outer loop is performed for the temperature. A modification was done where the stepsize for the temperature is adjusted to be strictly decreasing as the iteration proceeds.

A.4 Calculation of Energy Densities

When comparing the devices considered with batteries in terms of energy densities, it is necessary to consider the whole system, namely the device including the process units, the packaging, the fuel and the fuel cartridge. We sometimes considered only the *fuel* energy density, defined as the power output over the gravimetric or volumetric flowrate, e.g., $\frac{PW}{\sum_{i,j} MW_i N_{i,feedj}}$. This metric has the advantage that it is independent of the mission duration, but is only a valid approximation to the *system* energy density for long mission durations when the fuel size dominates over the device size.

In order to calculate the system size (mass and volume) the mission duration $\tau_{mission}$, i.e., the time period between refueling, needs to be specified. The required fuel volume/mass are the product of the mission duration and the volumetric/gravimetric flowrate. For supercritical components (N₂, O₂, H₂) the molar volume is calculated assuming ideal gas at a given storage pressure, while for subcritical components the liquid molar volume is used. For the supercritical components the storage pressure is a specified parameter, while for subcritical components it is calculated as a function of the ambient temperature. A cylindrical cartridge of a given aspect ratio is assumed. For liquids and solids a constant thickness is assumed, while for gases and subcritical components the thickness t is calculated according to the ASME Boiler and Pressure Vessel Code [226, 554-555]

$$t = \frac{P/\sigma_A}{1 - 0.6 P/\sigma_A} \frac{d}{2} + t_{min}, \quad P/\sigma_A \leq 0.385$$

$$t = \left(\sqrt{\frac{1 + P/\sigma_A}{1 - P/\sigma_A}} - 1 \right) \frac{d}{2} + t_{min}, \quad P/\sigma_A > 0.385,$$

where d is the inner diameter of the cartridge and σ_A is the maximal allowable tensile stress, an input parameter. For the special cases of hydrogen and oxygen generators a different calculation is undertaken, described in the respective subsections.

During the mission duration the mass of the fuel is constantly decreasing, whereas typical batteries have a constant mass, and the mass of zinc-air batteries increases with time as the metal is oxidized. We calculate both the initial and the average

(1/2 the initial) fuel mass. We assume a rigid fuel cartridge and therefore the whole cartridge volume is considered. The device volume is estimated by calculating the necessary volume for the units and multiplying with a factor, order of magnitude 2 – 10, accounting for the volume necessary for the packaging of the devices. The device mass is then calculated with a given density.

It is most likely that the devices will be coupled with a relatively small, most likely rechargeable battery, for start-up and shut-down operations; the function of the battery will be to provide the required power and possibly to heat the fuel cell and reactor stack. We calculate the battery size based on a given start-up time period $\tau_{startup}$ and energy densities

$$\begin{aligned} M^{bat} &= PW \tau_{startup} / e_{grav}^{bat} \\ V^{bat} &= PW \tau_{startup} / e_{vol}^{bat} \end{aligned}$$

A.5 Component Models

A.5.1 Splitter

Splitters have one incoming stream, which is split into two streams

$$\begin{aligned} N_{i,out1} &= \alpha N_{i,feed} \\ N_{i,out2} &= (1 - \alpha) N_{i,feed}. \end{aligned}$$

In the splitter before the recycle loop the split ratio α is a parameter, while in the other splitters it is a variable which is calculated by the overall heat balance.

No heat losses are considered for the splitter, and no heat exchange is allowed, so that the energy balance is simply

$$\sum_i N_{i,feed} H_{i,feed} = \sum_i N_{i,out1} H_{i,out1} + \sum_i N_{i,out2} H_{i,out2}.$$

A.5.2 Reactor and Burners

Reactors and burners have one output stream, but multiple inlet streams. In the reactor the outlet stream is calculated according to the stoichiometric matrix approach

$$N_{i,out} = \sum_j N_{i,feedj} + \xi_r \nu_{i,r}.$$

Only one idealized reaction is assumed to take place in each reactor or burner to a given conversion ζ_r , so that the extent of reaction ξ_r can be calculated as

$$\xi_r = \zeta_r \min_{i \in \text{reactants}} \left(\frac{\sum_j N_{i,feedj}}{-\nu_{i,r}} \right).$$

An excess of oxygen ($\Phi \geq 1$) is always assumed for the reactors and burners

$$\sum_j N_{O_2,feedj} = -\Phi \sum_r \xi_r \nu_{O_2,r}.$$

Φ is an operating parameter which needs to be specified prior to the simulation.

In the reactors and the burners heat losses are considered and heat exchange is allowed, so that the energy balance reads:

$$\sum_i N_{i,out} H_{i,out}(T_{out}) = \sum_j \sum_i N_{i,feedj} H_{i,feedj} + Q_p - Q_{loss}.$$

Note that the outlet temperature T_{out} is allowed to be different than the operating temperature T_{op} in order to allow heat recovery.

A.5.3 Mixer

Mixers have one output stream, but multiple inlet streams. In the mixer the outlet stream is calculated according to:

$$N_{i,out} = \sum_j N_{i,feedj}.$$

No heat losses are considered for the mixer, and no heat exchange is allowed, so that the energy balance is simply

$$\sum_i N_{i,out} H_{i,out} = \sum_j \sum_i N_{i,feedj} H_{i,feedj}.$$

A.5.4 Membrane

The membrane separates part of the hydrogen in the inlet stream, into a pure hydrogen stream. The rest of the hydrogen is lost into the waste stream. The part of the hydrogen that is recovered is specified by the membrane efficiency η_M

$$\begin{aligned} N_{i,feed} &= N_{i,out1} + N_{i,out2} \\ N_{H_2,out1} &= \eta_M N_{H_2,feed} \\ N_{i \neq H_2,out1} &= 0. \end{aligned}$$

This model is based on the assumption that hydrogen is the only element that can appreciably permeate through the membrane and there is a need for a partial pressure gradient which results in the hydrogen loss to the waste stream.

No heat losses are considered for the membrane, and no heat exchange is allowed, so that the energy balance is simply

$$\sum_i N_{i,feed} H_{i,feed} = \sum_i N_{i,out1} H_{i,out1} + \sum_i N_{i,out2} H_{i,out2}.$$

The pressure loss associated with membranes is not taken into account.

A.5.5 Compressor

While large-scale compressors are typically modeled as adiabatic, at the micro scale heat transfer is relatively high and therefore it is likely that compressors or pumps will be operated isothermally, most probably at near-ambient temperature. Motivated by the isothermal compression of ideal gases, the required power for the compression is

modeled as

$$PW = \frac{1}{\eta_P} R T \ln \left(\frac{P_{out}}{P_{in}} \right) \sum_i N_i = K_P T \sum_i N_i,$$

where $K_P \equiv \frac{1}{\eta_P} R \ln \left(\frac{P_{out}}{P_{in}} \right)$ is a parameter, that needs to be specified prior to the simulation. K_P can be estimated by knowledge of the pressure ratio that is to be achieved, and the compressor efficiency η_P . For our models the consumed power is assumed to be lost to the environment as heat, so that no energy balance is needed.

A.5.6 Fuel Cells

For all fuel cells we assume that the produced fuel cell power, PW , is proportional to the Gibbs free energy of reaction for the electrochemical reactions:

$$PW = \eta_{FC} \sum_r \xi_r \Delta_r G,$$

where the Gibbs free energy of reaction is given by

$$\Delta_r G = \sum_i \nu_{i,r} \left(G_i^c(T) + RT \ln \left(\frac{P}{P_{ref}} \right) \right).$$

We neglect the mixing term, because this would require knowledge of the composition along the fuel cell, which depends on the geometry. With the exception of the SOFC, only the final hydrogen reaction is considered to contribute to the produced power, by using the notion of equivalent hydrogen [9]. A heat balance over the fuel cell yields

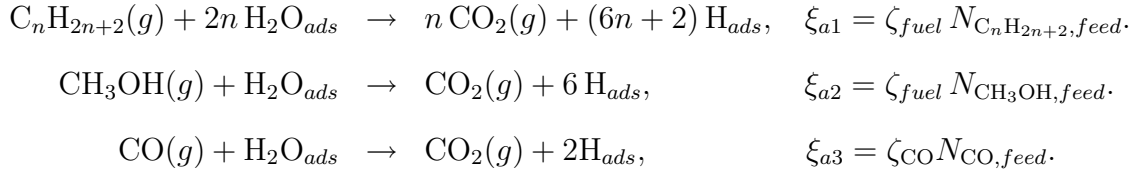
$$\begin{aligned} & \sum_i N_{i,outc} H_{i,outc}(T_{outc}) + \sum_i N_{i,outa} H_{i,outa}(T_{outa}) \\ &= \sum_i N_{i,inc} H_{i,inc}(T_{inc}) + \sum_i N_{i,ina} H_{i,ina}(T_{ina}) + Q_p - Q_{loss} + PW, \end{aligned}$$

where Q_{loss} is calculated as described in [201]. In all fuel cells we allow for specification of minimum water feed in the anode; this is calculated based on the water required for complete reforming reactions and water permeation, multiplied by the water factor Ψ , a given excess parameter; this factor is defined for each fuel cell in the following

subsections.

Proton Ceramic Fuel Cell (PCFC)

In this type of fuel cell, oxygen vacancies in a ceramic membrane allow for protonic conduction from the anode to the cathode and water conductivity [75] allows for water management via diffusion from the cathode to the anode. We assume that the mechanism described for methane reforming in [75] can be extended to arbitrary hydrocarbons and to methanol, but we do not consider ammonia conversion in this fuel cell model. We allow the direct feed of fuels to the PCFC, as well as feed of reactor effluents. We assume that hydrocarbons, methanol, and carbon monoxide undergo each one net reaction resulting in the production of carbon dioxide and hydrogen based on given conversions:

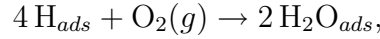


Hydrogen from the feed has to adsorb to and diffuse through the membrane. We assume that only a fraction, η_{ads} , of the inlet hydrogen will adsorb to the membrane, and a fraction, η_{diff} , thereof will diffuse to the cathode. Any hydrogen produced from reforming is produced as adsorbed hydrogen, so a fraction, η_{diff} , of it will diffuse. For both cases, we assume that the fraction of adsorbed hydrogen that does not diffuse fully de-adsorbs from the membrane, since we do not allow accumulation. Based on these assumptions, the amount of diffused hydrogen that is available on the cathode side is given by

$$N_{H, diff} = \eta_{diff} \sum_{a,r} \xi_{a,r} \nu_{H_{ads}, a, r} + 2 \eta_{ads} \eta_{diff} N_{H_2, feed},$$

where the summation is over all reforming reactions at the anode a . We assume that

all diffused hydrogen reacts with oxygen at the cathode side via



with an extent of reaction given by the diffusion rate of hydrogen

$$\xi_c = \frac{1}{4} N_{\text{H},diff}.$$

The modeling assumptions have the consequence that the rate of water production at the cathode is larger than the rate of water consumption at the anode. We further assume that the anode inlet stream, does not contain a sufficient amount of water to reverse the presumed concentration gradient, and that all water for the reforming reactions will diffuse from the cathode to the anode, which defines

$$N_{\text{H}_2\text{O},diff} = - \sum_{a,r} \xi_{a,r} \nu_{\text{H}_2\text{O},a,r},$$

where the summation is again over all anode reforming reactions. Therefore, a mass balance on the cathode side yields

$$\begin{aligned} N_{\text{H}_2,outc} &= N_{\text{H}_2,inc} = 0 \\ N_{\text{H}_2\text{O},outc} &= N_{\text{H}_2\text{O},inc} + \xi_c \nu_{\text{H}_2\text{O}} - N_{\text{H}_2\text{O},diff} \\ N_{i,outc} &= N_{i,inc} + \xi_c \nu_i, \quad i \neq \text{H}_2, \text{H}_2\text{O}. \end{aligned}$$

Similarly for the anode:

$$\begin{aligned} N_{\text{H}_2,outa} &= N_{\text{H}_2,ina} + \sum_{a,r} \xi_{a,r} \nu_{\text{H}_2,r} - N_{\text{H}_2,diff} \\ N_{\text{H}_2\text{O},outa} &= N_{\text{H}_2\text{O},ina} \\ N_{i,outa} &= N_{i,ina} + \sum_{a,r} \xi_{a,r} \nu_{i,a,r}, \quad i \neq \text{H}_2, \text{H}_2\text{O}. \end{aligned}$$

The oxygen feed to the cathode is specified relative to the stoichiometric requirement

to oxidize the diffused hydrogen completely:

$$N_{\text{O}_2,feedc} = \Phi \xi_c.$$

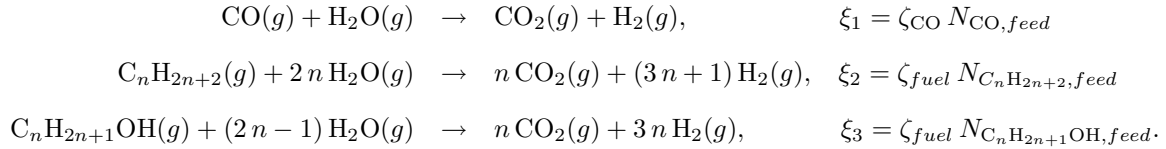
We allow for a specification of water feed to the anode by requiring

$$N_{\text{H}_2\text{O},feeda} \geq \Psi \sum_{a,r} \frac{-\nu_{\text{H}_2\text{O},r} \xi_{a,r}}{\zeta_i} - N_{\text{H}_2\text{O},diff},$$

where the reforming reactions are assumed to be complete. Strict inequality is observed when recycling streams or the reactor effluent stream provides sufficient water.

Solid Oxide Fuel Cell (SOFC)

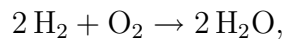
We consider simultaneously the possibilities of internal or external reforming, with the option of bypassing the reactor and feeding fuel directly to the fuel cell. A common assumption [9], is to assume that carbon monoxide, hydrocarbons and alcohols are consumed in reforming and water gas-shift reactions rather than in direct oxidation.



In order to model a fuel cell that is not capable of internal reforming, one must set ζ_{fuel} and/or ζ_{CO} equal to zero. We use the notion of “equivalent hydrogen”, which accounts for direct hydrogen feed as well as hydrogen produced from internal reforming of CO, hydrocarbons, or alcohols [9]:

$$N_{\text{H}_2,equiv} = \zeta_{\text{H}_2} N_{\text{H}_2,feed} + \sum_{r=1}^3 \nu_{\text{H}_2,r} \xi_r.$$

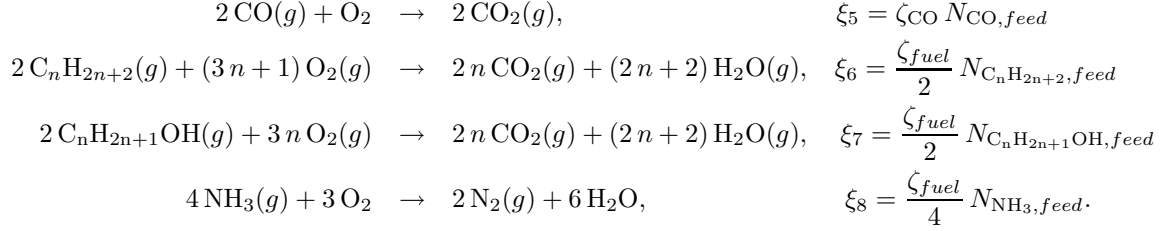
The hydrogen reacts with oxygen according to



with an extent of reaction defined by the equivalent hydrogen

$$\xi_4 = \frac{1}{2} N_{\text{H}_2, \text{equiv}}.$$

The other extreme possibility is to assume that all components are directly oxidized in an electrochemical reaction



We allow for either extreme; in the case of direct oxidation all electrochemical reactions are assumed to contribute to power generation

$$PW = \eta_{FC} \sum_{r=4}^8 \xi_r \Delta_r G,$$

while in the case of intermediate reforming reactions only the hydrogen oxidation is assumed to produce power

$$PW = \eta_{FC} \xi_4 \Delta_4 G.$$

The amount of oxygen fed to the cathode is calculated relative to the stoichiometric requirement, assuming that all diffused oxygen will react in the anode:

$$N_{\text{O}_2, \text{feedc}} = \Phi N_{\text{O}_2, \text{diff}} = \Phi \sum_{r=4}^8 \xi_r.$$

In order to prevent coking, an excess of water may be necessary. The water feed to the anode is set according to:

$$N_{\text{H}_2\text{O}, \text{feeda}} \geq \Psi \sum_{r=1}^3 \frac{-\nu_{\text{H}_2\text{O}, r} \xi_{a, r}}{\zeta_r} - \xi_4 \nu_{\text{H}_2\text{O}, 4},$$

where the reforming reactions are assumed to be complete, and the produced water is assumed to be available for reforming reactions. Strict inequality is observed when recycling streams or the reactor effluent stream provides sufficient water. A mass balance on the cathode yields

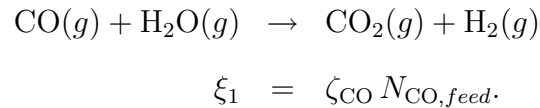
$$\begin{aligned} N_{i,outc} &= N_{i,inc}, \quad i \neq O_2, \\ N_{O_2,outc} &= N_{O_2,inc} - N_{O_2,diff}, \end{aligned}$$

and for the anode

$$\begin{aligned} N_{i,outa} &= N_{i,ina} + \sum_{r=1}^4 \xi_r \nu_{i,r}, \quad i \neq O_2, \\ N_{O_2,outa} &= N_{O_2,ina} + \sum_{r=1}^4 \xi_r \nu_{i,r} + N_{O_2,diff}. \end{aligned}$$

Single Chamber Fuel Cell

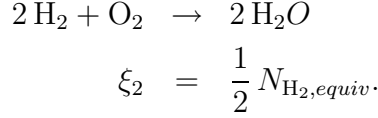
We consider the option of a single chamber fuel cell which uses external reforming, resulting in an inlet feed rich in H_2 and CO . The unreacted fuel is assumed to be inert. Inlet CO is reformed via the water gas-shift reaction with extent of reaction ξ_1 according to



We again use the notion of “equivalent hydrogen” [9], here calculated as

$$N_{H_2,equiv} = \zeta_{H_2} N_{H_2,feed} + \xi_1.$$

All equivalent hydrogen reacts with oxygen with an extent of reaction ξ_2 according to



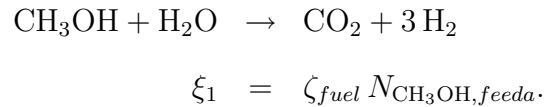
The water is set relative to complete reforming reaction of all components

$$N_{\text{H}_2\text{O}, \text{feeda}} \geq \Psi (N_{\text{CO}, \text{in}} + N_{\text{CH}_3\text{OH}} + 6 N_{\text{C}_3\text{H}_8, \text{in}} + 8 N_{\text{C}_4\text{H}_{10}, \text{in}}) - \xi_2 \nu_{\text{H}_2\text{O}, 2},$$

and the produced water is assumed to be available for reforming reactions.

Direct Methanol Fuel Cell (DMFC)

In this type of polymer electrolyte based fuel cell, an inlet stream of methanol and water is fed to the anode, where methanol is reformed at a relatively low temperature, around 350K. No external reforming is required and we therefore consider only direct feed of water and methanol. Operating requirements include relatively dilute solutions of methanol in water, and major technical challenges include methanol crossover and water management [136]. At the anode we assume methanol reforming with an extent of reaction ξ_1 based on a given fuel conversion, ζ_{fuel} :



A fraction $k_{\text{H}_2}^{\text{trns}}$ of the hydrogen produced by this reaction is assumed to diffuse to the cathode side

$$N_{\text{H}_2}^{\text{trns}} = k_{\text{H}_2}^{\text{trns}} \xi_1 \nu_{\text{H}_2, 1} = 3 k_{\text{H}_2}^{\text{trns}} \xi_1.$$

The hydrogen diffusion occurs in protonic form, and electro-osmotic drag results in water transport from the anode to the cathode [235]. On the other hand, concentration gradients may result in water diffusion from the cathode to the anode, while

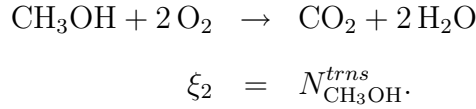
electro-osmotic drag and diffusion lead to methanol transport from the anode to cathode [235, 134, 188]. For the water transport we use an effective drag coefficient that includes the electro-osmotic drag and diffusion and the overall transfer rate of water is defined relative to the proton flux:

$$N_{\text{H}_2\text{O}}^{trns} = 2 k_{\text{H}_2\text{O}}^{trns} N_{\text{H}_2}^{diff}.$$

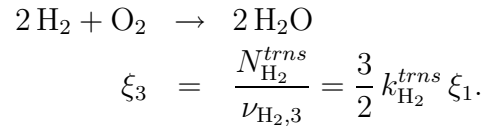
The coefficient $k_{\text{H}_2\text{O}}^{trns}$, can be negative or positive. We assume that a fraction, $k_{\text{CH}_3\text{OH}}^{trns}$ of the unreacted methanol will diffuse from the anode to the cathode:

$$N_{\text{CH}_3\text{OH}}^{trns} = k_{\text{CH}_3\text{OH}}^{trns} (1 - \zeta_{fuel}) N_{\text{CH}_3\text{OH}, feeda}.$$

We assume that methanol present at the cathode will be completely [235] oxidized to CO_2 with an extent of reaction ξ_2 ,



Similarly we assume that all diffused hydrogen reacts with oxygen at the cathode with an extent of reaction ξ_3



Mass balance on the cathode side yields

$$\begin{aligned} N_{i,outc} &= N_{i,inc} + \sum_{r=2,3} \xi_r \nu_{i,r} + N_i^{trns}, \quad i = \text{H}_2, \text{H}_2\text{O}, \text{CH}_3\text{OH} \\ N_{i,outc} &= N_{i,inc} + \sum_{r=2,3} \xi_r \nu_{i,r}, \quad i \neq \text{H}_2, \text{H}_2\text{O}, \text{CH}_3\text{OH} \end{aligned}$$

where the summation is over the two cathode reactions. Similarly for the anode:

$$\begin{aligned} N_{i,out} &= N_{i,in} + \xi_1 \nu_{i,1} - N_i^{trns}, & i = \text{H}_2, \text{H}_2\text{O}, \text{CH}_3\text{OH} \\ N_{i,out} &= N_{i,in} + \xi_1 \nu_{i,1}, & i \neq \text{H}_2, \text{H}_2\text{O}, \text{CH}_3\text{OH}. \end{aligned}$$

The oxygen feed to the cathode is specified relative to the net stoichiometric requirement of the two oxidation reactions:

$$N_{\text{O}_2,feed} = -\Phi (\xi_2 \nu_{\text{O}_2,2} + \xi_3 \nu_{\text{O}_2,3}).$$

The water feed to the anode is set as a multiple of the maximal water needed

$$\begin{aligned} N_{\text{H}_2\text{O},feed} &\geq \Psi \frac{-\nu_{\text{H}_2\text{O},1} \xi_1}{\zeta_i} + N_{\text{H}_2\text{O}}^{trns} & k_{\text{H}_2\text{O}}^{trns} < 0 \\ N_{\text{H}_2\text{O},feed} &\geq \Psi \left(\frac{-\nu_{\text{H}_2\text{O},1} \xi_1}{\zeta_i} + N_{\text{H}_2\text{O}}^{trns} \right) & k_{\text{H}_2\text{O}}^{trns} > 0, \end{aligned}$$

where the reforming reactions are assumed complete.

Water Management

Currently direct methanol fuel cells operate with relatively dilute solutions of methanol in water, but storing a diluted methanol solution greatly reduces the energy density. Therefore, water management becomes almost mandatory; at least in theory it is possible to use the produced water to run the methanol reforming reaction. This can be done either by ensuring that diffusion of water from the anode to the cathode matches the water requirements for the methanol reforming and the electro-osmotic drag of water, or by recycling water from the cathode and/or anode effluents to the anode. The latter possibility is included as an option in the superstructure, and our model allows the first option by specifying a negative drag coefficient $k_{\text{H}_2\text{O}}^{trns}$. The overall transport of water from the cathode to the anode is limited by the production rate of water in the cathode; with simple algebraic manipulations the requirement

$$k_{\text{H}_2\text{O}}^{trns} \geq -\frac{2 k_{\text{CH}_3\text{OH}}^{trns} (1 - \zeta) + 3 k_{\text{H}_2}^{trns} \zeta}{6 k_{\text{H}_2}^{trns} \zeta}$$

can be calculated. Water management is essentially achieved when the effective transport of water from the cathode to the anode covers the water needs for the methanol reforming reaction; in the absence of external recycling this is achieved for

$$k_{\text{H}_2\text{O}}^{trns} \leq -\frac{1}{6k_{\text{H}_2}^{trns}}.$$

There are regions in the parameter space spanned by $k_{\text{CH}_3\text{OH}}^{trns}$, $k_{\text{H}_2}^{trns}$ and ζ where water management is only possible through external recycling of the anode liquids, but these are for low values of the transport coefficient $k_{\text{H}_2}^{trns} \leq 1/3$, in which case the DMFC would not operate efficiently anyway.

Polymer Electrolyte Fuel Cell (PEM)

Only one reaction takes place, and the extent of reaction ξ is calculated, based on a conversion ζ , that is specified prior to the simulation

$$2 \text{H}_2 + \text{O}_2 \rightarrow 2 \text{H}_2\text{O} \quad \xi = \zeta \frac{N_{\text{H}_2, \text{feeda}}}{2},$$

assuming that the reaction rate is determined uniquely by the diffusion of the hydrogen ions to the anode

$$N_{\text{H}_2}^{trns} = \zeta N_{\text{H}_2, \text{feeda}}.$$

We also account for a net water transport

$$N_{\text{H}_2\text{O}}^{trns} = 2 k_{\text{H}_2\text{O}}^{trns} N_{\text{H}_2}^{diff},$$

from the anode to the cathode. Diffusion of all other components through the membrane is neglected

$$N_i^{trns} = 0 \quad i \neq \text{H}_2, \text{H}_2\text{O}.$$

The mass balance for the cathode side reads:

$$N_{i, \text{outc}} = N_{i, \text{feedc}} + N_i^{trns} + \xi \nu_i$$

and for the anode

$$N_{i,outc} = N_{i,feeda} - N_i^{trns}.$$

As a consequence $N_{H_2,outc} = 0$. We require a water feed to the anode relative to the net transport of water from the anode to the cathode

$$N_{H_2O,feeda} \geq \Psi N_{H_2O}^{trns}.$$

The feed of oxygen into the cathode is set according to

$$N_{O_2,feedc} = \Phi \xi.$$

A.5.7 Flash for Separation of Fuel Cell Effluents

We assume that the liquid and gaseous stream are in equilibrium and that the liquid stream is split into a purge stream and a recycle stream.

$$\begin{aligned} N_{i,rec} &= \alpha N_i^l \\ N_{i,purge} &= N_i^l - N_{i,rec}, \end{aligned}$$

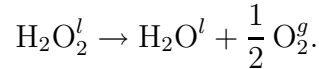
where α is a given splitratio. For simplicity, ideal liquid and gas phase are assumed, with supercritical components present only in the gas-phase $N_i^l = 0$, neglecting solution in the liquid phase according to Henry's law and using the Wagner equation [234] for the subcritical components. Depending on the implementation of the recycle stream, a pressure increase mechanism may be necessary, e.g., a pump. We impose an energetic penalty in terms of a compression power.

$$PW = K_P \sum_i V_{m,i} N_i,$$

where K_P is a parameter, that needs to be specified prior to the simulation. K_P can be estimated by knowledge of the pressure ratio that is to be achieved, and the compression efficiency.

A.5.8 Oxygen Generators

In volume-critical applications where atmospheric oxygen is not available, chemical oxygen generators can offer a significant increase in volumetric energy density as compared to compressed air or compressed oxygen. This increase in volumetric energy density is coupled, however, to a decrease in gravimetric energy density due to the added mass of the generator. We consider hydrogen peroxide as an example of such an oxygen generator. Due to its extensive current use, many of the safety concerns and transportation issues are already well understood [206, 179]. No special regulations apply to the transport of dilute solutions (≤ 8 weight%) of hydrogen peroxide, while concentrated solutions (≥ 40 weight%) are not admitted for air transport [127]. Hydrogen peroxide decomposes when heated, releasing oxygen according to



We consider a single cartridge that feeds all the units. For a given mission duration $\tau_{mission}$, the total moles and mass of hydrogen peroxide needed are given by

$$\begin{aligned} n_{\text{H}_2\text{O}_2} &= 2 \tau_{mission} N_{\text{O}_2, total} \\ m_{\text{H}_2\text{O}_2} &= n_{\text{H}_2\text{O}_2} MW_{\text{H}_2\text{O}_2}, \end{aligned}$$

where $N_{\text{O}_2, total}$ is the molar flow of oxygen necessary for all units and the factor 2 comes from stoichiometry. For a given weight fraction of hydrogen peroxide ($w_{\text{H}_2\text{O}_2}$) in solution with water, the mass of solvent water is calculated as

$$m_{\text{H}_2\text{O}, solvent} = \frac{m_{\text{H}_2\text{O}_2} (1 - w_{\text{H}_2\text{O}_2})}{w_{\text{H}_2\text{O}_2}}.$$

The total volume of solution is calculated as

$$V_{solution} = \frac{m_{\text{H}_2\text{O}_2}}{w_{\text{H}_2\text{O}_2} \rho_{solution}},$$

where the solution density, $\rho_{solution}$, is assumed to depend linearly on the peroxide weight fraction [98]

$$\rho_{solution} = \rho_{\text{H}_2\text{O}} + w_{\text{H}_2\text{O}_2} (\rho_{\text{H}_2\text{O}_2} - \rho_{\text{H}_2\text{O}}).$$

We do not account for the kinetics of hydrogen peroxide decomposition by assuming that they are sufficiently fast, nor do we account for the energy requirements by assuming that any required heat can be provided by ambient heat or by sweep gases from the fuel cell or reactor effluent streams.

A.5.9 Water Breathing Systems

The volume and mass of the water extraction is estimated by a given residence time in the extraction device τ_{extr}

$$\begin{aligned} V_{extr} &= \tau_{extr} V_{\text{H}_2\text{O}} \\ M_{extr} &= \tau_{extr} MW_{\text{H}_2\text{O}} \end{aligned}$$

The energetic penalty is assumed proportional to the vaporization enthalpy

$$PW_{extr} = \eta_{extr} N_{\text{H}_2\text{O}} H_{\text{H}_2\text{O}}^{vap}$$

An alternative would be to calculate it based on osmotic pressures.

A.5.10 Hydrogen Generators

Using hydrogen as a primary fuel is an appealing alternative, due to its high gravimetric energy density and the efficient operation of fuel cells with H_2 . However, since hydrogen is supercritical at ambient temperature, it cannot be liquefied, and gaseous hydrogen storage requires a large volume, resulting in a low volumetric energy density. These limitations, along with difficulty in storage and safety issues of pure hydrogen, have led to the consideration of safer, low-volume and low-pressure methods of hydro-

gen storage. As with oxygen generators, these hydrogen generators offer significant increases in volumetric energy densities but impose a penalty in terms of gravimetric energy densities. We consider the option of generic hydrogen storage in a single model, approximating many technological alternatives based on either physisorption, e.g., metal hydrides or carbon nanotubes [288], or even chemical hydrides that do not require the addition of water for decomposition [287, 256]. As parameters we use the hydrogen density $\rho_{\text{H}_2} = \frac{g\text{H}}{\text{cm}^3\text{storage}}$ and hydrogen weight fraction w_{H_2} . The range for this parameters is known for common metal hydrides [288, 149]:

$$\begin{aligned}\rho_{\text{H}_2} &\approx 0.02 - 0.2\text{g}/\text{cm}^3 \\ w_{\text{H}_2} &\approx 0.01 - 0.08.\end{aligned}$$

The mass of hydrogen m_{H_2} required over the mission duration is calculated as

$$m_{\text{H}_2} = \tau_{\text{mission}} N_{\text{H}_2} MW_{\text{H}_2},$$

where N_{H_2} is the flowrate of hydrogen required for the given power output. The mass of devoid hydrogen generator, i.e., the residual hydrogen storage material, that is needed for the mission duration is determined by the hydrogen weight percent:

$$m_{\text{H}_2\text{gen}} = \frac{m_{\text{H}_2} (1 - w_{\text{H}_2})}{w_{\text{H}_2}}.$$

This mass is constant and must be carried for the entire mission duration. The mass of stored hydrogen will decrease from m_{H_2} to zero as hydrogen is released over the course of the mission. The volume required to store the hydrogen generator depends on the density of hydrogen, ρ_{H_2} and initial stored mass of hydrogen:

$$V_{\text{H}_2\text{gen}} = \frac{m_{\text{H}_2}}{\rho_{\text{H}_2}}.$$

Similarly to the oxygen generators we do not account for kinetics or energy requirements.

A.5.11 Implementation and Convergence Scheme

The alternatives considered are represented as a steady-state simulation model using our in-house software packages ABACUSS II [273, 272] and DAEPACK [269, 271]. The mass and energy balances are formulated within the process simulator ABACUSS II, while the physical property calculations are performed in Fortran external procedures. In order to facilitate the use of the model we have embedded the simulation in a web-interface [199]. The requirements for the energy balance of high temperature systems result in disjoint conditions

```
IF  $Q < 0$  THEN  
    burn no fuel  
ELSE  
    burn enough fuel to set  $Q = 0$   
ENDIF.
```

Similarly the water requirement ($N_{\text{H}_2\text{O},in} \geq N_{\text{H}_2\text{O},min}$) results in a condition

$$N_{\text{H}_2\text{O}} = \max(0, N_{\text{H}_2\text{O},min}).$$

Both formulations lead to nonsmooth residuals and currently the solvers have no convergence guarantees for nonsmooth functions. One way for dealing with a nonsmooth function is to use a smooth approximation [47], but our experience shows that the water requirement does not pose significant convergence problems while the energy requirement would be too difficult to formulate as a smooth function. In order to assure robust convergence we remove the disjoint constraints for the energy balances and use the amount of fuels or products to be burned in each stack as a guess variable. The case of one guess variable is easy to solve since the heat load is monotonically decreasing with the flowrate and a secant method can be used. For the case of two variables we use a Broyden-like algorithm [47]; when the Broyden method calculates guess values outside the variable bounds, these guesses are replaced by the bounds. We initialize the approximate Jacobian by the negative identity matrix. This

choice is motivated by the fact that the diagonal elements dominate, and the heat loads decrease with increasing guess variable ($\frac{df}{dx} < 0$). The convergence is sufficiently fast and robust, so that we did not implement any line search method (e.g., Armijo rule).

A.6 Reduced Model for Heat Exchanger

A.6.1 Tube Model

With no reactions in the tubes, a mass balance on species i and an overall energy balance yield respectively

$$\begin{aligned} N_{i,out} &= N_{i,in}, \\ H_{out} &= H_{in} + Q_{loss} + Q_{exch}. \end{aligned}$$

Heat losses from radiation are estimated using the average temperature ($T_{avg} = \frac{T_{out} + T_{in}}{2}$), total surface area (A_{tube}), and emissivity (ϵ_{tube}):

$$Q_{rad} = -A_{tube} \epsilon_{tube} \sigma_{SB} T_{avg}^4.$$

The total heat losses, including conductive loss to the ambient depend on the total area and the overall heat loss coefficient, $U_{loss,tube}$:

$$Q_{loss} = Q_{rad} - A_{tube} U_{loss,tube} (T_{avg} - T_{amb}).$$

The heat exchange to the tubes results from radiation from the ambient, other tubes, the slabs, and the reactor. For simplicity we neglect the conductive heat transfer from slab to the tube. Therefore, the total heat exchange into tube i is given by:

$$Q_{exch,i} = - \sum_j F_{j,i} Q_{rad,j},$$

where $F_{j,i}$ represents the view factor from unit j to tube i . By defining the inlet temperature, heat losses, and heat exchange, the outlet enthalpy can be used to calculate the outlet temperature of the tube.

A.6.2 Slab Model

With no reactions and no mixing between individual streams flowing through the slabs, a mass balance on species i in stream j yields:

$$N_{i,jout} = N_{i,jin}.$$

An energy balance over the entire slab states:

$$\sum_j H_{jout} = \sum_j H_{jin} + Q_{loss} + Q_{exch}.$$

We here impose the condition that all stream outlet temperatures ($T_{out,j}$) are equal to the operating temperature of the slab.

$$T_{out,j} = T_{op}.$$

The heat losses due to radiation and total heat losses are given by:

$$Q_{rad} = -A_{slab} \epsilon_{slab} \sigma_{SB} T_{op}^4 \quad (\text{A.1})$$

$$Q_{tot} = Q_{rad} - A_{slab} U_{loss,slab} (T_{op} - T_{amb}). \quad (\text{A.2})$$

As heat exchange mechanisms we consider radiation of surrounding units, conduction along the tubes between consecutive slabs, as well as conduction along connecting wires between alternating slabs. Therefore, the total heat exchange into slab i is given by:

$$Q_{exch,i} = - \sum_j F_{j,i} Q_{rad,j} + Q_{cond-tube,i} + Q_{cond-wire,i},$$

where $Q_{cond-slab,i}$ and $Q_{cond-wire,i}$ represent the total heat transfer into the slabs via conduction.

A.6.3 Defining Unit Interactions

The streams flowing through a given slab are illustrated in Figure A-2, assuming a total of eight streams. We give inlet streams odd numbers and outlet streams even numbers.

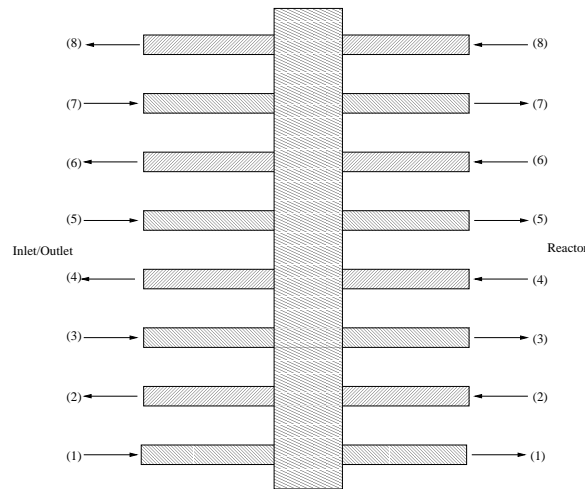


Figure A-2: Stream Numbering

We define a one-dimensional array of slab models with the index increasing towards the reactor as shown in Figure A-2. For a given stream, the number of tubes is equal to the number of slabs plus one, accounting for the inlet and outlet tube. We define a two-dimensional array of tube models with the first index increasing towards the reactor and the second index representing the stream which it is part of. For a cross section representing a given stream, the numbering is indicated in Figure A-3.

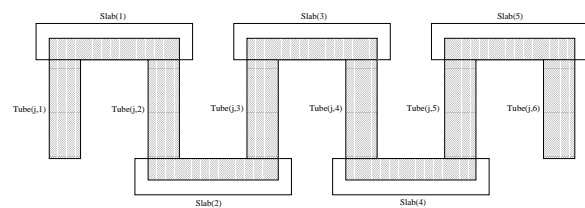


Figure A-3: Tube Numbering for Stream j

This numbering system is used to define the flow connections between all tubes and slabs, except the inlet flow and the reactor outlet flow. We also define the heat conduction along the SiN tubes and along the connecting wires. The conductive heat flow along the tubes from slab i to slab $i + 1$ depends on the thermal conductivity, cross sectional area, and length of the tube:

$$Q_{cond-tube,i \rightarrow i+1} = \frac{k_{tube}^{th} A_{CS,tube}}{L_{tube}} (T_{slab(i)} - T_{slab(i+1)}).$$

Similarly, the conductive heat flow along the connecting wires from slab i to slab $i + 2$ depends on the thermal conductivity, cross sectional area, and length of wire between the adjacent slabs.

$$Q_{cond-wire,i \rightarrow i+2} = \frac{k_{wire}^{th} A_{CS,wire}}{L_{wire}} (T_{slab(i)} - T_{slab(i+2)}).$$

In order to account for the heat exchange to the slabs and tubes, the radiation of the reactor is also calculated

$$Q_{rad,reactor} = A_{reactor} \epsilon_{reactor} \sigma_{SB} T_{reactor}^4.$$

Appendix B

Parametric Optimization Test Set

Example B.1. The following parametric LP is based on Murty [207]

$$\begin{aligned} \min_{\mathbf{x}} \quad & 16x_4 + 4x_5 + x_6 \\ \text{s.t. } x_1 \quad & +x_4 = 8 - 20(1 + 2p) \\ x_2 \quad & +x_4 + x_5 = 4 - 20(1 + 2p) \\ x_3 + 2x_4 + 2x_5 + x_6 = & 2 - 20(1 + 2p) \\ \mathbf{x} \in \mathbb{R}, \quad & \mathbf{x} \geq \mathbf{0}. \end{aligned} \tag{B.1}$$

There are 8 optimal solutions with the breakpoints $p = 0.55$, $p = 0.6$, $p = 0.65$, $p = 0.7$, $p = 0.75$, $p = 0.8$, and $p = 0.85$.

Example B.2. The right-hand side parametric MILP

$$\begin{aligned} \min_{x,y} \quad & y + x \\ \text{s.t. } \quad & x = -1 + 2p \\ x \in \mathbb{R}, \quad & x \geq 0 \\ y \in \quad & \{0, 1\} \end{aligned} \tag{B.2}$$

is infeasible for $p \in [0, 0.5)$, because $x \geq 0$ is violated. For $p \in [0.5, 1.0]$ the optimal solution is $y = 0$, $x = -1 + 2p$ with an optimal objective function of $-1 + 2p$.

Example B.3. The parametric MILP

$$\begin{aligned}
 & \min_{x,y} y + x \\
 & \text{s.t. } (-1 + 2p)x = 1 \\
 & \quad x \in [0, 10] \\
 & \quad y \in \{0, 1\}
 \end{aligned} \tag{B.3}$$

is feasible for $p \in [0.55, 1.0]$ with an optimal solution $y = 0$ and $x = 1/(-1 + 2p)$ and an optimal objective function $1/(-1 + 2p)$. For $p \in [0, 0.5)$ the solution of the equality constraint gives a negative x and (B.3) is infeasible. For $p = 0.5$ the matrix is singular and no solution exists for the equality constraint. For $p \in (0.5, 0.55)$ the solution to the equality constraint gives $x > 10$ which is infeasible.

Example B.4. The parametric LP

$$\begin{aligned}
 & \min_{\mathbf{x}} x_2 \\
 & \text{s.t. } x_2 - 100px_1 = 0 \\
 & \quad x_1 = p \\
 & \quad \mathbf{x} \in [0, 10]^2.
 \end{aligned} \tag{B.4}$$

is feasible for $p \in [0, 0.1\sqrt{10}] \approx [0, 0.3162]$ with an optimal solution $x_1 = p$, $x_2 = 100p^2$ and an optimal objective function $100p^2$.

Example B.5. The right-hand side parametric LP

$$\begin{aligned}
 & \min_{\mathbf{x}} x_3 \\
 & \text{s.t. } x_1 + x_2 = 0 \\
 & \quad x_3 = -0.5 + p \\
 & \quad \mathbf{x} \in [0, 10]^3
 \end{aligned} \tag{B.5}$$

if feasible for $p \in [0.5, 1]$ with an optimal solution of $x_1 = x_2 = 0$, $x_3 = p - 0.5$ with

an optimal objective function of $p - 0.5$.

Example B.6. The parametric MILP

$$\begin{aligned}
 \min_{x, \mathbf{y}} \quad & -x + y_1 \\
 \text{s.t.} \quad & y_2 + (0.99p + 0.01)x = 1 \\
 & y_1 + y_2 = 1 \\
 & x \in [0, 20] \\
 & \mathbf{y} \in \{0, 1\}^2
 \end{aligned} \tag{B.6}$$

has two candidate integer realizations. The first integer realization is $y_1 = 1, y_2 = 0$ gives $x = 1/(0.99p + 0.01)$ and an objective value of $1 - 1/(0.99p + 0.01)$. This solution is feasible for $p \in [0.0404, 1]$ and optimal for $p \in [0.0404, 1]$. The second integer realization $y_1 = 0, y_2 = 1$ gives $x = 0$ and an objective value of 0. This solution is feasible for $p \in [0, 1]$ and optimal for $p \in [0, 0.0404] \cup \{1\}$.

Example B.7. The parametric LP

$$\begin{aligned}
 \min_{\mathbf{x}} \quad & x_2 \\
 \text{s.t.} \quad & (1 - p)x_2 = 0.1 \\
 & x_1 = 0 \\
 & \mathbf{x} \in [0, 1]^2
 \end{aligned} \tag{B.7}$$

is feasible for $p \in [0, 0.9]$ with an optimal solution $x_1 = 0, x_2 = 0.1/(1 - p)$ and an objective function $0.1/(1 - p)$.

Example B.8. The cost vector parametric LP

$$\begin{aligned}
 \min_x \quad & (-1 + 2p)x \\
 \text{s.t.} \quad & x \leq 1 \\
 & x \in [0, 10]
 \end{aligned} \tag{B.8}$$

is feasible for $p \in [0, 1]$. For $p \in [0, 0.5)$ the unique optimal solution is $x = 1$ (active inequality constraint) and an objective function of $-1 + 2p$. For $p = 0.5$ any $x \in [0, 1]$ is optimal with an objective value of 0. For $p \in (0.5, 1]$, the unique optimal solution is $x = 0$ (inactive inequality constraint) with an objective function of 0.

Example B.9. The parametric MILP

$$\begin{aligned}
& \min_{\mathbf{y}, \mathbf{x}} +10 y_1 + 12 y_2 + x_1 + x_2 + x_3 + x_4 + x_5 \\
& \text{s.t.} (-3 - 10p) x_3 + (-3 - 12p) x_4 + (9 - 10p) x_5 + x_9 = 0 \\
& \qquad \qquad \qquad (-1 - 1p) x_1 - x_2 + x_6 = 0 \\
& \qquad \qquad \qquad -x_1 + (-2 - p) x_2 + x_7 = 0 \\
& \qquad \qquad \qquad \qquad \qquad -20p x_3 + x_8 = 0 \\
& \qquad \qquad \qquad \qquad \qquad \qquad x_6 + x_8 = 8 \\
& \qquad \qquad \qquad \qquad \qquad \qquad x_7 + x_9 = 8 \\
& \qquad \qquad \qquad -10 y_1 + x_1 \leq 0 \qquad \qquad \qquad (\text{B.9}) \\
& \qquad \qquad \qquad -10 y_1 + x_2 \leq 0 \\
& \qquad \qquad \qquad -10 y_2 + x_3 \leq 0 \\
& \qquad \qquad \qquad -10 y_2 + x_4 \leq 0 \\
& \qquad \qquad \qquad -10 y_2 + x_5 \leq 0 \\
& \qquad \qquad \qquad \qquad \qquad x_1 + x_3 \leq 10 \\
& \qquad \qquad \qquad \qquad \qquad x_2 + x_4 \leq 10 \\
& \qquad \qquad \qquad \qquad \qquad \qquad \mathbf{y} \in \{0, 1\}^2 \\
& \qquad \qquad \qquad \qquad \qquad \qquad \mathbf{x} \in [0, 10]^9
\end{aligned}$$

has three optimality regions. For $p \in [0, 0.093]$ the optimal solution is $y_1 = 1, y_2 = 0$ with x_3, x_4, x_5, x_8 and x_9 fixed to zero. For $p \in [0.093, 0.299]$ the optimal solution is $y_1 = 0, y_2 = 1$ with x_1, x_2, x_4, x_6, x_7 fixed to zero. For $p \in [0.299, 1]$ the optimal solution is $y_1 = 0, y_2 = 1$ with x_1, x_2, x_5, x_6, x_7 fixed to zero.

Example B.10 (Small Superstructure). This example considers the flowsheet design

of a process superstructure (Figure B-1) for man-portable power generation devices subject to an unknown parameter. It corresponds to a simple variant of the superstructure described in [201] with the solid oxide fuel cell efficiency as the unknown parameter p . The MILP used does not correspond to a typical model, because we tried to introduce the least number of few variables and constraints possible; it contains 14 inequality constraints, 2 equality constraints, 7 continuous variables and 4 binary variables. The matrix contains 55 nonzero elements and only one element depends on the parameter. Note that since only one column is affected by the parameter variation, an analytical solution of the matrix inversion in the LPs is possible [117].

$$\begin{aligned}
& \min_{\mathbf{y}, \mathbf{x}} +0.058124 x_1 \\
& \text{s.t.} \quad -100 y_1 + x_2 \leq 0 \\
& \quad \quad -0.1 x_1 + x_2 \leq 0 \\
& \quad \quad -100 y_3 + x_3 \leq 0 \\
& \quad \quad -0.1 x_1 + x_2 + x_3 + x_4 \leq 0 \\
& +100 y_3 + 0.1 x_1 - x_2 - x_3 - x_4 \leq +100 \\
& \quad \quad -100 y_2 + x_4 \leq 0 \\
& \quad \quad -0.1 x_1 + x_2 + x_4 \leq 0 \\
& +100 y_2 + 0.1 x_1 - x_2 - x_4 \leq +100 \\
& \quad \quad -100 y_4 + x_5 \leq 0 \\
& -0.72 x_1 + 7.2 x_2 + 7.2 x_4 + x_5 \leq 0 \tag{B.10} \\
& +100 y_4 + 0.72 x_1 - 7.2 x_2 - 7.2 x_4 - x_5 \leq +100 \\
& \quad \quad -100 y_4 + x_6 \leq 0 \\
& \quad \quad -x_4 + x_6 \leq 0 \\
& +100 y_4 + x_4 - x_6 \leq +100 \\
& -0.090335 x_1 - 17.0224 x_2 - 10.8312 x_3 - 10.6295 x_4 \\
& \quad \quad -0.171977 x_5 - 4.15303 x_6 + x_7 = -1 \\
& -0.400388 x_1 + 5.02563 x_2 + 0.471905 x_3 \\
& +(4.95454 + -12.8319p) x_4 + 0.00813629 x_5 + 0.23758 x_6 = -1 \\
& \quad \quad \mathbf{y} \in \{0, 1\}^4 \\
& \quad \quad \mathbf{x} \in \mathbb{R}^7, \quad \mathbf{x} \geq \mathbf{0}
\end{aligned}$$

We considered only $p \in [0, 0.97]$, because after 0.97 the current basis becomes infeasible.

ble. There are two optimal solutions with optimality intervals $[0:0.35]$ and $[0:35,0.97]$.

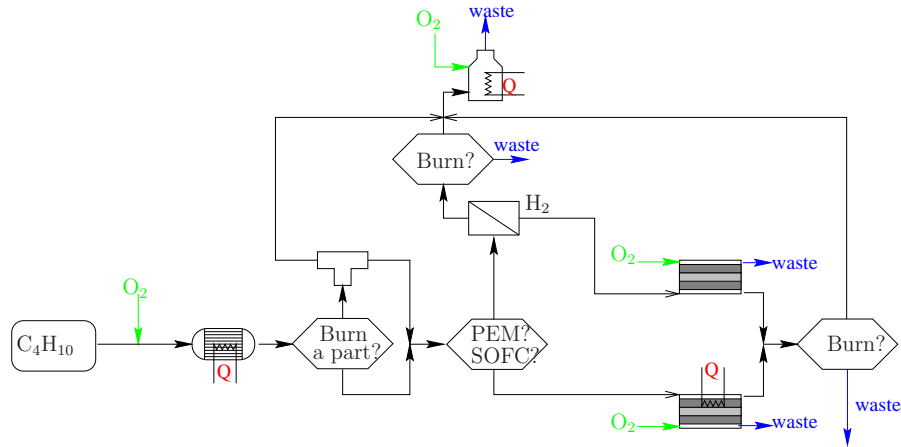


Figure B-1: Small micropower case study.

Example B.11 (Big Superstructure). This example considers the flowsheet design of a process superstructure (Figure B-2). It corresponds to a simple variant of the superstructure described in [201] with the solid oxide fuel cell efficiency as the unknown parameter p . The model is more typical than (B.10) and contains many intermediate variables and equality constraints. The resulting system has a total of $m_1 = 216$ inequality constraints, $m_2 = 226$ equality constraints, $n_x = 329$ continuous variables, $n_y = 20$ binary variables, while the matrix (before the augmentation) contains 1253 nonzero entries. Only one row depends on the parameter. Note that since only one column is affected by the parameter variation, an analytical solution of the matrix inversion in the LPs is possible [117].

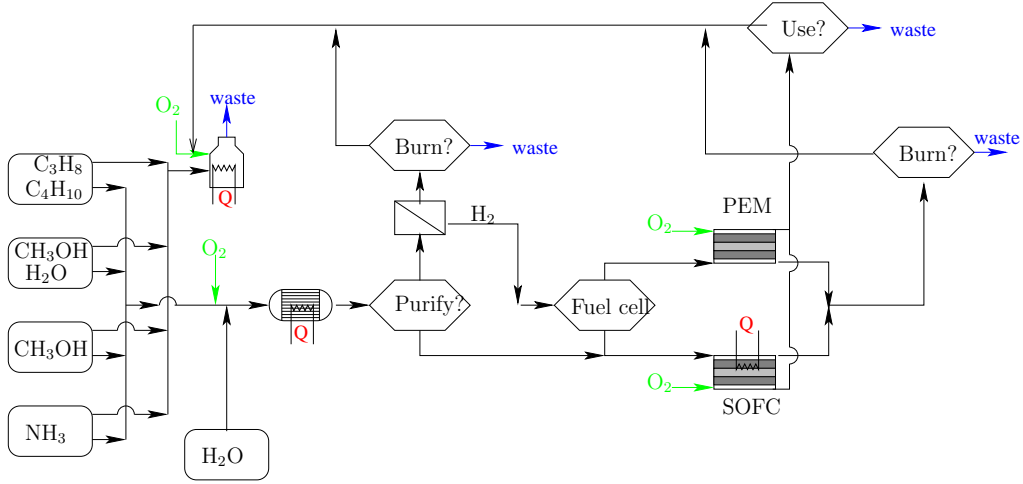


Figure B-2: Larger micropower case study.

$$\begin{aligned}
 \min_{\mathbf{y}, \mathbf{x}} & +0.0001 y_{14} + 0.0001 y_{15} + 0.0001 y_{16} + 0.0001 y_{17} + x_{326} \\
 \text{s.t.} & +y_{12} - y_{13} \leq 0 \\
 & -y_{11} + y_{15} \leq 0 \\
 & +y_{10} - y_{11} \leq 0 \\
 & +100 y_{12} + x_{38} \leq +100 \\
 & +100 y_{12} + x_{39} \leq +100 \\
 & +100 y_{12} + x_{40} \leq +100 \\
 & +100 y_{12} + x_{41} \leq +100 \\
 & -3.76 x_{34} + x_{38} \leq 0 \\
 & -3.76 x_{35} + x_{39} \leq 0 \\
 & -3.76 x_{36} + x_{40} \leq 0 \\
 & -3.76 x_{37} + x_{41} \leq 0 \\
 & -100 y_{12} + 3.76 x_{34} - x_{38} \leq 0 \\
 & -100 y_{12} + 3.76 x_{35} - x_{39} \leq 0 \\
 & -100 y_{12} + 3.76 x_{36} - x_{40} \leq 0 \\
 & -100 y_{12} + 3.76 x_{37} - x_{41} \leq 0 \\
 & -0.031999 x_{30} - 0.028013 x_{31} + x_{32} \leq 0 \\
 & -6.0012 y_{13} + x_{32} \leq 0 \\
 & +6.0012 y_{13} + 0.031999 x_{30} + 0.028013 x_{31} - x_{32} \leq +6.0012 \\
 & -0.247757 x_{30} - 0.247757 x_{31} + x_{42} \leq 0 \\
 & -49.5514 y_{13} + x_{42} \leq 0 \\
 & +49.5514 y_{13} + 0.247757 x_{30} + 0.247757 x_{31} - x_{42} \leq +49.5514 \\
 & -0.00298 x_{30} - 0.00298 x_{31} + x_{33} \leq 0
 \end{aligned} \tag{B.11}$$

$$\begin{aligned}
& +0.596 y_{13} + x_{33} \leq +0.596 \\
-0.00596 y_{13} + 0.00298 x_{30} + 0.00298 x_{31} - x_{33} & \leq 0 \\
& -x_1 + x_2 \leq 0 \\
& -100 y_{18} + x_2 \leq 0 \\
+100 y_{18} + x_1 - x_2 & \leq +100 \\
& -x_1 + x_9 \leq 0 \\
& -100 y_{19} + x_9 \leq 0 \\
+100 y_{19} + x_1 - x_9 & \leq +100 \\
& -x_1 + x_{10} \leq 0 \\
& -100 y_{20} + x_{10} \leq 0 \\
+100 y_{20} + x_1 - x_{10} & \leq +100 \\
& -x_{14} + x_{15} \leq 0 \\
& -100 y_1 + x_{15} \leq 0 \\
+100 y_1 + x_{14} - x_{15} & \leq +100 \\
& -x_{14} + x_{22} \leq 0 \\
& -100 y_2 + x_{22} \leq 0 \\
+100 y_2 + x_{14} - x_{22} & \leq +100 \\
& -x_{14} + x_{23} \leq 0 \\
& -100 y_6 + x_{23} \leq 0 \\
+100 y_6 + x_{14} - x_{23} & \leq +100 \\
& -100 y_{11} + x_{114} \leq 0 \\
& -100 y_{11} + x_{115} \leq 0 \\
& -100 y_{11} + x_{116} \leq 0 \\
& -100 y_{11} + x_{117} \leq 0 \\
& -100 y_{11} + x_{118} \leq 0 \\
& -100 y_{11} + x_{119} \leq 0 \\
& -100 y_{11} + x_{120} \leq 0 \\
& -100 y_{11} + x_{121} \leq 0 \\
& -100 y_{11} + x_{122} \leq 0 \\
& -100 y_{11} + x_{123} \leq 0 \\
& +100 y_{11} + x_{95} \leq +100 \\
& +100 y_{11} + x_{96} \leq +100 \\
& +100 y_{11} + x_{97} \leq +100 \\
& +100 y_{11} + x_{98} \leq +100 \\
& +100 y_{11} + x_{99} \leq +100 \\
& +100 y_{11} + x_{100} \leq +100 \\
& +100 y_{11} + x_{101} \leq +100 \\
& +100 y_{11} + x_{102} \leq +100 \\
& +100 y_{11} + x_{103} \leq +100
\end{aligned}$$

$$\begin{aligned}
+100 y_{11} + x_{104} &\leq +100 \\
-100 y_{15} + x_{154} &\leq 0 \\
-100 y_{15} + x_{155} &\leq 0 \\
-100 y_{15} + x_{156} &\leq 0 \\
-100 y_{15} + x_{157} &\leq 0 \\
-100 y_{15} + x_{158} &\leq 0 \\
-100 y_{15} + x_{159} &\leq 0 \\
-100 y_{15} + x_{160} &\leq 0 \\
-100 y_{15} + x_{161} &\leq 0 \\
-100 y_{15} + x_{162} &\leq 0 \\
-100 y_{15} + x_{163} &\leq 0 \\
+100 y_{15} + x_{144} &\leq +100 \\
+100 y_{15} + x_{145} &\leq +100 \\
+100 y_{15} + x_{146} &\leq +100 \\
+100 y_{15} + x_{147} &\leq +100 \\
+100 y_{15} + x_{148} &\leq +100 \\
+100 y_{15} + x_{149} &\leq +100 \\
+100 y_{15} + x_{150} &\leq +100 \\
+100 y_{15} + x_{151} &\leq +100 \\
+100 y_{15} + x_{152} &\leq +100 \\
+100 y_{15} + x_{153} &\leq +100 \\
-100 y_9 + x_{248} &\leq 0 \\
-100 y_9 + x_{249} &\leq 0 \\
-100 y_9 + x_{250} &\leq 0 \\
-100 y_9 + x_{251} &\leq 0 \\
-100 y_9 + x_{252} &\leq 0 \\
-100 y_9 + x_{253} &\leq 0 \\
-100 y_9 + x_{254} &\leq 0 \\
-100 y_9 + x_{255} &\leq 0 \\
-100 y_9 + x_{256} &\leq 0 \\
-100 y_9 + x_{257} &\leq 0 \\
-100 y_{10} + x_{174} &\leq 0 \\
-100 y_{10} + x_{175} &\leq 0 \\
-100 y_{10} + x_{176} &\leq 0 \\
-100 y_{10} + x_{177} &\leq 0 \\
-100 y_{10} + x_{178} &\leq 0 \\
-100 y_{10} + x_{179} &\leq 0
\end{aligned}
\tag{B.12}$$

$$\begin{aligned}
-100 y_{10} + x_{180} &\leq 0 \\
-100 y_{10} + x_{181} &\leq 0 \\
-100 y_{10} + x_{182} &\leq 0 \\
-100 y_{10} + x_{183} &\leq 0 \\
-100 y_{16} + x_{280} &\leq 0 \\
-100 y_{16} + x_{281} &\leq 0 \\
-100 y_{16} + x_{282} &\leq 0 \\
-100 y_{16} + x_{283} &\leq 0 \\
-100 y_{16} + x_{284} &\leq 0 \\
-100 y_{16} + x_{285} &\leq 0 \\
-100 y_{16} + x_{286} &\leq 0 \\
-100 y_{16} + x_{287} &\leq 0 \\
-100 y_{16} + x_{288} &\leq 0 \\
-100 y_{16} + x_{289} &\leq 0 \\
+100 y_{16} + x_{270} &\leq +100 \\
+100 y_{16} + x_{271} &\leq +100 \\
+100 y_{16} + x_{272} &\leq +100 \\
+100 y_{16} + x_{273} &\leq +100 \\
+100 y_{16} + x_{274} &\leq +100 \\
+100 y_{16} + x_{275} &\leq +100 \\
+100 y_{16} + x_{276} &\leq +100 \\
+100 y_{16} + x_{277} &\leq +100 \\
+100 y_{16} + x_{278} &\leq +100 \\
+100 y_{16} + x_{279} &\leq +100 \\
-100 y_{17} + x_{310} &\leq 0 \\
-100 y_{17} + x_{311} &\leq 0 \\
-100 y_{17} + x_{312} &\leq 0 \\
-100 y_{17} + x_{313} &\leq 0 \\
-100 y_{17} + x_{314} &\leq 0 \\
-100 y_{17} + x_{315} &\leq 0 \\
-100 y_{17} + x_{316} &\leq 0 \\
-100 y_{17} + x_{317} &\leq 0 \\
-100 y_{17} + x_{318} &\leq 0 \\
-100 y_{17} + x_{319} &\leq 0 \\
+100 y_{17} + x_{300} &\leq +100 \\
+100 y_{17} + x_{301} &\leq +100 \\
+100 y_{17} + x_{302} &\leq +100 \\
+100 y_{17} + x_{303} &\leq +100 \\
+100 y_{17} + x_{304} &\leq +100
\end{aligned}$$

(B.13)

$$\begin{aligned}
+100 y_{17} + x_{305} &\leq +100 \\
+100 y_{17} + x_{306} &\leq +100 \\
+100 y_{17} + x_{307} &\leq +100 \\
+100 y_{17} + x_{308} &\leq +100 \\
+100 y_{17} + x_{309} &\leq +100 \\
-100 y_{16} + x_{206} &\leq 0 \\
-100 y_{16} + x_{207} &\leq 0 \\
-100 y_{16} + x_{208} &\leq 0 \\
-100 y_{16} + x_{209} &\leq 0 \\
-100 y_{16} + x_{210} &\leq 0 \\
-100 y_{16} + x_{211} &\leq 0 \\
-100 y_{16} + x_{212} &\leq 0 \\
-100 y_{16} + x_{213} &\leq 0 \\
-100 y_{16} + x_{214} &\leq 0 \\
-100 y_{16} + x_{215} &\leq 0 \\
+100 y_{16} + x_{196} &\leq +100 \\
+100 y_{16} + x_{197} &\leq +100 \\
+100 y_{16} + x_{198} &\leq +100 \\
+100 y_{16} + x_{199} &\leq +100 \\
+100 y_{16} + x_{200} &\leq +100 \\
+100 y_{16} + x_{201} &\leq +100 \\
+100 y_{16} + x_{202} &\leq +100 \\
+100 y_{16} + x_{203} &\leq +100 \\
+100 y_{16} + x_{204} &\leq +100 \\
+100 y_{16} + x_{205} &\leq +100 \\
-100 y_{17} + x_{236} &\leq 0 \\
-100 y_{17} + x_{237} &\leq 0 \\
-100 y_{17} + x_{238} &\leq 0 \\
-100 y_{17} + x_{239} &\leq 0 \\
-100 y_{17} + x_{240} &\leq 0 \\
-100 y_{17} + x_{241} &\leq 0 \\
-100 y_{17} + x_{242} &\leq 0 \\
-100 y_{17} + x_{243} &\leq 0 \\
-100 y_{17} + x_{244} &\leq 0 \\
-100 y_{17} + x_{245} &\leq 0 \\
+100 y_{17} + x_{226} &\leq +100 \\
+100 y_{17} + x_{227} &\leq +100 \\
+100 y_{17} + x_{228} &\leq +100 \\
+100 y_{17} + x_{229} &\leq +100
\end{aligned}$$

(B.14)

$$\begin{aligned}
+100 y_{17} + x_{230} &\leq +100 \\
+100 y_{17} + x_{231} &\leq +100 \\
+100 y_{17} + x_{232} &\leq +100 \\
+100 y_{17} + x_{233} &\leq +100 \\
+100 y_{17} + x_{234} &\leq +100 \\
+100 y_{17} + x_{235} &\leq +100 \\
-100 y_1 + x_{105} &\leq 0 \\
-0.45 x_{72} + x_{105} &\leq 0 \\
-100 y_3 + x_{106} &\leq 0 \\
-0.9 x_{79} + x_{106} &\leq 0 \\
-100 y_4 + x_{107} &\leq 0 \\
-0.9 x_{79} + x_{107} &\leq 0 \\
-100 y_5 + x_{108} &\leq 0 \\
-0.45 x_{79} + x_{108} &\leq 0 \\
-100 y_7 + x_{109} &\leq 0 \\
-0.9 x_{80} + x_{109} &\leq 0 \\
-100 y_7 + x_{110} &\leq 0 \\
-0.9 x_{81} + x_{110} &\leq 0 \\
-100 y_8 + x_{111} &\leq 0 \\
-0.45 x_{80} + x_{111} &\leq 0 \\
-100 y_8 + x_{112} &\leq 0 \\
-0.9 x_{81} + x_{112} &\leq 0 \\
+100 y_7 - x_{76} + 3 x_{80} + 4 x_{81} - x_{84} &\leq +100 \\
+100 y_4 - x_{76} + x_{79} - x_{84} &\leq +100 \\
+100 y_5 - x_{76} + x_{79} - x_{84} &\leq +100 \\
-x_{74} - x_{82} + 1.2 x_{108} + 3.6 x_{111} + 2.4 x_{112} &\leq 0 \\
-0.2375 x_{65} &\leq 0 \\
-0.475 x_{66} &\leq 0 \\
-0.475 x_{67} &\leq 0 \\
-0.475 x_{68} &\leq 0 \\
-0.95 x_{69} &\leq 0 \\
-0.475 x_{70} &\leq 0 \\
-x_{45} - x_{53} + 3.6 x_{65} + 1.2 x_{66} + 1.2 x_{67} + 3.6 x_{68} + 6 x_{69} + 15.6 x_{70} &\leq 0 \\
-0.4 x_{247} &\leq 0 \\
-x_{184} + 1.2 x_{247} &\leq 0 \\
+x_{246} - 0.315491 x_{247} &\leq 0 \\
-0.4 x_{321} &\leq 0 \\
-0.4 x_{322} &\leq 0 \\
-x_{258} + 1.5 x_{321} + 1.5 x_{322} &\leq 0
\end{aligned}
\tag{B.15}$$

$$\begin{aligned}
& +x_{320}(0 + -0.420522p) x_{321}(0 + -0.401527p) x_{322} \leq 0 \\
& \qquad \qquad \qquad +y_1 + y_2 + y_6 = + \\
& \qquad \qquad \qquad -y_2 + y_3 + y_4 + y_5 = 0 \\
& \qquad \qquad \qquad -y_{14} + y_{18} + y_{19} + y_{20} = 0 \\
& \qquad \qquad \qquad -y_6 + y_7 + y_8 = 0 \\
& \qquad \qquad \qquad +y_9 + y_{10} = + \\
& +x_{30} - x_{34} - x_{35} - x_{36} - x_{37} = 0 \\
& +x_{31} - x_{38} - x_{39} - x_{40} - x_{41} = 0 \\
& \qquad \qquad \qquad -0.018015 x_{27} + x_{28} = 0 \\
& \qquad \qquad \qquad -0.018066 x_{27} + x_{29} = 0 \\
& \qquad \qquad \qquad +x_3 = 0 \\
& \qquad \qquad \qquad +x_4 = 0 \\
& \qquad \qquad \qquad +x_7 = 0 \\
& \qquad \qquad \qquad +x_8 = 0 \\
& \qquad \qquad \qquad +x_5 = 0 \\
& \qquad \qquad \qquad +x_6 = 0 \\
& \qquad \qquad \qquad -x_{10} + x_{11} = 0 \\
& -0.017031 x_2 - 0.028013 x_3 - 0.031999 x_4 - 0.002016 x_5 - 0.018015 x_6 - 0.02801 x_7 \\
& \qquad \qquad \qquad -0.04401 x_8 - 0.032042 x_9 - 0.044094 x_{10} - 0.058124 x_{11} + x_{12} = 0 \\
& -0.028 x_2 - 0.247757 x_3 - 0.247757 x_4 - 0.247757 x_5 - 0.018066 x_6 \\
& \qquad \qquad \qquad -2.4709 x_7 - 2.35 x_8 - 0.04 x_9 - 0.089567 x_{10} - 0.1012 x_{11} + x_{13} = 0 \\
& \qquad \qquad \qquad +x_{16} = 0 \\
& \qquad \qquad \qquad +x_{17} = 0 \\
& \qquad \qquad \qquad +x_{20} = 0 \\
& \qquad \qquad \qquad +x_{21} = 0 \\
& \qquad \qquad \qquad +x_{18} = 0 \\
& \qquad \qquad \qquad +x_{19} = 0 \\
& \qquad \qquad \qquad -x_{23} + x_{24} = 0 \\
& -0.017031 x_{15} - 0.028013 x_{16} - 0.031999 x_{17} - 0.002016 x_{18} - 0.018015 x_{19} - 0.02801 x_{20} \\
& \qquad \qquad \qquad -0.04401 x_{21} - 0.032042 x_{22} - 0.044094 x_{23} - 0.058124 x_{24} + x_{25} = 0 \\
& -0.028 x_{15} - 0.247757 x_{16} - 0.247757 x_{17} - 0.247757 x_{18} - 0.018066 x_{19} - 2.4709 x_{20} \\
& \qquad \qquad \qquad -2.35 x_{21} - 0.04 x_{22} - 0.089567 x_{23} - 0.1012 x_{24} + x_{26} = 0 \\
& \qquad \qquad \qquad -x_{15} + x_{72} = 0 \\
& \qquad \qquad \qquad -x_{16} + x_{73} = 0 \\
& \qquad \qquad \qquad -x_{17} + x_{74} = 0 \\
& \qquad \qquad \qquad -x_{18} + x_{75} = 0 \\
& \qquad \qquad \qquad -x_{19} + x_{76} = 0 \\
& \qquad \qquad \qquad -x_{20} + x_{77} = 0 \quad (\text{B.16})
\end{aligned}$$

$$\begin{aligned}
-x_{21} + x_{78} &= 0 \\
-x_{22} + x_{79} &= 0 \\
-x_{23} + x_{80} &= 0 \\
-x_{24} + x_{81} &= 0 \\
-x_{34} + x_{82} &= 0 \\
-x_{38} + x_{83} &= 0 \\
-x_{27} + x_{84} &= 0 \\
+x_{85} - x_{95} - x_{114} &= 0 \\
+x_{86} - x_{96} - x_{115} &= 0 \\
+x_{87} - x_{97} - x_{116} &= 0 \\
+x_{88} - x_{98} - x_{117} &= 0 \\
+x_{89} - x_{99} - x_{118} &= 0 \\
+x_{90} - x_{100} - x_{119} &= 0 \\
+x_{91} - x_{101} - x_{120} &= 0 \\
+x_{92} - x_{102} - x_{121} &= 0 \\
+x_{93} - x_{103} - x_{122} &= 0 \\
+x_{94} - x_{104} - x_{123} &= 0 \\
+x_{134} - x_{144} - x_{154} &= 0 \\
+x_{135} - x_{145} - x_{155} &= 0 \\
+x_{136} - x_{146} - x_{156} &= 0 \\
+x_{137} - x_{147} - x_{157} &= 0 \\
+x_{138} - x_{148} - x_{158} &= 0 \\
+x_{139} - x_{149} - x_{159} &= 0 \\
+x_{140} - x_{150} - x_{160} &= 0 \\
+x_{141} - x_{151} - x_{161} &= 0 \\
+x_{142} - x_{152} - x_{162} &= 0 \\
+x_{143} - x_{153} - x_{163} &= 0 \\
-x_{95} - x_{124} + x_{164} &= 0 \\
-x_{96} - x_{125} + x_{165} &= 0 \\
-x_{97} - x_{126} + x_{166} &= 0 \\
-x_{98} - x_{127} + x_{167} &= 0 \\
-x_{99} - x_{128} + x_{168} &= 0 \\
-x_{100} - x_{129} + x_{169} &= 0 \\
-x_{101} - x_{130} + x_{170} &= 0 \\
-x_{102} - x_{131} + x_{171} &= 0 \\
-x_{103} - x_{132} + x_{172} &= 0 \\
-x_{104} - x_{133} + x_{173} &= 0 \\
+x_{164} - x_{174} - x_{248} &= 0
\end{aligned}$$

(B.17)

$$\begin{aligned}
+x_{165} - x_{175} - x_{249} &= 0 \\
+x_{166} - x_{176} - x_{250} &= 0 \\
+x_{167} - x_{177} - x_{251} &= 0 \\
+x_{168} - x_{178} - x_{252} &= 0 \\
+x_{169} - x_{179} - x_{253} &= 0 \\
+x_{170} - x_{180} - x_{254} &= 0 \\
+x_{171} - x_{181} - x_{255} &= 0 \\
+x_{172} - x_{182} - x_{256} &= 0 \\
+x_{173} - x_{183} - x_{257} &= 0 \\
-x_{36} + x_{258} &= 0 \\
-x_{40} + x_{259} &= 0 \\
+x_{260} - x_{270} - x_{280} &= 0 \\
+x_{261} - x_{271} - x_{281} &= 0 \\
+x_{262} - x_{272} - x_{282} &= 0 \\
+x_{263} - x_{273} - x_{283} &= 0 \\
+x_{264} - x_{274} - x_{284} &= 0 \\
+x_{265} - x_{275} - x_{285} &= 0 \\
+x_{266} - x_{276} - x_{286} &= 0 \\
+x_{267} - x_{277} - x_{287} &= 0 \\
+x_{268} - x_{278} - x_{288} &= 0 \\
+x_{269} - x_{279} - x_{289} &= 0 \\
+x_{290} - x_{300} - x_{310} &= 0 \\
+x_{291} - x_{301} - x_{311} &= 0 \\
+x_{292} - x_{302} - x_{312} &= 0 \\
+x_{293} - x_{303} - x_{313} &= 0 \\
+x_{294} - x_{304} - x_{314} &= 0 \\
+x_{295} - x_{305} - x_{315} &= 0 \\
+x_{296} - x_{306} - x_{316} &= 0 \\
+x_{297} - x_{307} - x_{317} &= 0 \\
+x_{298} - x_{308} - x_{318} &= 0 \\
+x_{299} - x_{309} - x_{319} &= 0 \\
-x_{37} + x_{184} &= 0 \\
-x_{41} + x_{185} &= 0 \\
+x_{186} - x_{196} - x_{206} &= 0 \\
+x_{187} - x_{197} - x_{207} &= 0 \\
+x_{188} - x_{198} - x_{208} &= 0 \\
+x_{189} - x_{199} - x_{209} &= 0 \\
+x_{190} - x_{200} - x_{210} &= 0
\end{aligned}$$

(B.18)

$$\begin{aligned}
& +x_{191} - x_{201} - x_{211} = 0 \\
& +x_{192} - x_{202} - x_{212} = 0 \\
& +x_{193} - x_{203} - x_{213} = 0 \\
& +x_{194} - x_{204} - x_{214} = 0 \\
& +x_{195} - x_{205} - x_{215} = 0 \\
& +x_{216} - x_{226} - x_{236} = 0 \\
& +x_{217} - x_{227} - x_{237} = 0 \\
& +x_{218} - x_{228} - x_{238} = 0 \\
& +x_{219} - x_{229} - x_{239} = 0 \\
& +x_{220} - x_{230} - x_{240} = 0 \\
& +x_{221} - x_{231} - x_{241} = 0 \\
& +x_{222} - x_{232} - x_{242} = 0 \\
& +x_{223} - x_{233} - x_{243} = 0 \\
& +x_{224} - x_{234} - x_{244} = 0 \\
& +x_{225} - x_{235} - x_{245} = 0 \\
& -x_2 + x_{43} - x_{154} - x_{206} - x_{236} - x_{280} - x_{310} = 0 \\
& -x_3 + x_{44} - x_{155} - x_{207} - x_{237} - x_{281} - x_{311} = 0 \\
& -x_4 + x_{45} - x_{156} - x_{208} - x_{238} - x_{282} - x_{312} = 0 \\
& -x_5 + x_{46} - x_{157} - x_{209} - x_{239} - x_{283} - x_{313} = 0 \\
& -x_6 + x_{47} - x_{158} - x_{210} - x_{240} - x_{284} - x_{314} = 0 \\
& -x_7 + x_{48} - x_{159} - x_{211} - x_{241} - x_{285} - x_{315} = 0 \\
& -x_8 + x_{49} - x_{160} - x_{212} - x_{242} - x_{286} - x_{316} = 0 \\
& -x_9 + x_{50} - x_{161} - x_{213} - x_{243} - x_{287} - x_{317} = 0 \\
& -x_{10} + x_{51} - x_{162} - x_{214} - x_{244} - x_{288} - x_{318} = 0 \\
& -x_{11} + x_{52} - x_{163} - x_{215} - x_{245} - x_{289} - x_{319} = 0 \\
& \quad \quad \quad -x_{35} + x_{53} = 0 \\
& \quad \quad \quad -x_{39} + x_{54} = 0 \\
& \quad \quad \quad -x_{71} - x_{113} - x_{323} + x_{329} = 0 \\
& \quad \quad \quad +x_{328} - 12x_{329} = +0.24 \\
& -0.04572x_2 - 0.285658x_6 - 0.1106x_7 - 0.3938x_8 - 0.237793x_9 - 0.1039x_{10} - 0.1262x_{11} \\
& \quad -0.04572x_{15} - 0.285658x_{19} - 0.1106x_{20} - 0.3938x_{21} - 0.237793x_{22} - 0.1039x_{23} \\
& \quad -0.1262x_{24} - 0.285658x_{27} + 0.03165x_{55} - 0.011256x_{56} - 0.012462x_{57} - 0.011658x_{58} \\
& \quad +0.229538x_{59} + 0.099344x_{60} + 0.3737x_{61} + 0.1812x_{62} + 0.02752x_{63} + 0.04982x_{64} \\
& \quad +0.03165x_{144} - 0.011256x_{145} - 0.012462x_{146} - 0.011658x_{147} + 0.229538x_{148} \\
& \quad +0.099344x_{149} + 0.3737x_{150} + 0.1812x_{151} + 0.02752x_{152} + 0.04982x_{153} + 0.03165x_{174} \\
& -0.011256x_{175} - 0.012462x_{176} - 0.011658x_{177} + 0.229538x_{178} + 0.099344x_{179} + 0.3737x_{180} \\
& \quad +0.1812x_{181} + 0.02752x_{182} + 0.04982x_{183} - 0.0418x_{206} + 0.003136x_{207} + 0.003472x_{208}
\end{aligned}$$

(B.19)

$$\begin{aligned}
& +0.003248 x_{209} - 0.238528 x_{210} - 0.107464 x_{211} - 0.3882 x_{212} - 0.1957 x_{213} - 0.08262 x_{214} \\
& -0.10492 x_{215} - 0.0418 x_{236} + 0.003136 x_{237} + 0.003472 x_{238} + 0.003248 x_{239} - 0.238528 x_{240} \\
& \quad -0.107464 x_{241} - 0.3882 x_{242} - 0.1957 x_{243} - 0.08262 x_{244} - 0.10492 x_{245} + 0.03165 x_{270} \\
& -0.011256 x_{271} - 0.012462 x_{272} - 0.011658 x_{273} + 0.229538 x_{274} + 0.099344 x_{275} + 0.3737 x_{276} \\
& \quad +0.1812 x_{277} + 0.02752 x_{278} + 0.04982 x_{279} + 0.03165 x_{300} - 0.011256 x_{301} - 0.012462 x_{302} \\
& \quad -0.011658 x_{303} + 0.229538 x_{304} + 0.099344 x_{305} + 0.3737 x_{306} + 0.1812 x_{307} + 0.02752 x_{308} \\
& \quad +0.04982 x_{309} - x_{320} - x_{327} - 0.915675 x_{328} = 0 \\
& \quad \quad \quad +x_{33} - x_{246} - x_{320} + x_{324} = 0 \\
& \quad \quad \quad \quad \quad \quad \quad +x_{324} = + \\
& \quad \quad \quad -x_{12} - x_{25} - x_{28} - x_{32} + x_{326} = 0 \\
& \quad \quad \quad -x_{13} - x_{26} - x_{29} - x_{42} + x_{325} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{72} + x_{85} + 2 x_{105} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{73} - x_{83} + x_{86} - x_{105} = 0 \\
& \quad \quad \quad -x_{74} - x_{82} + x_{87} + x_{108} + 3 x_{111} + 2 x_{112} = 0 \\
& -x_{75} + x_{88} - 3 x_{105} - 2 x_{106} - 3 x_{107} - 4 x_{108} - 7 x_{109} - 9 x_{110} - 8 x_{111} - 5 x_{112} = 0 \\
& \quad \quad \quad -x_{76} - x_{84} + x_{89} + x_{107} + 3 x_{109} + 4 x_{110} = 0 \\
& \quad \quad \quad -x_{77} + x_{90} - x_{106} - 3 x_{109} - 4 x_{110} - 6 x_{111} - 4 x_{112} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{78} + x_{91} - x_{107} - 2 x_{108} = 0 \\
& \quad \quad \quad -x_{79} + x_{92} + x_{106} + x_{107} + 2 x_{108} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{80} + x_{93} + x_{109} + 2 x_{111} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{81} + x_{94} + x_{110} + x_{112} = 0 \\
& \quad \quad \quad -7.38657e - 05 x_{85} - 7.38657e - 05 x_{86} - 7.38657e - 05 x_{87} \\
& -7.38657e - 05 x_{88} - 7.38657e - 05 x_{89} - 7.38657e - 05 x_{90} - 7.38657e - 05 x_{91} \\
& \quad \quad \quad -7.38657e - 05 x_{92} - 7.38657e - 05 x_{93} - 7.38657e - 05 x_{94} + x_{113} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{43} + x_{55} + 4 x_{65} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{44} - x_{54} + x_{56} - 2 x_{65} = 0 \\
& -x_{45} - x_{53} + x_{57} + 3 x_{65} + x_{66} + x_{67} + 3 x_{68} + 5 x_{69} + 13 x_{70} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{46} + x_{58} + 2 x_{66} = 0 \\
& \quad \quad \quad -x_{47} + x_{59} - 6 x_{65} - 2 x_{66} - 4 x_{68} - 4 x_{69} - 10 x_{70} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{48} + x_{60} + 2 x_{67} = 0 \\
& \quad \quad \quad -x_{49} + x_{61} - 2 x_{67} - 2 x_{68} - 3 x_{69} - 8 x_{70} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{50} + x_{62} + 2 x_{68} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{51} + x_{63} + x_{69} = 0 \\
& \quad \quad \quad \quad \quad \quad -x_{52} + x_{64} + 2 x_{70} = 0 \\
& -7.38657e - 05 x_{55} - 7.38657e - 05 x_{56} - 7.38657e - 05 x_{57} - 7.38657e - 05 x_{58} \\
& -7.38657e - 05 x_{59} - 7.38657e - 05 x_{60} - 7.38657e - 05 x_{61} - 7.38657e - 05 x_{62} \\
& \quad \quad \quad -7.38657e - 05 x_{63} - 7.38657e - 05 x_{64} + x_{71} = 0
\end{aligned}$$

(B.20)

$$\begin{aligned}
& +x_{124} = 0 \\
& +x_{125} = 0 \\
& +x_{126} = 0 \\
-0.8 x_{117} + x_{127} & = 0 \\
& +x_{128} = 0 \\
& +x_{129} = 0 \\
& +x_{130} = 0 \\
& +x_{131} = 0 \\
& +x_{132} = 0 \\
& +x_{133} = 0 \\
-x_{114} + x_{134} & = 0 \\
-x_{115} + x_{135} & = 0 \\
-x_{116} + x_{136} & = 0 \\
-0.2 x_{117} + x_{137} & = 0 \\
-x_{118} + x_{138} & = 0 \\
-x_{119} + x_{139} & = 0 \\
-x_{120} + x_{140} & = 0 \\
-x_{121} + x_{141} & = 0 \\
-x_{122} + x_{142} & = 0 \\
-x_{123} + x_{143} & = 0 \\
-x_{174} + x_{186} & = 0 \\
-x_{175} + x_{187} & = 0 \\
-x_{176} + x_{188} & = 0 \\
-0.2 x_{177} + x_{189} & = 0 \\
-x_{178} + x_{190} & = 0 \\
-x_{179} + x_{191} & = 0 \\
-x_{180} + x_{192} & = 0 \\
-x_{181} + x_{193} & = 0 \\
-x_{182} + x_{194} & = 0 \\
-x_{183} + x_{195} & = 0 \\
& +x_{216} = 0 \\
& +x_{217} = 0 \\
-x_{184} - x_{185} + x_{218} + x_{247} & = 0 \\
-0.8 x_{177} + x_{219} + 2 x_{247} & = 0 \\
+x_{220} - 2 x_{247} & = 0 \\
& +x_{221} = 0 \\
& +x_{222} = 0 \\
& +x_{223} = 0 \\
& +x_{224} = 0
\end{aligned}$$

(B.21)

$$\begin{aligned}
& +x_{225} = 0 \\
& -x_{248} + x_{260} = 0 \\
& -x_{249} + x_{261} = 0 \\
& -x_{250} - 0.666667 x_{258} + x_{262} + x_{321} + x_{322} = 0 \\
& -x_{251} + x_{263} + 2x_{321} = 0 \\
& -x_{252} + x_{264} - 2x_{321} = 0 \\
& -x_{253} + x_{265} + 2x_{322} = 0 \\
& -x_{254} + x_{266} - 2x_{322} = 0 \\
& -x_{255} + x_{267} = 0 \\
& -x_{256} + x_{268} = 0 \\
& -x_{257} + x_{269} = 0 \\
& +x_{290} = 0 \\
& -x_{259} + x_{291} = 0 \\
& -0.333333 x_{258} + x_{292} = 0 \\
& +x_{293} = 0 \\
& +x_{294} = 0 \\
& +x_{295} = 0 \\
& +x_{296} = 0 \\
& +x_{297} = 0 \\
& +x_{298} = 0 \\
& +x_{299} = 0 \\
& -0.00147731 x_{260} - 0.00147731 x_{261} - 0.00147731 x_{262} - 0.00147731 x_{263} \\
& -0.00147731 x_{264} - 0.00147731 x_{265} - 0.00147731 x_{266} - 0.00147731 x_{267} \\
& -0.00147731 x_{268} - 0.00147731 x_{269} - 0.00147731 x_{290} - 0.00147731 x_{291} \\
& -0.00147731 x_{292} - 0.00147731 x_{293} - 0.00147731 x_{294} - 0.00147731 x_{295} \\
& -0.00147731 x_{296} - 0.00147731 x_{297} - 0.00147731 x_{298} - 0.00147731 x_{299} + x_{323} = 0 \\
& \mathbf{y} \in \{0, 1\}^{20} \\
& \mathbf{x} \in \mathbb{R}^{329}, \quad \mathbf{x} \geq \mathbf{0} \quad (\text{B.22})
\end{aligned}$$

Appendix C

Bilevel Optimization Test Set

C.1 Original Examples

Example C.1 (Linear bilevel program with opposite objectives). The bilevel program

$$\begin{aligned} \min_y & y \\ \text{s.t. } & y \in \arg \min_z -z \\ & y, z \in [-1, 1] \end{aligned} \tag{mb_0_1_01}$$

has the unique optimal solution $y = 1$ with an objective value of 1.

This program shows that merely replacing the inner program with its constraints does not lead to a convergent lower bound. The inner program has the unique optimal solution $y = 1$ and therefore the bilevel program (mb_0_1_01) has the unique feasible point and unique global minimum $y = 1$. The inner program is linear and therefore the KKT conditions are both necessary and sufficient for a minimum. At the optimal solution, the KKT multiplier associated with the constraint $y \leq 1$ is equal to 1 and therefore any $\mu_{max} > 1$ can be used.

Example C.2 (Infeasible linear bilevel program). The bilevel program

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y \leq 0 \\
 & \quad y \in \arg \min_z -z \\
 & \quad y, z \in [-1, 1]
 \end{aligned} \tag{mb_0_1_02}$$

is infeasible.

This program shows that merely replacing the inner program with its constraints does not lead to a convergent lower bound. The inner program is linear and has the unique optimal solution $y = 1$ and therefore the bilevel program (mb_0_1_02) is infeasible. The inner program is linear and therefore the KKT conditions are both necessary and sufficient for a minimum. At the optimal solution, the KKT multiplier associated with the constraint $y \leq 1$ is equal to 1 and therefore any $\mu_{max} > 1$ can be used. Note that the outer constraint $y \leq 0$, cannot be included in the host set, because the problem would be changed.

Example C.3 (Concave inner constraint). The bilevel program

$$\begin{aligned}
 & \min_y y \\
 & \text{s.t. } y \in \arg \min_z z^2 \\
 & \quad \text{s.t. } 1 - z_2^2 \leq 0 \\
 & \quad y, z \in [-10, 10]
 \end{aligned} \tag{mb_0_1_03}$$

has the unique optimal solution $y = -1$ with an objective value of -1 .

The purpose of this example is to show that constructing a convex relaxation of the inner problem and replacing the relaxed program with its KKT conditions does not lead to a valid lower bound [197]. The feasible set of the inner program is $[-10, -1] \cup [1, 10]$ and its optimal solutions are $y = \pm 1$. The unique feasible solution of the bilevel program is therefore $y = -1$.

The inner program satisfies the linear/concave constraint qualification [41, p. 322] and at the constrained KKT points the multiplier associated with the concave constraint is given by $\mu = 1$ and therefore any $\mu_{max} > 1$ can be used.

Example C.4 (Concave inner objective). The bilevel program

$$\begin{aligned} & \min_y y \\ & \text{s.t. } y \in \arg \min_z -z^2 \\ & y, z \in [-0.5, 1] \end{aligned} \tag{mb_0_1_04}$$

has the unique optimal solution $y = 1$ with an objective value of 1.

This example shows that merely replacing the inner problem with its KKT constraints does not result to convergent lower bounds. It also shows that replacing the inner program with its KKT conditions and solving the resulting Mathematical Program with Equilibrium Constraints (MPEC) locally does not lead to valid upper bounds [197]. The inner program has the unique optimal solution $y = 1$ and therefore the bilevel program (mb_0_1_04) has the unique feasible point $y = 1$.

The inner program has a concave objective function and linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity not sufficient for a local or a global minimum, see also Figure C-1. The KKT conditions for the inner program are given

by

$$\begin{aligned} -0.5 - y &\leq 0 \\ y - 1 &\leq 0 \\ -2y - \lambda_1 + \lambda_2 &= 0 \\ \lambda_1(-y - 0.5) &= 0 \\ \lambda_2(y - 1) &= 0 \\ \lambda_1 &\geq 0 \\ \lambda_2 &\geq 0. \end{aligned} \tag{C.5}$$

System (C.5) has three solutions. The first solution is $y = -0.5$, $\lambda_1 = 1$, $\lambda_2 = 0$, a suboptimal local minimum of the inner program ($h = -0.25$). The second solution is $y = 0$, $\lambda_1 = 0$, $\lambda_2 = 0$, the global maximum of the inner program ($h = 0$). Finally the third solution is $y = 1$, $\lambda_1 = 0$, $\lambda_2 = 2$, the global minimum of the inner program ($h = -1$). Any $\mu_{max} \geq 0$ can be used as a bound for the KKT multipliers.

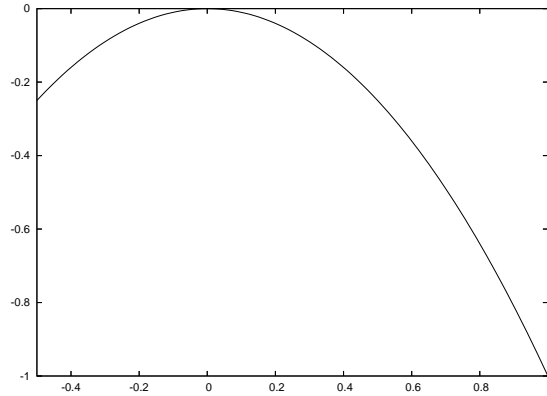


Figure C-1: Inner level objective function for Example C.4.

Example C.5 (Nonconvex inner objective). The bilevel program

$$\begin{aligned} & \min_y y \\ & \text{s.t. } y \in \arg \min_z 16z^4 + 2z^3 - 8z^2 - \frac{3}{2}z + \frac{1}{2} \\ & y, z \in [-1, 1] \end{aligned} \tag{mb_0_1_05}$$

has the unique optimal solution $y = \frac{1}{2}$ with an objective value of $\frac{1}{2}$.

This example shows that merely replacing the inner problem with its KKT constraints does not result to convergent lower bounds. The inner program has the unique optimal solution $y = \frac{1}{2}$ and therefore the bilevel program (mb_0_1_05) has the unique feasible point.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a local minimum. Due to nonconvexity of the objective function they are not sufficient for a local or global minimum, see also Figure C-2. Out of the three KKT points $y = \frac{1}{2}$ is the unique global minimum, $y = -\frac{1}{2}$ is a suboptimal local minimum and $y \approx -0.09375$ is a local maximum. The inequality constraints are inactive at all three KKT points and therefore any $\mu_{max} > 0$ can be used.

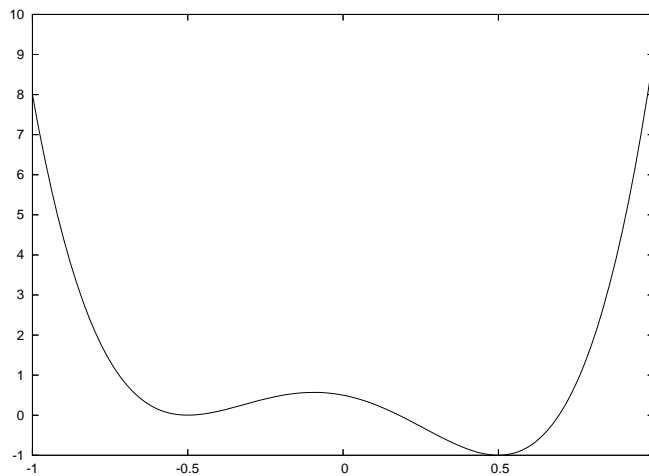


Figure C-2: Inner level objective function for Example C.5.

Example C.6 (Nonconvex monomial inner objective). The bilevel program

$$\begin{aligned} \min_y & \\ \text{s.t. } & y \in \arg \min_z z^3 \\ & y, z \in [-1, 1] \end{aligned} \tag{mb_0_1_06}$$

has the unique optimal solution $y = -1$ with an objective value of -1 .

The purpose of this example is to show that constructing a convex relaxation of the inner problem and replacing the relaxed program with its KKT conditions does not lead to a valid lower bound [197]. The inner problem has the unique optimal solution $y = -1$ and therefore the bilevel program has a single feasible point. The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a minimum. They are not sufficient since $y = 0$ satisfies the KKT conditions (unconstrained) but is a saddle point. Note that the inner objective is strictly monotone increasing and points to the same direction as the outer objective. At the constrained KKT points the value of the KKT multiplier is given by $3y^2 = 3$ and therefore any $\mu_{max} \geq 3$ can be used.

Example C.7 (Infeasible linear program). The bilevel program

$$\begin{aligned} \min_y & \\ \text{s.t. } & y \leq 0 \\ & y \in \arg \min_z -z \\ & y, z \in [-1, 1] \end{aligned} \tag{mb_0_1_07}$$

is infeasible.

The purpose of this program is to study infeasible problems. The unique solution to the inner problem is $y = 1$ and therefore (mb_0_1_07) is infeasible. The inner program is linear and therefore the KKT conditions are both necessary and sufficient for a minimum. At the optimal solution, the KKT multiplier associated with the

constraint $y \leq 1$ is equal to 1.

Example C.8 (Linear inner program). The bilevel program

$$\begin{aligned}
 & \min_{x,y} y^2 \\
 & \text{s.t. } y - 0.1 \leq 0 \\
 & \quad -y - 0.1 \leq 0 \\
 & \quad y \in \arg \min_z z(x + \exp(x)) \\
 & x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned} \tag{mb_1.1_01}$$

has the unique optimal solution $x = \bar{x}$, $y = 0$ and an optimal objective value of 0, where $\bar{x} \approx -0.567$ is the solution of $x + \exp(x) = 0$.

The purpose of this example is to show that a discretization approach on the variables \mathbf{x} can fail to find feasible points even for simple programs. The inner program is linear in y and therefore the KKT conditions are both necessary and sufficient for a minimum. Let $\bar{x} \approx -0.567$ denote the solution of $x + \exp(x) = 0$. For $x < \bar{x}$ the unique optimal solution of the inner problem is $y = 1$ which is infeasible by the outer constraint $y \leq 0.1$; for $x > \bar{x}$ the unique optimal solution of the inner problem is $y = -1$ which is infeasible by the outer constraint $y \geq -0.1$. For $x = \bar{x}$ any $y \in [-1, 1]$ is optimal for the inner problem. Therefore the feasible set of the bilevel program is $x = \bar{x}$, $y \in [-0.1, 0.1]$ which leads to the unique optimal solution of the bilevel program $x = \bar{x}$, $y = 0$. The magnitude of the KKT multipliers for the inner program can be estimated by $|x + \exp(x)| < 4$.

Example C.9 (Nonconvex monomial inner objective). The bilevel program

$$\begin{aligned}
 & \min_{x,y} x \\
 & \text{s.t. } -x + y \leq 0 \\
 & \quad y \in \arg \min_z z^3 \\
 & x \in [-10, 10], \quad y, z \in [-1, 1]
 \end{aligned} \tag{mb_1.1_02}$$

has the unique optimal solution $x = -1, y = -1$ with an objective value of -1 .

This example builds upon Example C.6. The outer variable x does not add much complication to the analysis, since the inner program is not parameterized by it.

Example C.10 (Nonconvex inner objective parameterized in x). The bilevel program

$$\begin{aligned} \min_{x,y} y \\ \text{s.t. } y \in \arg \min_z x \left(16z^4 + 2z^3 - 8z^2 - \frac{3}{2}z + \frac{1}{2} \right) \quad (\text{mb_1.1.03}) \\ x \in [0.1, 1], \quad y, z \in [-1, 1] \end{aligned}$$

has infinitely many optimal solution points $x \in [0.1, 1], y = 0.5$ with an optimal objective value of 0.5 .

This example builds upon Example C.5 introducing also a variable x . This addition demonstrates algorithmic difficulties when the set of suboptimal KKT points exists for each x . The analysis is equivalent to Example C.5. Figure C-3 shows a plot of the inner objective.

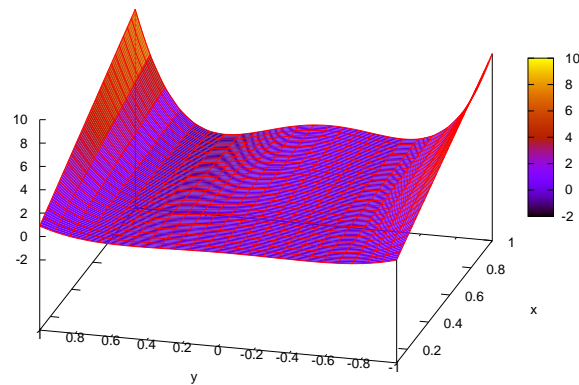


Figure C-3: Inner level objective function for Example C.10.

Example C.11 (Nonconvex inner objective parameterized in x). The bilevel program

$$\begin{aligned} \min_{x,y} y \\ \text{s.t. } y \in \arg \min_z x (16z^4 + 2z^3 - 8z^2 - \frac{3}{2}z + \frac{1}{2}) \quad (\text{mb_1_1_04}) \\ x \in [-1, 1], \quad y, z \in [-0.8, 1] \end{aligned}$$

has the unique optimal solution point $x = 0$, $y = -0.8$ with an optimal objective value of -0.8 .

This example builds upon Example C.5 and is a variation of Example C.10. For $x > 0$ the inner objective has the unconstrained unique global minimum $y = 0.5$. For $x < 0$ the inner objective has the constrained unique global minimum $y = 1$. Finally for $x = 0$ any $y \in [-0.8, 1]$ is trivially optimal in the inner program. Therefore the unique optimal solution of the bilevel program $x = 0$, $y = -0.8$.

Similarly to Example C.5 the inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but not sufficient for a minimum.

In addition to the global minima there are several suboptimal KKT points. For $x > 0$, similarly to Example C.5 $y = -\frac{1}{2}$ is a suboptimal local minimum and $y \approx -0.09375$ is a local maximum. For $x < 0$ $y = \pm\frac{1}{2}$ are unconstrained local maxima, $y \approx -0.09375$ is an unconstrained suboptimal local minimum and $y = -0.8$ is a constrained local minimum. Figure C-4 shows a plot of the inner objective and the minima and suboptimal KKT points of the inner program.

The values of the KKT multipliers can be bounded by the derivative of the inner objective function $|x(64y^3 + 6y^2 - 16y - 1.5)|$ evaluated at $y = -0.8$ and $y = 1$ which gives $|17.628x|$ and $|52.5x|$ respectively. Therefore $\mu_{max} = 100$ is a conservative estimate.

Example C.12 (Supercritical pitchfork bifurcation with x as the parameter). The

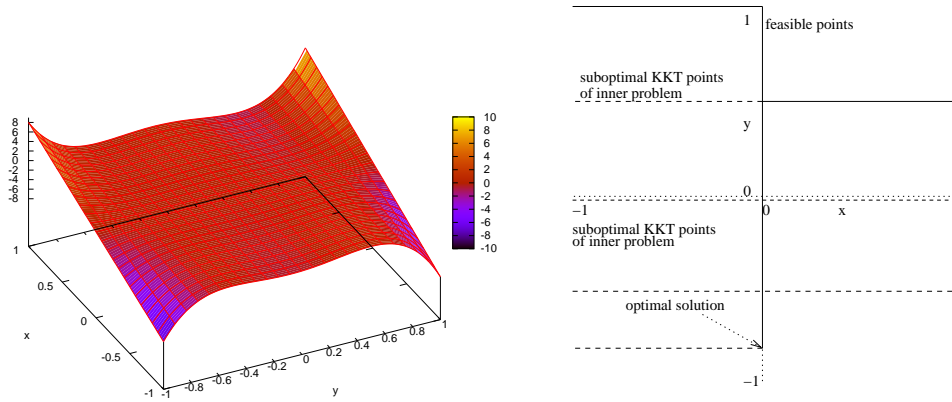


Figure C-4: Inner level objective function, its KKT points and its minima for Example C.11.

bilevel program

$$\begin{aligned}
 & \min_{x,y} -x + xy + 10y^2 \\
 & \text{s.t. } y \in \arg \min_z -x z^2 + \frac{z^4}{2} \quad (\text{mb_1_1_05}) \\
 & \quad x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned}$$

has the unique optimal solution $x = 0, y = 0$ with an objective value of 0.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

Stationarity of the inner objective gives $-2xy + 2y^3 = 0$, which describes a bifurcation for y as a function of x , see also Figure C-5. For $-1 \leq x \leq 0$, we have $y = 0$ as the unique global optimum and only KKT point of the inner problem due to strict convexity. For $0 < x \leq 1$, $y = \pm\sqrt{x}$ are the global minima for the inner problem while $y = 0$ is a suboptimal KKT point of the inner problem. All KKT points are unconstrained, so that any $\mu_{max} > 0$ can be used.

Example C.13 (Supercritical pitchfork bifurcation with y as the parameter). The

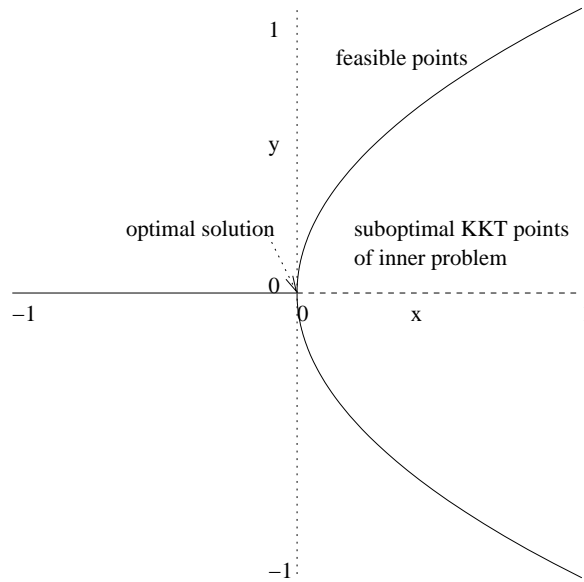


Figure C-5: Minima and suboptimal KKT points for the inner problem of Example C.12.

bilevel program

$$\begin{aligned}
 & \min_{x,y} x - y \\
 & \text{s.t. } y \in \arg \min_z \frac{xz^2}{2} - zx^3 \\
 & \quad x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned} \tag{mb_1.1.06}$$

has the unique optimal solution $x = 0, y = 1$ with an objective value of -1 .

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

Stationarity of the inner objective gives $xy - x^3 = 0$, which describes a bifurcation for x as a function of y , see also Figure C-6. For $-1 \leq x < 0$, we have $y = -1$ as the unique global optimum of the inner problem while for $x = 0$, all y points are trivially

optimal for the inner problem. Finally for $0 < x \leq 1$ we have $y = x^2$ as the unique global minimum of the inner problem. In addition to these minima for $-1 \leq x < 0$ we have $y = 1$ and $y = x^2$ as suboptimal KKT points of the inner problem. At the constrained KKT points, the magnitude of the KKT multipliers can be estimated by $|x(\pm 1) - x^3| < 2$.

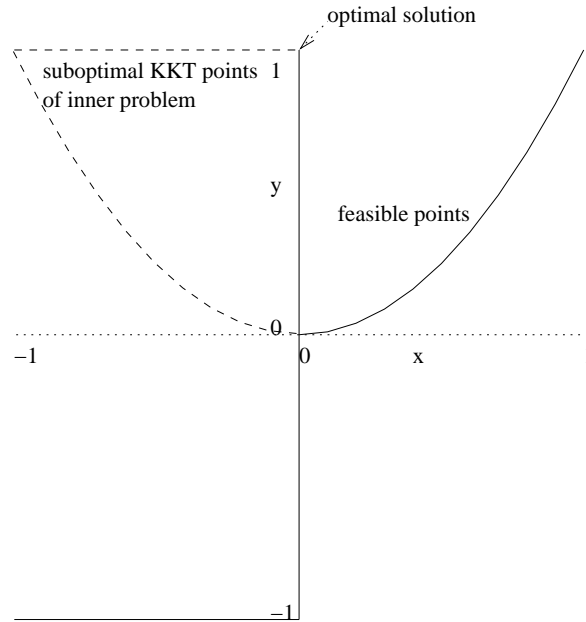


Figure C-6: Minima and suboptimal KKT points for the inner problem of Example C.13.

Example C.14 (Turning point bifurcation). The bilevel program

$$\begin{aligned} \min_{x,y} (x - 1/4)^2 + y^2 \\ \text{s.t. } y \in \arg \min_z z^3/3 - xz \\ x \in [-1, 1], \quad y, z \in [-1, 1] \end{aligned} \quad (\text{mb_1_1_07})$$

has the unique optimal solution $x = 1/4$, $y = 1/2$ with an objective value of $1/4$.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to

nonconvexity of the objective function they are not sufficient for a local/global minimum.

Stationarity of the inner objective gives $y^2 - x = 0$ which describes a turning point bifurcation of y as a function of x , see also Figure C-7. For $-1 \leq x < 0$ it is strictly monotone increasing with y and therefore the unique KKT and minimum is $y = -1$. For $0 \leq x \leq 1$ there are three KKT points $y = -1$, $y = \sqrt{x}$ and $y = -\sqrt{x}$; for $0 \leq x < 1/4$ we have the unique minimum $y = -1$; for $x = 1/4$ we have two minima $y = -1$ and $y = 1/2$; for $1/4 < x \leq 1$ we have the unique minimum $y = \sqrt{x}$.

At the constrained KKT point $y = -1$, we can calculate a bound for the KKT multiplier as $|y^2 - x| = |1 - x|$ and a maximal value of $\mu_{max} = 2$ can be taken.

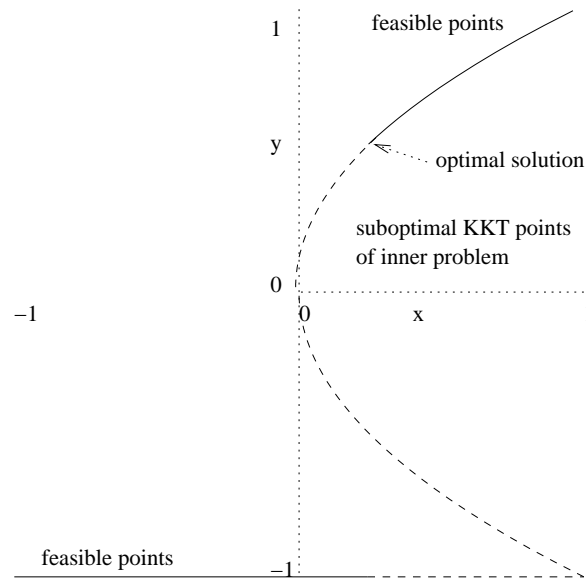


Figure C-7: Minima and suboptimal KKT points for the inner problem of Example C.14.

Example C.15 (Transcritical bifurcation). The bilevel program

$$\begin{aligned} \min_{x,y} x + y \\ \text{s.t. } y \in \arg \min_z xz^2/2 - z^3/3 \\ x \in [-1, 1], \quad y, z \in [-1, 1] \end{aligned} \quad (\text{mb_1_1_08})$$

has the unique optimal solution $x = -1$, $y = 1$ with an objective value of 0.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

Stationarity of the inner objective gives $xy - y^2 = 0$ which describes a transcritical bifurcation of y as a function of x , see also Figure C-8. For all x values $y = 1$, $y = 0$ and $y = x$ are KKT points of the inner problem. Out of them $y = 1$ is optimal for $0 \leq x \leq 2/3$ and $y = 0$ is optimal for $2/3 \leq x \leq 1$.

At the constrained KKT point $y = 1$, we can calculate a bound for the KKT multiplier as $|xy - y^2| = |x - 1|$ and a maximal value of $\mu_{max} = 2$ can be taken.

Example C.16 (Subcritical pitchfork bifurcation). The bilevel program

$$\begin{aligned} \min_{x,y} 2x + y \\ \text{s.t. } y \in \arg \min_z -xz^2/2 - z^4/4 \\ x \in [-1, 1], \quad y, z \in [-1, 1] \end{aligned} \quad (\text{mb_1_1_09})$$

has two optimal solutions $x = -1$, $y = 0$ and $x = -1/2$, $y = -1$ with an objective value of -2 .

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

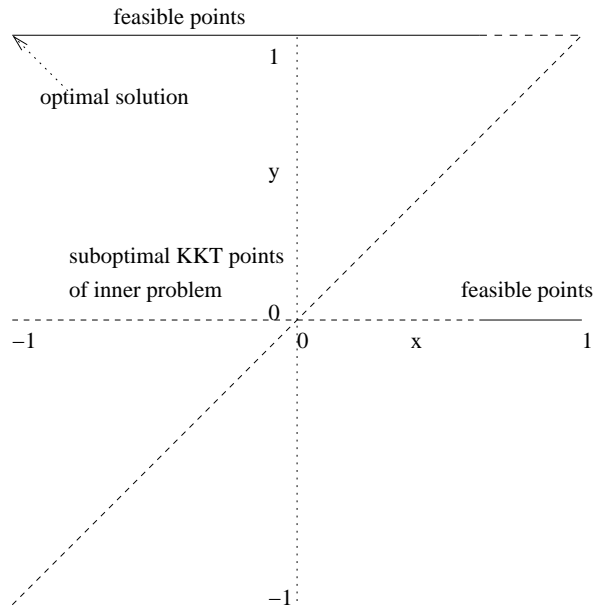


Figure C-8: Minima and suboptimal KKT points for the inner problem of Example C.15.

Stationarity of the inner objective gives $-xy - y^3 = 0$ which describes a subcritical pitchfork bifurcation of y as a function of x . The feasible set for the inner problem is shown in Figure C-9. For $-1 \leq x < -1/2$ the inner problem has the unique minimum $y = 0$ and four suboptimal KKT points $y = \pm\sqrt{-x}$ and $y = \pm 1$. For $x = -1/2$ the inner problem has three minima $y = 0$ and $y = \pm 1$ and two suboptimal KKT points $y = \pm\sqrt{-x}$. For $-1/2 < x \leq 0$ the inner problem has two minima $y = \pm 1$ and three suboptimal KKT points $y = \pm\sqrt{-x}$ and $y = 0$. For $0 < x \leq 1$ the inner problem has two minima $y = \pm 1$ and one suboptimal KKT point $y = 0$.

At the constrained KKT points $y = \pm 1$, we can calculate a bound for the KKT multipliers as $|xy + y^3| = |\pm x \pm 1|$ and a maximal value of $\mu_{max} = 2$ can be taken.

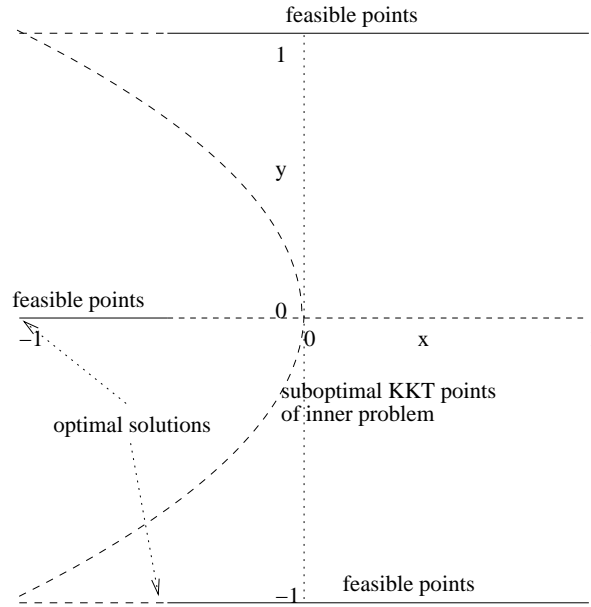


Figure C-9: Minima and suboptimal KKT points for the inner problem of Example C.16.

Example C.17 (Subcritical pitchfork bifurcation). The bilevel program

$$\begin{aligned}
 & \min_{x,y} (x + 1/2)^2 + 1/2 y^2 \\
 & \text{s.t. } y \in \arg \min_z 1/2 x z^2 + 1/4 z^4 \quad (\text{mb_1.1_10}) \\
 & \quad x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned}$$

has two optimal solutions $x = -1/4$, $y = \pm 1/2$ with an objective value of 0.1875.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

Stationarity of the inner objective gives $xy + y^3 = 0$, which describes a subcritical pitchfork bifurcation of y as a function of x . The feasible set for the inner problem is shown in Figure C-10. For $-1 \leq x < 0$ the inner problem has two minima $y = \pm \sqrt{-x}$

mum.

Stationarity of the inner objective gives $xy - y^3 = 0$, which describes a bifurcation for y as a function of x , see also Figure C-11. For $-1 \leq x \leq 0$, we have $y = \pm 1$ as the minima of the inner problem and $y = 0$ as a suboptimal KKT point. For $0 \leq x < 1/2$, we have $y = \pm 1$ as the minima of the inner problem and $y = 0, y = \pm\sqrt{x}$ as suboptimal KKT points. For $x = 1/2$, we have $y = 0$ and $y = \pm 1$ as the minima of the inner problem and $y = \pm\sqrt{1/2}$ as suboptimal KKT points. For $0 < x \leq 1$, we have $y = 0$ as the unique minimum of the inner problem and $y = \pm\sqrt{1/2}, y = \pm 1$ as suboptimal KKT points.

At the constrained KKT points $y = \pm 1$, we can calculate a bound for the KKT multipliers as $|xy - y^3| = |\pm x \mp 1|$ and a maximal value of $\mu_{max} = 1$ can be taken.

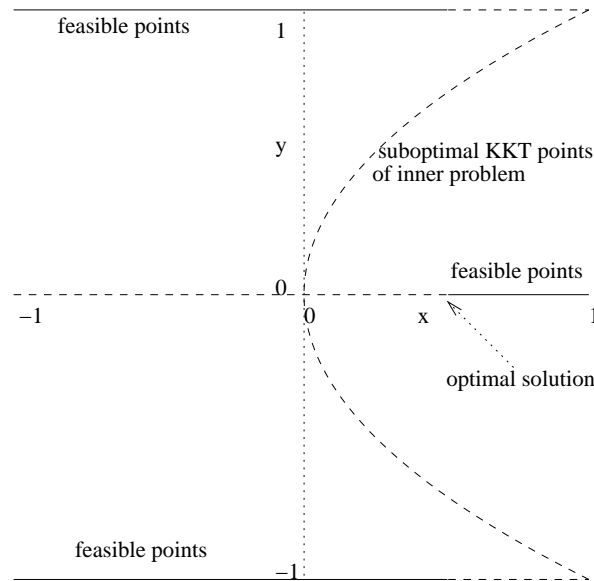


Figure C-11: Minima and suboptimal KKT points for the inner problem of Example C.18.

Example C.19 (Supercritical pitchfork bifurcation with x as the parameter). The

bilevel program

$$\begin{aligned}
 \min_{x,y} \quad & x y - y + y^2/2 \\
 \text{s.t.} \quad & y \in \arg \min_z -x z^2 + z^4/2 \\
 & x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned} \tag{mb_1.1_12}$$

has the unique optimal solution $x = \frac{7-\sqrt{13}}{18} \approx 0.189$, $y = -\sqrt{\frac{7-\sqrt{13}}{18}} \approx -0.768$ with an objective value of $f \approx -0.258$.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum. Stationarity of the inner objective gives $-2xy + 2y^3 = 0$, which describes a bifurcation for y as a function of x , see also Figure C-12. For $-1 \leq x \leq 0$, we have $y = 0$ as the unique global optimum of the inner problem. For $0 < x \leq 1$, $y = \pm\sqrt{x}$ are the global minima for the inner problem while $y = 0$ is a suboptimal KKT. All KKT points are unconstrained, so that any $\mu_{max} > 0$ can be used.

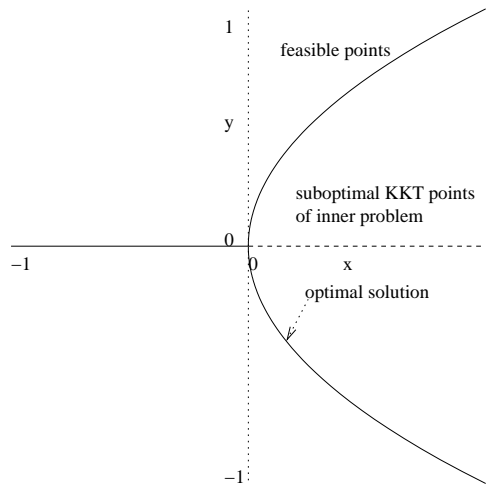


Figure C-12: Minima and suboptimal KKT points for the inner problem of Example C.19.

Example C.20 (Crossing KKT points). The bilevel program

$$\begin{aligned} & \min_{x,y} (x - 1/4)^2 + y^2 \\ & y \in \arg \min_z z^3/3 - x^2 z \\ & x \in [-1, 1], \quad y, z \in [-1, 1] \end{aligned} \tag{mb_1_1_13}$$

has the unique optimal solutions $x = 1/2$, $y = 1/2$ with an optimal objective value of $5/16$.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

The inner objective has the following minima, see also Figure C-13. For $-1 \leq x < -1/2$ the unique global minimum is $y = -x$. For $x = -1/2$ there are two minima $y = -1$ and $y = 1/2$. For $-1/2 < x < 1/2$ the unique global minimum is $y = -1$. For $x = 1/2$ there are two minima $y = -1$ and $y = 1/2$. For $1/2 < x \leq 1$ the unique global minimum is $y = x$.

Stationarity of the inner objective gives $y^2 - x^2 = 0$ and therefore $y = \pm x$ are KKT points for all $x \in [-1, 1]$. At the constrained KKT points, the value of the KKT multipliers can be bounded by $|y^2 - x^2| = |1 - x^2| \leq 1$.

Example C.21 (Switching KKT points). The bilevel program

$$\begin{aligned} & \min_{x,y} (x - 0.6)^2 + y^2 \\ & \text{s.t. } y \in \arg \min_z z^4 + 4/30 (-x + 1) z^3 + (-0.02 x^2 + 0.16 x - 0.4) z^2 \\ & \quad \quad \quad + (0.004 x^3 - 0.036 x^2 + 0.08 x) z \\ & x \in [-1, 1], \quad y, z \in [-1, 1] \end{aligned} \tag{mb_1_1_14}$$

has the unique optimal solution $x \approx -0.5545$, $y \approx 0.4554$ and an optimal objective value of approximately 0.2095.

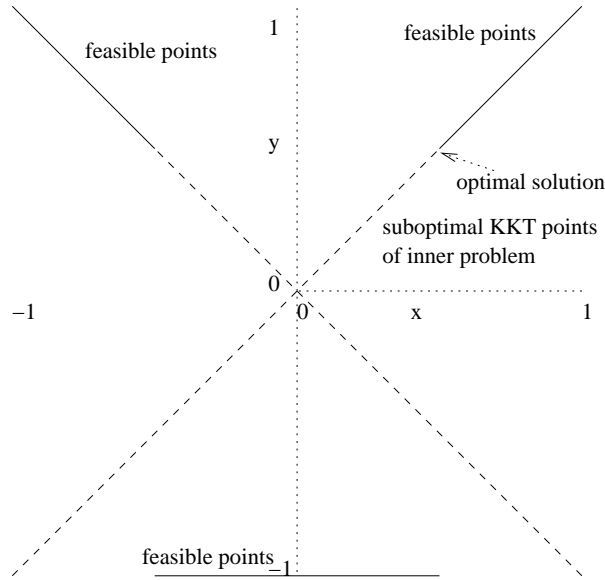


Figure C-13: Minima and suboptimal KKT points for the inner problem of Example C.20.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

The inner objective has the following minima and KKT points, see also Figure C-14. For $-1 \leq x < -1/2$ the unique global minimum is $y = 0.4 - 0.1x$. For $x = -1/2$ there are two minima $y = -0.55$ and $y = 0.45$. For $-1/2 < x \leq 1$ the unique global minimum is $y = -0.5 + 0.1x$. For all $x \in [-1, 1]$ there exist three KKT points, namely $y = -0.5 + 0.1x$, $y = 0.4 - 0.1x$ and $0.1x$. All KKT points are unconstrained so that any $\mu_{max} > 0$ can be used.

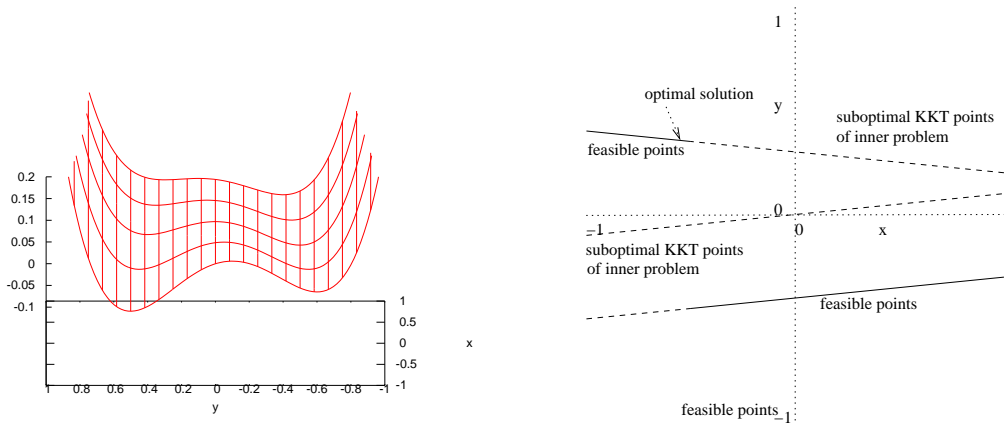


Figure C-14: Inner objective function, its KKT points and its minima for Example C.21.

Example C.22 (Switching KKT points). The bilevel program

$$\begin{aligned}
 & \min_{x,y} (x - 0.6)^2 + y^2 \\
 & \text{s.t. } y \in \arg \min_z z^4 + \frac{4}{30}(-x + 1)z^3 + (-0.02x^2 + 0.16x - 0.4)z^2 \\
 & \quad \quad \quad + (0.004x^3 - 0.036x^2 + 0.08x)z \\
 & \text{s.t. } 0.01(1 + x^2) - z^2 \leq 0 \qquad \qquad \qquad (\text{mb_1.1_15}) \\
 & x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned}$$

has the unique optimal solution $x \approx -0.5545$, $y \approx 0.4554$ with an objective value of approximately 0.2095.

This example build upon Example C.21. The inner program satisfies the linear/concave constraint qualification [41, p. 322] and therefore the KKT conditions are necessary for a local minimum.

The inner objective has the following minima, see also Figure C-15. For $-1 \leq x < -\frac{1}{2}$ the unique global minimum is $y = 0.4 - 0.1x$. For $x = -\frac{1}{2}$ there are two minima $y = -0.55$ and $y = 0.45$. For $-\frac{1}{2} < x \leq 1$ the unique global minimum is $y = -0.5 + 0.1x$. For all $x \in [-1, 1]$ there exist two KKT points, namely $y = -0.5 + 0.1x$

and $y = 0.4 - 0.1x$. All KKT points are unconstrained so that any $\mu_{max} > 0$ can be used.

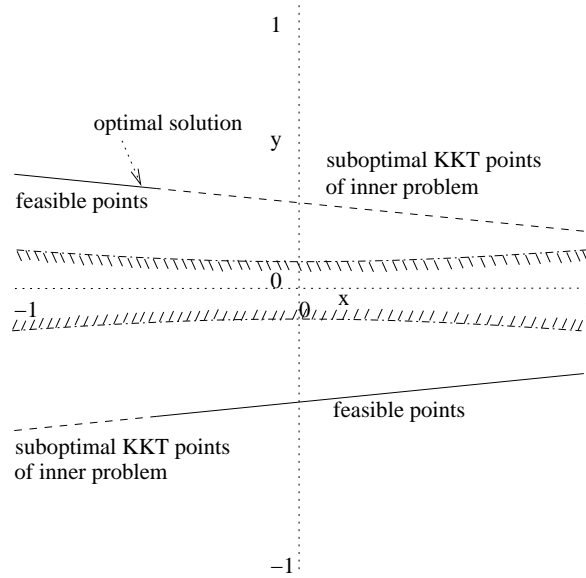


Figure C-15: Minima and suboptimal KKT points for the inner problem of Example C.22.

Example C.23 (No KKT constraint qualification). The bilevel program

$$\begin{aligned}
 & \min_{x,y} x^2 \\
 & \text{s.t. } 1 + x - 9x^2 - y \leq 0 \\
 & \qquad y \in \arg \min_z \qquad \qquad \qquad (\text{mb_1.1_16}) \\
 & \qquad \qquad \qquad \text{s.t. } z_1^2(x - 0.5) \leq 0 \\
 & x \in [-1, 1], \quad y, z \in [-1, 1]
 \end{aligned}$$

has the unique optimal solution $x = \frac{1-\sqrt{73}}{18} \approx -0.4191$, $y_1 = -1$ with an objective value of ≈ 0.1756 .

The inner problem is feasible for all x . For $x \leq 0.5$ any $y \in [-1, 1]$ is feasible and $y = -1$ optimal. For $x > 0.5$ only $y = 0$ is feasible and optimal. Together

with the outer constraint the feasible set of (mb_1.1_16) is therefore $y = -1$ for $x \leq \frac{1-\sqrt{73}}{18} \approx -0.4191$ and $y = 0$ for $x \geq \frac{1+\sqrt{73}}{18} \approx -0.53022$.

For $x > 0.5$ the inner problem has no Slater point, and the unique minimum $y = 0$, does not satisfy the KKT conditions. Therefore the KKT conditions are not necessary for a minimum and the KKT-based lower bound is not applicable.

The purpose of this example is to study programs which satisfy Assumption 4.3, but do not allow any combination of branching and node selection heuristics. For this example, the first iteration gives a lower bound of 0 and no upper bound. If after the first iteration one branches X to $[-1, 1] \cap [0, 1]$ and then considers $[0, 1]$ without branching Algorithm 4.1 would never generate an upper bound. It would tend to $x = \frac{1+\sqrt{73}}{18}$ but never reach it.

Example C.24 (Randomly generated example). The bilevel program

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}} x_1 y_1 + x_2 y_1^2 - x_1 x_2 y_3 \\
& \text{s.t. } 0.1 y_1 y_2 - x_1^2 \leq 0 \\
& \quad x_2 y_1^2 \leq 0 \\
& \quad \mathbf{y} \in \arg \min_{\mathbf{z}} x_1 z_1^2 + x_2 z_2 z_3 \quad (\text{mb_2.3_01}) \\
& \quad \text{s.t. } z_1^2 - z_2 z_3 \leq 0 \\
& \quad \quad z_2^2 z_3 - z_1 x_1 \leq 0 \\
& \quad \quad -z_3^2 + 0.1 \leq 0 \\
& \quad \mathbf{x} \in [-1, 1]^2, \quad \mathbf{y}, \mathbf{z} \in [-1, 1]^3
\end{aligned}$$

has a best known objective value of -1 and one of the points satisfying this is $x_1 = -1$, $x_2 = -1$, $y_1 = -1$, $y_2 = 1$, $y_3 = 1$.

This example was generated randomly and not analyzed yet. Since there are only 2 \mathbf{x} variables we were able to confirm the result with a discretization of 100×100 points.

Example C.25 (Separable example). The bilevel program

$$\begin{aligned}
 & \min_{\mathbf{x}, \mathbf{y}} x_1 y_1 + x_2 y_2^2 + x_1 x_2 y_3^3 \\
 & \text{s.t.} \quad 0.1 - x_1^2 \leq 0 \\
 & \quad 1.5 - y_1^2 - y_2^2 - y_3^2 \leq 0 \quad (\text{mb_2.3.02}) \\
 & \quad 2.5 + y_1^2 + y_2^2 + y_3^2 \leq 0 \\
 & \quad \mathbf{y} \in \arg \min_{\mathbf{z}} x_1 z_1^2 + x_2 z_2^2 + (x_1 - x_2) z_3^2 \\
 & \quad \mathbf{x} \in [-1, 1]^2, \quad \mathbf{y}, \mathbf{z} \in [-1, 1]^3
 \end{aligned}$$

has the unique optimal solution $x_1 = -1$, $x_2 = -1$, $y_1 = 1$, $y_2 = 1$, $y_3 = -\sqrt{0.5} \approx -0.707$ with an objective value of $-2 - \sqrt{0.5}^3 \approx -2.35$.

The purpose of this example is to generate a larger size problem that is easy to analyze. The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] but due to nonconvexity of the objective function they are not sufficient for a local/global minimum.

The inner problem can be decomposed into three programs, one for each variable y_i . For these subproblems

- The optimal value of y_1 depends only on x_1 . Noting the outer level constraint on x_1 we have
 - For $x_1 \geq \sqrt{0.1}$ the unique optimum and KKT is $y_1 = 0$.
 - For $x_1 \leq -\sqrt{0.1}$ there are two minima $y_1 = \pm 1$ and a suboptimal KKT point $y_1 = 0$.
- The optimal value of y_2 depends only on x_2 .
 - For $x_2 > 0$ the unique optimum and KKT is $y_2 = 0$.
 - For $x_2 = 0$ any $y_2 \in [-1, 1]$ is optimal and KKT point.

- For $x_2 < 0$ there are two minima $y_2 = \pm 1$ and a suboptimal KKT point $y_2 = 0$.
- The optimal value of y_3 depends only on $x_1 - x_2$.
 - For $x_1 > x_2$ the unique optimum and KKT is $y_3 = 0$.
 - For $x_1 = x_2$ any $y_3 \in [-1, 1]$ is optimal and KKT point.
 - For $x_1 < x_2$ there are two minima $y_3 = \pm 1$ and a suboptimal KKT point $y_3 = 0$.

Together with the outer problem constraints we obtain four subsets for the feasible set, see also Figure C-16

- $-1 \leq x_1 \leq -\sqrt{0.1}$, $0 < x_2 \leq 1$, $y_1 = \pm 1$, $y_2 = 0$, $y_3 = \pm 1$. In this subset the best possible solution is found for $x_1 = -1$, $x_2 = 1$, $y_1 = 1$, $y_2 = 0$, $y_3 = -1$ with an objective value of -2 .
- $-1 \leq x_1 \leq -\sqrt{0.1}$, $x_2 = 0$, $y_1 = \pm 1$, $y_2 \in [-1, 1]$, $y_3 = \pm 1$. In this subset the best possible solution is found for $x_1 = -1$, $x_2 = 0$, $y_1 = 1$, $y_3 = -1$ with an objective value of -1 .
- $-1 \leq x_1 \leq -\sqrt{0.1}$, $x_2 = x_1$, $y_1 = \pm 1$, $y_2 = \pm 1$, $y_3 \in [-1, 1]$. In this subset the best possible solution is found for $x_1 = -1$, $x_2 = -1$, $y_1 = 1$, $y_2 = 1$, $y_3 = -\sqrt{0.5} \approx -0.707$ with an objective value of $-2 - 0.5^{1.5} \approx -2.35$.
- $-1 \leq x_1 \leq -\sqrt{0.1}$, $-1 \leq x_2 < x_1$, $y_1 = \pm 1$, $y_2 = \pm 1$, $y_3 = 0$. In this subset the best possible solution is found for $x_1 = -1 + \epsilon$, $x_2 = -1$, $y_1 = 1$, $y_2 = 1$, $y_3 = 0$ with an objective value of $-2 + \epsilon$ for arbitrarily small $\epsilon > 0$.

At the constrained KKT points $y_i = \pm 1$, we can calculate a bound for the KKT multipliers as

$$\max\{|2x_1y_1|, |2x_2y_2|, |2(x_1 - x_2)y_3|\} \leq 4$$

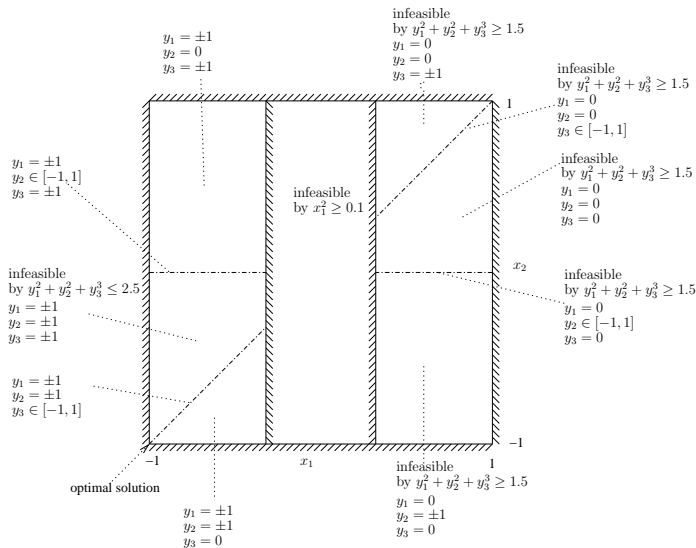


Figure C-16: Feasible set in the x_1, x_2 space for Example C.25.

C.2 Examples from Gümüs and Floudas [133]

Example C.26 (Example 1 in [133]). The bilevel program

$$\begin{aligned}
 & \min_{x,y} 16x^2 + 9y^2 \\
 & \text{s.t. } -4x + y \leq 0 \\
 & \quad y \in \arg \min_{\mathbf{z}} (x + z - 20)^4 \quad (\text{gf.1}) \\
 & \quad \text{s.t. } 4x + z - 50 \leq 0 \\
 & \quad x \in [0, 12.5], \quad y, z \in [0, 50]
 \end{aligned}$$

has the unique optimal solution $x = 45/4 = 11.25$, $y = 5$ with an objective value of 2250. Note that we added explicit upper bounds on the variables that can be inferred by the inner constraint $4x + z - 50 \leq 0$ along with the variable lower bounds of $(x \geq 0, y \geq 0)$.

Note that for $x = 12.5$ no point \tilde{z} exists such that $4 \cdot 12.5 + \tilde{z} < 0$ violating (4.21). But (4.22) holds because for $y = 0$ (the only feasible for $x = 12.5$) $f = 2500$, which is significantly above $f^* = 2250$.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a local minimum. Since the objective function is convex they are also sufficient for a global minimum.

For each fixed x the feasible set for y is $y \in [0, 50 - 4x]$. Since the objective function is strictly convex for z , its minimum can only be attained at the bounds or at the stationary point $20 - x$. A parametric comparison of these three possibilities gives the parametric optimal solution of the inner program as

$$\bar{y}(x) = \begin{cases} 20 - x & \text{if } x \in [0, 10] \\ 50 - 4x & \text{otherwise.} \end{cases}$$

With the outer constraint ($y \leq 4x$) this solution is feasible for $x \geq 4$ and therefore the optimization program is equivalent to minimizing the function

$$\bar{f}(x) = \begin{cases} 25x^2 - 360x + 3600 & \text{if } x \in [4, 10] \\ 160x^2 - 3600x + 22500 & \text{if } x \in (10, 12.5], \end{cases}$$

see also Figure C-17. The bilevel program therefore has a local minimum at $x = 36/5$, $y = 64/5$ with an objective value of 2304 and a global minimum at $x = 45/5$, $y = 5$ with an objective value of 2250.

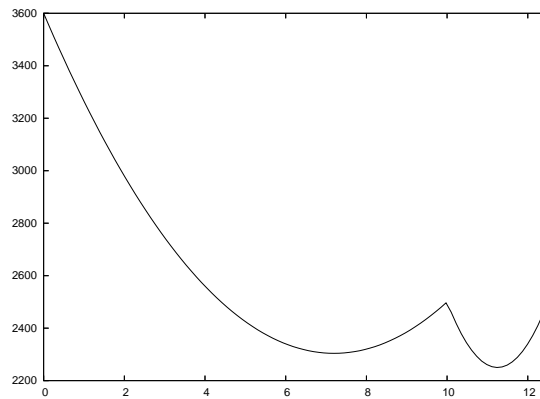


Figure C-17: Equivalent objective function of Example C.26.

Example C.27 (Example 2 in [133]). The bilevel program

$$\begin{aligned}
& \min_{x, \mathbf{y}} x^3 y_1 + y_2 \\
& \text{s.t. } \mathbf{y} \in \arg \min_{\mathbf{z}} -z_2 \\
& \quad \text{s.t. } xz_1 - 10 \leq 0 \\
& \quad \quad z_1^2 + xz_2 - 1 \leq 0 \\
& x \in [0, 1], \quad \mathbf{y}, \mathbf{z} \in [-1, 1] \times [0, 100]
\end{aligned} \tag{gf_2}$$

has the unique optimal solution $x = 1$, $y_1 = 0$, $y_2 = 1$ with an objective value of 1. Note that as formulated in [133] the inner program is unbounded for $x = 0$; in that case replacing the inner program with its KKT conditions is not a valid relaxation. We added an upper bound to the variable y_2 that does not affect the optimal solution of the bilevel program. An alternative would be to consider $x \geq \delta$ for some small $\delta > 0$.

The constraints of the inner program are partially convex in \mathbf{y} and $\mathbf{y} = \mathbf{0}$ is a Slater point for all \mathbf{x} and therefore the KKT conditions are necessary [41, p. 325] for a local minimum. Since the objective function is convex they are also sufficient for a global minimum.

The inner program is a maximization of z_2 and therefore setting $z_1 = 0$ the binding constraint is $xz_2 \leq 1$ which gives the parametric optimal solution $\bar{y}_2(x) = \min(1/x, 100)$. Note that the constraint $xz_1 \leq 10$ is always inactive. As a consequence for $x \geq 1/100$ the inner program gives $y_1 = 0$ while for $x < 1/100$ any $y_1 \in [0, \sqrt{1 - 100x}]$ is feasible. For $x \geq 1/100$ the constraint $z_1^2 + xz_2 \leq 1$ has a KKT multiplier $1/x$ and therefore $\mu_{max} = 100$ can be taken.

Since $x \geq 0$ the outer optimizer will always choose $y_1 = 0$ if given a choice. Therefore the bilevel program is equivalent to the minimization of $\min(1/x, 100)$ and the optimal solution $x = 1$, $y_1 = 0$, $y_2 = 1$ is obtained.

Example C.28 (Example 3 in [133]). The bilevel program

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}} -8x_1 - 4x_2 + 4y_1 - 40y_2 - 4y_3 \\
& \text{s.t. } \mathbf{y} \in \arg \min_{\mathbf{z}} \frac{1 + x_1 + x_2 + 2z_1 - z_2 + z_3}{6 + 2x_1 + z_1 + z_2 - 3z_3} \\
& \quad \text{s.t.} \quad -z_1 + z_2 + z_3 \leq 1 \quad (\text{gf.3}) \\
& \quad \quad 2x_1 - z_1 + 2z_2 - \frac{1}{2}z_3 \leq 1 \\
& \quad \quad 2x_2 + 2z_1 - z_2 - \frac{1}{2}z_3 \leq 1 \\
& \mathbf{x} \in [0, 2]^2, \quad \mathbf{y}, \mathbf{z} \in [0, 2]^3
\end{aligned}$$

has the best known optimal solution $x_1 = 0$, $x_2 = 0.9$, $y_1 = 0$, $y_2 = 0.6$, $y_3 = 0.4$ with an objective value of -29.2 . Note that we have eliminated the slack variables y_4, y_5, y_6 from the original formulation and have introduced explicit upper bounds for the variables that can be inferred by the inner constraints and the lower bounds of the variables ($\mathbf{x} \geq \mathbf{0}$, $\mathbf{y} \geq \mathbf{0}$). Note also that for the KKT conditions we eliminate the denominator of the objective function derivatives and obtain linear stationarity conditions.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a local minimum. The inner program is a linear fractional programming problem [39, p. 524], parameterized in \mathbf{x} . It can be easily verified, e.g., through the solution of a linear program with (\mathbf{x} and \mathbf{y} as variables) that the denominator ($6 + 2x_1 + z_1 + z_2 - 3z_3$) of the inner objective function can vary only between 1 and 8. Since the denominator is positive, the objective function is pseudo-concave and the KKT conditions are necessary for a global minimum [39, p. 525].

Since the denominator is positive, one can replace the inner program with an equivalent linear program, [39, 66, p. 529]. The equivalent LP can be replaced by its equivalent KKT conditions. This method would probably introduce nonlinearity in the outer objective.

We have not yet analyzed the behavior of this example. We have confirmed the

solution by a discretization in the \mathbf{x} space with 100×100 points.

Example C.29 (Example 4 in [133]). The bilevel program

$$\begin{aligned}
 & \min_{x,y} (x - 3)^2 + (y - 2)^2 \\
 & \text{s.t. } -2x + y - 1 \leq 0 \\
 & \quad x - 2y + 2 \leq 0 \\
 & \quad x + 2y - 14 \leq 0 \\
 & \quad y \in \arg \min_z (z - 5)^2 \\
 & \quad x \in [0, 8], \quad y, z \in [0, 10]
 \end{aligned} \tag{gf.4}$$

has the unique optimal solution $x = 3, y = 5$ with an objective value of 9. Note that we chose arbitrary bounds for y ; since the minimum of the inner program is obtained for $y = 5$ they do not affect the inner program. They also do not affect the outer program since the outer constraints directly give $y \in [0, 6]$.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a local minimum. Since the objective function is convex they are also sufficient for a global minimum.

The inner program is not parameterized in x and we obtain directly $y = 5$ from the inner program. Since the KKT points are all unconstrained any $\mu_{max} > 0$ can be used. The bilevel program is therefore equivalent to

$$\begin{aligned}
 & \min_{x \in [0,8]} (x - 3)^2 + 9 \\
 & \text{s.t. } -2x + 4 \leq 0 \\
 & \quad x - 8 \leq 0 \\
 & \quad x - 4 \leq 0
 \end{aligned}$$

or

$$\min_{x \in [2,4]} (x - 3)^2 + 9,$$

which gives $x = 3$. Note that $x = 3$ corresponds to the unconstrained minimum of the outer objective function.

Example C.30 (Example 5 in [133]). The bilevel program

$$\begin{aligned} & \min_{x, \mathbf{y}} x \\ & \text{s.t. } y \in \arg \min_{\mathbf{z}} -z_1 + 0.5864z_1^{0.67} \\ & \quad \text{s.t. } \frac{0.0332333}{z_2} + 0.1z_1 - 1 \leq 0 \quad (\text{gf.5}) \\ & \quad 4\frac{x}{z_2} + 2\frac{x^{-0.71}}{z_2} + 0.0332333x^{-1.3} - 1 \leq 0 \\ & \quad x \in [0.1, 10], \quad \mathbf{y}, \mathbf{z} \in [0.1, 10]^2 \end{aligned}$$

has the unique optimal solution $x = 0.193616$, $y_1 = 9.966766700$, $y_2 = 10$ with an objective value of 0.193616. Note that this is different than the solution reported in [133]. This may be due to typos of the formulated program, but we could not find the example in the original reference [111] to confirm the formulation. Note also that we directly solve the problem as opposed to reformulating it using exponential functions.

The inner variables do not affect the outer problem. In the inner program the objective function depends only on y_1 . The outer variable x appears only in the second constraint and setting $y_2 = 10$ will allow most freedom for the variable x giving approximately $x \in [0.1936160966, 2.182605852]$. The optimal x is therefore $x = 0.1936160966$. If the outer optimizer chooses this x , the second constraint directly gives $y_2 = 10$. The first constraint then gives $y_1 \in [0.1, 9.966766700]$. Since the inner objective is monotone decreasing, we obtain $y_1 = 9.966766700$. Note also that for $x = 0.1936160966$ the second constraint cannot be satisfied with strict inequality and $f(0.1936160966) = f^*$ and therefore this example violates Assumption 4.3 for the optimal \mathbf{x} . Solving the upper bounding problem at the root node gives the same value as the lower bounding problem and this slight modification of the algorithm

converges at the root node.

The inner program is a geometric program for each x (x affects merely the coefficient of the posynomials); therefore the reformulation performed by Gümüs and Floudas [133] leads to a convex program [39, p. 531]. For the reformulated program for $x \in (0.1936160966, 2.182605852)$ the inner program satisfies the constraint qualifications by the existence of Slater points. For $x \in \{0.1936160966, 2.182605852\}$ a constraint qualification needs to be shown. In either case, the second constraint gives directly $y_2 = 10$ and no Slater point exists; the second constraint is linearly dependent of the constraint $y_2 \leq 10$ which is also active. Neither of these two constraints depend on y_1 and the objective function does not depend on y_2 ; since the derivative of the first constraint with respect to y_2 is negative, it is possible to find KKT points at the minimum (they are not unique because of the linear dependence).

Note also that for bilevel geometric programs, an algorithm has been proposed by Segall [249, 250].

An alternative reformulation of the inner program is to note that z_2 appears only in the inverse functions and therefore $z_2' = 1/z_2$ can be used instead (with the same bounds). The resulting program has linear constraints, and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a local minimum.

Multiplying the constraints by z_2 is not advisable because it would create a bilinear term $z_1 z_2$ in the first constraint.

Constraint qualification for the original program remains to be shown.

Note that if we inverse the outer optimization sign we will get $x = 2.182605852$ but the same \mathbf{y} .

C.3 Examples from Sahin and Ciric [246]

Example C.31 (Example 1 in [246]). The bilevel program

$$\begin{aligned}
 & \min_{x, \mathbf{y}} -x - 3y_1 + 2y_2 \\
 & \text{s.t. } y \in \arg \min_{\mathbf{z}} -z_1 \\
 & \quad \text{s.t. } -2x + z_1 + 4z_2 - 16 \leq 0 \quad (\text{sc.1}) \\
 & \quad \quad 8x + 3z_1 - 2z_2 - 48 \leq 0 \\
 & \quad \quad -2x + z_1 - 3z_2 + 12 \leq 0 \\
 & x \in [0, 8], \quad \mathbf{y}, \mathbf{z} \in [0, 4] \times [0, 6]
 \end{aligned}$$

has an objective value of -13 attained at $x = 5$, $y_1 = 4$, $y_2 = 2$. Note that we reformulated the problem to minimization. Note also that we added explicit bounds for y_2 which can be inferred by the constraints of the inner program.

The inner program is linear and therefore the KKT conditions are both necessary and sufficient for a minimum.

The inner program can be solved parametrically as a function of x . The solution obtained is

$$\bar{y}_1(x) = \begin{cases} 2x & \text{if } x \in [0, 2] \\ 4 & \text{if } x \in (2, 6] \\ 16 - 2x & \text{if } x \in (6, 8]. \end{cases}$$

We can now obtain the optimal solution by solving the LP

$$\begin{aligned}
 & \min_{x, y_2} -x - 3\bar{y}_1(x) + 2y_2 \\
 & -2x + \bar{y}_1(x) + 4y_2 - 16 \leq 0 \\
 & 8x + 3\bar{y}_1(x) - 2y_2 - 48 \leq 0 \\
 & -2x + \bar{y}_1(x) - 3y_2 + 12 \leq 0 \\
 & x \in X_i \quad y_2 \in [0, 6]
 \end{aligned}$$

in the three intervals X_i . For $x \in [0, 2]$ we obtain an optimal objective value of -6 . For $x \in [2, 6]$ we obtain an optimal objective value of -13 attained at $x = 5$, $y_1 = 4$, $y_2 = 2$. For $x \in [6, 8]$ we obtain an optimal objective value of -6 .

Example C.32 (Example 2 in [246]). The bilevel program

$$\begin{aligned}
 & \min_{x,y} (x-3)^2 + (y-2)^2 \\
 & \text{s.t. } y \in \arg \min_z (z-5)^2 \\
 & \quad \text{s.t. } -2x + z - 1 \leq 0 \tag{sc_2} \\
 & \quad \quad x - 2z + 2 \leq 0 \\
 & \quad \quad x + 2z - 14 \leq 0 \\
 & \quad x \in [0, 8], \quad y \in [0, 6]
 \end{aligned}$$

has an optimal objective of 5, attained at $x = 1$, $y = 3$. Note also that we added explicit bounds for y which can be inferred by the constraints of the inner program.

The inner program has linear inequality constraints and therefore by the Adabie constraint qualification the KKT conditions are necessary [39, p. 187] for a local minimum. Since the objective function is convex they are also sufficient for a global minimum. The inner program can be easily solved parametrically. Figure C-18 shows the geometry. The parametric solution obtained is

$$\bar{y}(x) = \begin{cases} 1 + 2x & \text{if } x \in [0, 2] \\ 5 & \text{if } x \in (2, 4] \\ 7 - x/2 & \text{if } x \in (4, 6]. \end{cases}$$

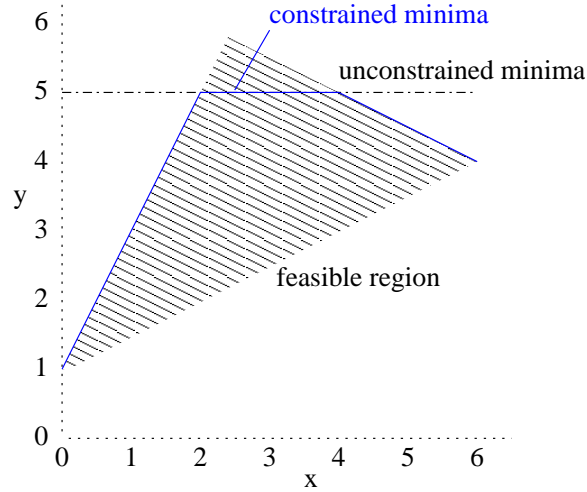


Figure C-18: Inner program of Example C.32.

Therefore the bilevel program is equivalent to a minimization of

$$\bar{f}(x) = \begin{cases} 5x^2 - 10x + 10 & \text{if } x \in [0, 2] \\ x^2 - 6x + 18 & \text{if } x \in (2, 4] \\ 5/4x^2 - 11x + 34 & \text{if } x \in (4, 6] \end{cases}$$

which has the unique global minimum $x = 1, y = 3$ with an objective value of 5. It also has two suboptimal local minima, for $x = 3, y = 2$ with an objective value of 9 and $x = 4.4, y = 4.8$ and an objective value 9.8, see also Figure C-19.

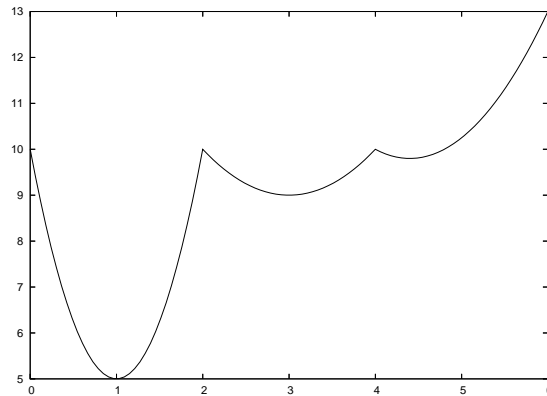


Figure C-19: Equivalent objective function for Example C.32.

Bibliography

- [1] <http://web.mit.edu/cheme/people/faculty/jensen.html>.
- [2] <http://www.uiuc.edu/~r-masel/fuelcell/>.
- [3] http://www.space.com/business/technology/technology/onthededge_0306.html.
- [4] <http://www.comsol.com/>.
- [5] <http://hsl.rl.ac.uk/archive/hslarchive.html>.
- [6] GNU multiple precision arithmetic library. <http://swox.com/gmp/>.
- [7] ILOG CPLEX 9.0 reference manual. <http://www.ilog.com/products/cplex/>.
- [8] One laptop per child <http://laptop.org/>.
- [9] Fuel cell handbook (fifth edition). EG&G Services; Parsons, Inc. Science Applications International Corporation, 2000.
- [10] Joaquin Acevedo and Efstratios N. Pistikopoulos. A parametric MINLP algorithm for process synthesis problems under uncertainty. *Industrial and Engineering Chemistry Research*, 35(1):147–158, 1996.
- [11] Joaquin Acevedo and Efstratios N. Pistikopoulos. A multiparametric programming approach for linear process engineering problems under uncertainty. *Industrial and Engineering Chemistry Research*, 36(3):717–728, 1997.
- [12] Joaquin Acevedo and Efstratios N. Pistikopoulos. An algorithm for multiparametric mixed-integer linear programming problems. *Operations Research Letters*, 24(3):139–148, 1999.

- [13] E. Achenbach. Three-dimensional and time-dependent simulation of a planar solid oxide fuel cell. *Journal of Power Sources*, 49(1-3):333–348, 1994.
- [14] Claire S. Adjiman, Ioannis P. Androulakis, and Christodoulos A. Floudas. Global optimization of mixed-integer nonlinear problems. *AIChE Journal*, 46(9):1769–1797, 2000.
- [15] Claire S. Adjiman and Christodoulos A. Floudas. Rigorous convex underestimators for general twice-differentiable problems. *Journal of Global Optimization*, 9(1):23–40, 1996.
- [16] Patricia Aguiar, Claire Adjiman, and Nigel P. Brandon. Anode-supported intermediate-temperature direct internal reforming solid oxid fuel cell I: Model-based steady-state performance. *Journal of Power Sources*, 138(1-2):120–136, 2004.
- [17] Berit S. Ahmad and Paul I. Barton. Process-wide integration of solvent mixtures. *Computers & Chemical Engineering*, 23(10):1365–1380, 1999.
- [18] Sameer K. Ajmera, Cyrill Delattre, Martin A. Schmidt, and Klavs F. Jensen. Microfabricated differential reactor for heterogeneous gas phase catalyst testing. *Journal of Catalysis*, 209(2):401–412, 2002.
- [19] Götz Alefeld and Günter Mayer. Interval analysis: Theory and applications. *Journal of Computational and Applied Mathematics*, 121(1-2):421–464, 2000.
- [20] Khaled Ali Alfadhel and Mayuresh V. Kothare. Microfluidic modeling and simulation of flow in membrane microreactors. *Chemical Engineering Science*, 60(11):2911–2926, 2005.
- [21] Khaled Ali Alfadhel and Mayuresh V. Kothare. Modeling of multicomponent concentration profiles in membrane microreactors. *Industrial & Engineering Chemistry Research*, 44(26):9794–9804, 2005.

- [22] Eugene L. Allgower and Kurt Georg. *Numerical Continuation Methods: An Introduction*. Springer-Verlag, New York, 1980.
- [23] Mahyar A. Amouzegar. A global optimization method for nonlinear bilevel programs. *IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics*, 29(6):771–777, 1999.
- [24] Leonel R. Arana. *High-Temperature Microfluidic Systems for Thermally-Efficient Fuel Processing*. PhD thesis, Massachusetts Institute of Technology, 2003.
- [25] Leonel R. Arana, Chelsey D. Baertsch, R. C. Schmidt, Martin A. Schmidt, and Klavs F. Jensen. Combustion - assisted hydrogen production in a high-temperature chemical reactor/heat exchanger for portable fuel cell applications. In *12th International Conference on Solid-State Sensors, Actuators, and Microsystems (Transducers 03)*, Boston, MA, 2003.
- [26] Leonel R. Arana, Samuel B. Schaevitz, Aleksander J. Franz, and Klavs F. Jensen. A microfabricated suspended-tube chemical reactor for fuel processing. In *Proceedings of the 15th IEEE International Conference on Micro ElectroMechanical Systems, IEEE, New York*, pages 212–215, 2002.
- [27] Leonel R. Arana, Samuel B. Schaevitz, Aleksander J. Franz, Martin A. Schmidt, and Klavs F. Jensen. A microfabricated suspended-tube chemical reactor for thermally efficient fuel processing. *Journal of Microelectromechanical Systems*, 12(5):600–612, 2003.
- [28] Leonel R. Arana, Martin A. Schmidt, and Klavs F. Jensen. Catalytic hydrogen production in a microfabricated chemical reactor. In *AICHE Annual Meeting, Indianapolis IN*, 2002.
- [29] Uri M. Asher and Linda R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia, 1998.

- [30] Simon Atkinson. The implications of scaling down fuel cells. *Membrane Technology*, 2002(1):10–12, 2002.
- [31] Chelsey D. Baertsch, Klavs F. Jensen, Joshua L. Hertz, Harry L. Tuller, Srikar T. Vengallatore, S. Mark Spearing, and Martin A. Schmidt. Fabrication and structural characterization of self-supporting electrolyte membranes for a micro solid-oxide fuel cell. *Journal of Materials Research*, 19(9):2604–2615, 2004.
- [32] Miguel J. Bagajewicz, Robert Pham, and Vasilios Manousiouthakis. On the state space approach to mass/heat exchanger network design. *Chemical Engineering Science*, 53(14):2595–2621, 1998.
- [33] Egon Balas and Robert Jeroslow. Canonical cuts on the unit hypercube. *SIAM Journal of Applied Mathematics*, 23(1):61–79, 1972.
- [34] Ipsita Banerjee and Marianthi G. Ierapetritou. Parametric process synthesis for general nonlinear models. *Computers and Chemical Engineering*, 27(10):1499–1512, 2003.
- [35] B. Bank, J. Guddat, D. Klatte, B. Kummer, and K. Tammer. *Non-Linear Parametric Optimization*. Birkhäuser Verlag, Stuttgart, 1983.
- [36] Jonathan F. Bard. An algorithm for solving the general bilevel programming problem. *Mathematics of Operations Research*, 8(2):260–272, 1983.
- [37] Jonathan F. Bard. *Practical Bilevel Optimization: Algorithms and Applications*. Nonconvex Optimization and Its Applications. Kluwer Academic Publishers, Dordrecht, 1998.
- [38] Paul I. Barton, Alexander Mitsos, and Benoît Chachuat. Optimal start-up of micro power generation processes. In C. Puigjaner and A. Espuña, editors, *Computer Aided Chemical Engineering*, volume 20B, pages 1093–1098. Elsevier, ESCAPE 15, Barcelona, Spain, 29th May–1st June, 2005.

- [39] Mokhtar S. Bazaraa, Hanif D. Sherali, and C. M. Shetty. *Nonlinear Programming*. John Wiley & Sons, New York, Second edition, 1993.
- [40] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Model predictive control based on linear programming – The explicit solution. *IEEE Transactions on Automatic Control*, 47(12), 2002.
- [41] Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Massachusetts, Second edition, 1999.
- [42] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Belmont, Massachusetts, 1997.
- [43] R. Besser and M. Prevot. Linear scale-up of micro-reaction systems. In *IMRET 4, Atlanta*, pages 278–281, 2000.
- [44] Binita Bhattacharjee, William H. Green Jr., and Paul I. Barton. Interval methods for semi-infinite programs. *Computational Optimization and Applications*, 30(1):63–93, 2005.
- [45] Binita Bhattacharjee, Panayiotis Lemonidis, William H. Green Jr., and Paul I. Barton. Global solution of semi-infinite programs. *Mathematical Programming, Series B*, 103(2):283–307, 2005.
- [46] Wayne F. Bialas and Mark H. Karwan. Two-level linear-programming. *Management Science*, 30(8):1004–1020, 1984.
- [47] Lorenz T. Biegler, Ignacio E. Grossmann, and Arthur W. Westerberg. *Systematic Methods of Chemical Process Design*. Prentice Hall, New Jersey, 1997.
- [48] J. W. Blankenship and J. E. Falk. Infinitely constrained optimization problems. *Journal of Optimization Theory and Applications*, 19(2):261–281, 1976.
- [49] L. G. Bleris, J. Garcia, M. V. Kothare, and M. G. Arnold. Towards embedded model predictive control for system-on-a-chip applications. *Journal of Process Control*, 16(3):255–264, 2006.

- [50] Leonidas G. Bleris and Mayuresh V. Kothare. Reduced order distributed boundary control of thermal transients in microsystems. *IEEE Transactions on Control Systems Technology*, 13(6):853–867, 2005.
- [51] E. Bodewig. *Matrix Calculus*. North Holland Publ. Co, 2nd edition, 1959.
- [52] R. D. Braatz, R. C. Alkire, E. Seebauer, E. Rusli, R. Gunawan, T. O. Drews, X. Li, and Y. He. Perspectives on the design and control of multiscale systems. *Journal of Process Control*, 16(3):193–204, 2006.
- [53] Jerome Bracken and James T. McGill. Mathematical programs with optimization problems in constraints. *Operations Research*, 21(1):37–44, 1973.
- [54] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Classics in Applied Mathematics. SIAM, Philadelphia, 1996.
- [55] Ralph J. Brodd. Recent developments in batteries for portable consumer applications. *The Electrochemistry Society Interface*, 8(3):20–23, 1999.
- [56] Anthony Brooke, David Kendrick, Alexander Meeraus, and Ramesh Raman. GAMS: A user’s guide. <http://www.gams.com/>, 2004.
- [57] Darten Browning, Peter Jones, and Ken Packer. An investigation of hydrogen storage methods for fuel cell operation with man-portable equipment. *Journal of Power Sources*, 65(1-2):187–195, 1997.
- [58] Valerie Browning.
<http://www.darpa.mil/dso/thrust/matdev/palmpower/index.html>.
- [59] Chunshe Cao, Yong Wang, Jamie D. Holladay, Evan O. Jones, and Daniel R. Palo. Design of micro-scale fuel processors assisted by numerical modeling. *AIChE Journal*, 51(3):982–988, 2005.
- [60] Suk-Won Cha, Ryan O’ Hayre, and Fritz B. Prinz. The influence of size scale on the performance of fuel cells. *Solid State Ionics*, 175(1-4):789–795, 2004.

- [61] Benoît Chachuat. *libMINLP.a Version 1.1 Manual*. MIT.
- [62] Benoît Chachuat, Alexander Mitsos, and Paul I. Barton. Optimal design and transient operation of micro power generation employing fuel cells. *In preparation*.
- [63] Benoît Chachuat, Alexander Mitsos, and Paul I. Barton. Optimal operation and design of micro power generation processes. Austin, TX, 7th–12th November, 2004. AICHE Annual Meeting.
- [64] Benoît Chachuat, Alexander Mitsos, and Paul I. Barton. Optimal design and steady-state operation of micro power generation employing fuel cells. *Chemical Engineering Science*, 60(16):4535–4556, 2005.
- [65] Benoît Chachuat, Alexander Mitsos, and Paul I. Barton. Optimal start-up of micro power generation processes employing fuel cells. Cincinnati, OH, 30th October–4th November, 2005. AICHE Annual Meeting.
- [66] A. Charnes and W. W. Cooper. Programming with linear fractionals. *Naval Research Logistics Quarterly*, 9:181–186, 1962.
- [67] Zhongxiang Chen, Yibin Yan, and Said S. E. H. Elnashaie. Modeling and optimization of a novel membrane reformer for higher hydrocarbons. *AIChE Journal*, 49(5):1250–1265, 2003.
- [68] Peter A. Clark and Arthur W. Westerberg. Optimization for design problems having more than one objective. *Computers & Chemical Engineering*, 7(4):259–278, 1983.
- [69] Peter A. Clark and Arthur W. Westerberg. Bilevel programming for steady-state chemical process design. 1. Fundamentals and algorithms. *Computers & Chemical Engineering*, 14(1):87–97, 1990.
- [70] Joel S. Cohen. *Computer Algebra and Symbolic Computation. Elementary Algorithms*. A. K. Peters, Natick, 2003.

- [71] Joel S. Cohen. *Computer Algebra and Symbolic Computation: Mathematical Methods*. A. K. Peters, Natick, 2003.
- [72] J. M. Commenge, L. Falk, J. P. Corriou, and M. Matlosz. Optimal design for flow uniformity in microchannel reactors. In *IMRET 4, Atlanta*, pages 23–30, 2000.
- [73] J. M. Commenge, L. Falk, J. P. Corriou, and M. Matlosz. Optimal design for flow uniformity in microchannel reactors. *AIChE Journal*, 48(2):345–358, 2002.
- [74] National Research Council Committee of Soldier Power/Energy Systems. Meeting the energy needs of future warriors. <http://www.nap.edu/catalog/11065.html>.
- [75] W. Grover Coors. Protonic ceramic fuel cells for high-efficiency operation with methane. *Journal of Power Sources*, 118(1-2):150–156, 2003.
- [76] Paola Costamagna, Azra Selimovicb, Marco Del Borghi, and Gerry Agnew. Electrochemical model of the integrated planar solid oxide fuel cell (IP-SOFC). *Chemical Engineering Journal*, 102(1):61–69, 2004.
- [77] T. J. Coutts. A review of progress in thermophotovoltaic generation of electricity. *Renewable & Sustainable Energy Reviews*, 3(2):77–184, 1999.
- [78] Alejandro Crema. A procedure to verify the completeness of the right-hand-side parametric analysis for a mixed integer linear programming problem. *European Journal of Operational Research*, 108(3):684–695, 1998.
- [79] Edward L. Cussler and James Wei. Chemical product engineering. *AIChE Journal*, 49(5):1072–1075, 2003.
- [80] Rita D’Aquino. Perpetually powering wireless sensors. *Chemical Engineering Practice*, pages 7–12, October 2004.

- [81] J. H. Davenport, Y. Siret, and E. Tournier. *Computer algebra : Systems and algorithms for algebraic computation*. Academic Press, London, second edition, 1993.
- [82] William M. Deen. *Analysis of Transport Phenomena*. Oxford University Press, New York, 1998.
- [83] G. A. Deluga, J. R. Salge, L. D. Schmidt, and X. E. Verykios. Renewable hydrogen from ethanol by autothermal reforming. *Science*, 303(5660):993–997, 2004.
- [84] Stephan Dempe. First-order necessary optimality conditions for general bilevel programming problems. *Journal of Optimization Theory and Applications*, 95(3):735–739, 1997.
- [85] Stephan Dempe. *Foundations of Bilevel Programming*. Nonconvex Optimization and its Applications. Kluwer Academic Publishers, Dordrecht, 2002.
- [86] Stephan Dempe. Annotated bibliography on bilevel programming and mathematical programs with equilibrium constraints. *Optimization*, 52(3):333–359, 2003.
- [87] S. R. Deshmukh, A. B. Mhadeshwar, and D. G. Vlachos. Microreactor modeling for hydrogen production from ammonia decomposition on ruthenium. *Industrial and Engineering Chemistry Research*, 43(12):2986–2999, 2004.
- [88] S. R. Deshmukh and D. G. Vlachos. CFD simulations of coupled, countercurrent combustor/reformer microdevices for hydrogen production. *Industrial & Engineering Chemistry Research*, 44(14):4982–4992, 2005.
- [89] S. R. Deshmukh and D. G. Vlachos. Effect of flow configuration on the operation of coupled combustor/reformer microdevices for hydrogen production. *Chemical Engineering Science*, 60(21):5718–5728, 2005.

- [90] S. R. Deshmukh and D. G. Vlachos. Novel micromixers driven by flow instabilities: Application to post-reactors. *AIChE Journal*, 51(12):3193–3204, 2005.
- [91] Werner Dinkelbach, editor. *Sensitivitätsanalysen und parametrische Programmierung*. Econometrics and Operations Research XII. Springer, Heidelberg, 1969.
- [92] James M. Douglas. *Conceptual Design of Chemical Processes*. McGraw-Hill, 1988.
- [93] Vivek Dua, Nikolaos Bozinis, and Efstratios N. Pistikopoulos. A multiparametric programming approach for mixed-integer quadratic engineering problems. *Computers & Chemical Engineering*, 26(4-6):715–733, 2002.
- [94] Vivek Dua, Katerina P. Papalexandri, and E. N. Pistikopoulos. Global optimization issues in multiparametric continuous and mixed-integer optimization problems. *Journal of Global Optimization*, 30(1):59–89, 2004.
- [95] Vivek Dua and Efstratios N. Pistikopoulos. Algorithms for the solution of multiparametric mixed-integer nonlinear optimization problems. *Industrial and Engineering Chemistry Research*, 38(10):3976–3987, 1999.
- [96] C. K. Dyer. A novel thin-film electrochemical device for energy conversion. *Nature*, 343(6258):547–548, 1990.
- [97] C. K. Dyer. Fuel cells for portable applications. *Journal of Power Sources*, 106(1-2):31–34, 2002.
- [98] M. F. Easton, A. G. Mitchell, and W. F. K. Wynne-Jones. The behaviour of mixtures of hydrogen peroxide and water. *Transactions of the Faraday Society*, 48:796–801, 1952.
- [99] Thomas A. Edmunds and Jonathan F. Bard. Algorithms for nonlinear bilevel mathematical programs. *IEEE Transactions on Systems, Man and Cybernetics*, 21(1):83–89, 1991.

- [100] Wolfgang Ehrfeld, Volker Hessel, and Verena Haverkamp. *Ullmann's Encyclopedia of Industrial Chemistry*, chapter Microreactors. Wiley-VCH on line, 1999.
- [101] Mark J. Eisner and Dennis G. Severance. Mathematical techniques for efficient record segmentation in large shared databases. *Journal of the Association for Computing Machinery*, 23(4):619–635, 1976.
- [102] David Eppstein. Setting parameters by example. *SIAM Journal on Computing*, 32(3):643–653, 2003.
- [103] A. H. Epstein and S. D. Senturia. Macro power from micro machinery. *Science*, 276(5316):1211–1211, 1997.
- [104] J. E. Falk and J. M. Liu. On bilevel programming, part I: General nonlinear cases. *Mathematical Programming*, 70(1):47–72, 1995.
- [105] James E. Falk and Karla Hoffman. A nonconvex max-min problem. *Naval Research Logistics*, 24(3):441–450, 1977.
- [106] Mohammad Farooque and Hans C. Maru. Fuel cells - the clean and efficient power generators. *Proceedings of the IEEE*, 89(12):1819–1829, 2001.
- [107] Martin Feinberg and Diane Hildebrandt. Optimal reactor design from a geometric viewpoint: 1. Universal properties of the attainable region. *Chemical Engineering Science*, 52(10):1637–1665, 1997.
- [108] David Fernández-Baca. Space-sweep algorithms for parametric optimization. *Lecture Notes in Computer Science*, 447:264–275, 1990.
- [109] David Fernández-Baca and S. Srinivasan. Constructing the minimization diagram of a two-parameter problem. *Operations Research Letters*, 10(2):87–93, 1991.
- [110] Anthony V. Fiacco. *Introduction to Sensitivity and Stability Analysis in Non-linear Programming*. Mathematics in Science and Engineering. Academic Press, 1983.

- [111] Christodoulos A. Floudas, Panos M. Pardalos, Claire S. Adjiman, William R. Esposito, Zeynep H. Gümüs, Stephen T. Harding, John L. Klepeis, Clifford A. Meyer, and Carl A. Schweiger. *Handbook of Test Problems in Local and Global Optimization*. Nonconvex Optimization and Its Applications. Kluwer Academic Publishers, Dordrecht, 1999.
- [112] Woodie Flowers, John McBean, and Kailas Narendran.
<http://www.mit.edu/afs/athena/org/a/activejointbrace/index.shtml>.
- [113] José Fortuny-Amat and Bruce McCarl. A representation and economic interpretation of a two-level programming problem. *Journal of the Operational Research Society*, 32(9):783–792, 1981.
- [114] Aleksander J. Franz, Klavs F. Jensen, and Martin A. Schmidt. Palladium membrane microreactors. In *IMRET 3, Frankfurt, Germany*, pages 267–276, 1999.
- [115] Robert M. Freund. Postoptimal analysis of a linear program under simultaneous changes in matrix coefficients. *Mathematical Programming Study*, 24:1–13, 1985.
- [116] Frost and Sullivan. Micro fuel cells go big. *Technology Review*, May 2005.
- [117] Tomas Gal. *Postoptimal Analyses, Parametric Programming and Related Topics*. deGruyter, Berlin, 2nd edition, 1995.
- [118] Jason C. Ganley, E. G. Seebauer, and Richard I. Masel. Porous anodic alumina microreactors for production of hydrogen from ammonia. *AIChE Journal*, 50(4):829–834, 2004.
- [119] A. M. Geoffrion and R. Nauss. Parametric and postoptimality analysis in integer linear programming. *Management Science*, 23(5):453–466, 1977.
- [120] Thomas Gervais and Klavs F. Jensen. Mass transport and surface reactions in microfluidic systems. *Chemical Engineering Science*, 61(4):1102–1121, 2006.

- [121] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. User's guide for NPSOL 5.0: a Fortran package for nonlinear programming. Technical report, Report NA 98-2, Department of Mathematics, University of California, San Diego, 1998.
- [122] David Glasser, Diane Hildebrandt, and Cameron Crowe. A geometric approach to steady flow reactors: The attainable region and optimization in concentration space. *Industrial & Engineering Chemistry Research*, 26(9):1803–1810, 1987.
- [123] Fred Glover. Improved linear integer programming formulations of nonlinear integer programs. *Management Science*, 22(4):455–460, 1975.
- [124] Fred Glover and Eugene Woolsey. Further reduction of zero-one polynomial programming problems to zero-one linear programming problems. *Operations Research*, 21(1):156–161, 1973.
- [125] Fred Glover and Eugene Woolsey. Converting the 0-1 polynomial programming problem to a 0-1 linear program. *Operations Research*, 22(1):180–182, 1974.
- [126] D. Gobby, I. Eames, and A. Gavriilidis. A vertically-averaged formulation of wall catalytic reactions in microchannel flows: Single isothermal & non-isothermal reactions. In *IMRET 5, Strasburg, France*, pages 141–149, 2001.
- [127] Gustaaf Goor and Jürgen Glenneberg. *Ullmann's Encyclopedia of Industrial Chemistry*, chapter Hydrogen Peroxide. John Wiley & Sons, 2000.
- [128] Harvey J. Greenberg. An annotated bibliography on post-solution analysis in mixed integer programming and combinatorial optimization. <http://carbon.cudenver.edu/hgreenbe/aboutme/papers/survey/mip.bib>, 1999.
- [129] Harvey J. Greenberg. Matrix sensitivity analysis from an interior solution of a linear program. *Informs Journal on Computing*, 11(3):316–327, 1999.
- [130] Ignacio E. Grossmann. Review of nonlinear mixed-integer and disjunctive programming techniques. *Optimization and Engineering*, 3(3):227–252, 2002.

- [131] Ignacio E. Grossmann and Keshava P. Halemane. Decomposition strategy for designing flexible chemical-plants. *AIChE Journal*, 28(4):686–694, 1982.
- [132] Ignacio E. Grossmann and Christodoulos A. Floudas. Active constraint strategy for flexibility analysis in chemical processes. *Computers and Chemical Engineering*, 11(6):675–693, 1987.
- [133] Zeynep H. Gümüs and Christodoulos A. Floudas. Global optimization of nonlinear bilevel programming problems. *Journal of Global Optimization*, 20(1):1–31, 2001.
- [134] Hang Guo and Chong-Fang Ma. 2d analytical model of a direct methanol fuel cell. *Electrochemistry Communications*, 6(3):306–312, 2004.
- [135] Dan Gusfield. Parametric combinatorial computing and a problem of program module distribution. *Journal of the Association for Computing Machinery*, 30(3):551–563, 1983.
- [136] Sossina M. Haile. Fuel cell materials and components. *Acta Materialia*, 51(19):5981–6000, 2003.
- [137] Elaine T. Hale and S. Joe Qin. Multi-parametric nonlinear programming and the evaluation of implicit optimization model adequacy. In *DYCOPS, Boston, MA*, 2004.
- [138] K. P. Halemane and I. E. Grossmann. Optimal process design under uncertainty. *AIChE Journal*, 29(3):425–433, 1983.
- [139] W. K. Klein Haneveld, C. L. J. Vandermeer, and R. J. Peters. Construction method in parametric programming. *Mathematical Programming*, 16(1):21–36, 1979.
- [140] Shinji Hasebe. Design and operation of micro-chemical plants—bridging the gap between nano, micro and macro technologies. *Computers and Chemical Engineering*, 29(1):57–64, 2004.

- [141] C. Hebling, A. Heinzl, D. Golombowski, T. Meyer, M. Müller, and M. Zedda. Fuel cells for low power applications. In *IMRET 3, Frankfurt, Germany*, pages 383–401, 1999.
- [142] A. Heinzl, C. Hebling, M. Müller, M. Zedda, and C. Müller. Fuel cells for low power applications. *Journal of Power Sources*, 105:250–255, 2002.
- [143] Michael M. Hencke, Alexander Mitsos, and Paul I. Barton. Fluidic valve modeling results. Technical report, Massachusetts Institute of Technology, 2004.
- [144] Michael M. Hencke, Alexander Mitsos, and Paul I. Barton. Modeling results of the heat exchanger portion of Ole M. Nielsen’s microreactor. Technical report, Massachusetts Institute of Technology, 2004.
- [145] Joshua L. Hertz and Harry L. Tuller. Electrochemical characterization of thin films for a micro-solid oxide fuel cell. *Journal of Electroceramics*, 13(1-3):663–668, 2004.
- [146] Volker Hessel and Holger Löwe. Mikroverfahrenstechnik: Komponenten - Anlagenkonzeption - Anwenderakzeptanz - Teil 1. *Chemie Ingenieur Technik*, 74(1-2):17–37, 2002.
- [147] Volker Hessel and Holger Löwe. Mikroverfahrenstechnik: Komponenten - Anlagenkonzeption - Anwenderakzeptanz - Teil 2. *Chemie Ingenieur Technik*, 74(3):185–208, 2002.
- [148] Volker Hessel and Holger Löwe. Mikroverfahrenstechnik: Komponenten - Anlagenkonzeption - Anwenderakzeptanz - Teil 3. *Chemie Ingenieur Technik*, 74(4):381–399, 2002.
- [149] L. K. Heung. Using metal hydride to store hydrogen. Technical report, Savannah River Technology Center. <http://sti.srs.gov/fulltext/ms2003172/ms2003172.pdf>, 2003.

- [150] J. D. Holladay, J. S. Wainright, E. O. Jones, and S. R. Gano. Power generation using a mesoscale fuel cell integrated with a microscale fuel processor. *Journal of Power Sources*, 130(1-2):111–118, 2004.
- [151] J. D. Holladay, Y. Wang, and E. Jones. Review of developments in portable hydrogen production using microreactor technology. *Chemical Reviews*, 104(10):4767–4789, 2004.
- [152] J. F. M. Horn. Attainable and nonattainable regions in chemical reaction technique. In *Chemical Reaction Engineering (Proceedings of the Third European Symposium)*, pages 1–10, London, 1964. Pergamon Press.
- [153] Reiner Horst. Deterministic global optimization with partition sets whose feasibility is not known: Application to concave minimization, reverse convex constraints, DC-programming, and Lipschitzian optimization. *Journal of Optimization Theory and Applications*, 58(1):11–37, 1988.
- [154] Reiner Horst and Hoang Tuy. *Global Optimization: Deterministic Approaches*. Springer, Berlin, Third edition, 1996.
- [155] I. Ming Hsing, Ravi Srinivasan, Michael P. Harold, Klavs F. Jensen, and Martin A. Schmidt. Simulation of micromachined chemical reactors for heterogeneous partial oxidation reactions. *Chemical Engineering Science*, 55(1):3–13, 2000.
- [156] Gregory T. Huang. Wearable devices add strength. *Technology Review*, page 26, February 2004.
- [157] Gregory T. Huang. Wearable robots. *Technology Review*, pages 70–73, July/August 2004.
- [158] J. P. Huijsmans. Ceramics in solid oxide fuel cells. *Current Opinion Solid State Material Science*, 5(4):317–323, 2001.

- [159] Raphael O. Idem and Narendra N. Bakhshi. Kinetic modeling of the production of hydrogen from the methanol-steam reforming process over Mn-promoted co-precipitated Cu-Al catalyst. *Chemical Engineering Science*, 51(14):3697–3708, 1996.
- [160] R. Jacobs, H. Christopher, R. Hamlen, R. Rizzo, R. Paur, and S. Gilman. Portable power source needs of the future army - Batteries and fuel cells. *IEEE Aerospace and Electronic Systems Magazine*, 11(6):19–23, 1996.
- [161] Vipul Jain and Ignacio E. Grossmann. Resource-constrained scheduling of tests in new product development. *Industrial & Engineering Chemistry Research*, 38(8):3013–3026, 1999.
- [162] Larry Jenkins. Parametric mixed integer programming: An application to solid waste management. *Management Science*, 28(11):1270–1284, 1982.
- [163] Larry Jenkins. Using parametric integer programming to plan the mix of an air transport fleet. *INFOR*, 25(2):117–135, 1987.
- [164] Larry Jenkins. Parametric methods in integer linear programming. *Annals of Operations Research*, 27:77–96, 1990.
- [165] Klavs F. Jensen. Microchemical systems: Status, challenges and opportunities. *AIChE Journal*, 45(10):2051–2054, 1999.
- [166] Klavs F. Jensen. Microreaction engineering – is small better? *Chemical Engineering Science*, 56(2):293–303, 2001.
- [167] Niket S. Kaisare. *Modeling, Analysis and Control of Nonlinear Switching Systems*. PhD thesis, Georgia Institute of Technology, 2004.
- [168] Yoshihiro Kawamura, Naotsugu Ogura, Tadao Yamamoto, and Akira Igarashi. A miniaturized methanol reformer with Si-based microreactor for a small PEMFC. *Chemical Engineering Science*, 61(4):1092–1101, 2006.

- [169] Padmanaban Kesavan, Russell J. Allgor, Edward P. Gatzke, and Paul I. Barton. Outer approximation algorithms for separable nonconvex mixed-integer nonlinear programs. *Mathematical Programming*, 100(3):517–535, 2004.
- [170] Padmanaban Kesavan and Paul I. Barton. Decomposition algorithms for nonconvex mixed-integer nonlinear programs. In *AIChE Symposium Series*, volume 96, pages 458–461, 2000.
- [171] Padmanaban Kesavan and Paul I. Barton. Generalized branch-and-cut framework for mixed-integer nonlinear optimization problems. *Computers & Chemical Engineering*, 24(2-7):1361–1366, 2000.
- [172] C. J. King. *Separation Processes*. McGraw-Hill, New York, 1980.
- [173] Dirk Kirschneck and Rolf Marr. Anlagenkonzepte in der Mikroverfahrenstechnik. *Chemie Ingenieur Technik*, 78(1-2):29–38, 2006.
- [174] G. Kolb, V. Cominos, C. Hofmann, H. Pennemann, J. Shürer, D. Tiemann, M. Wichert, R. Zapf, V. Hessel, and H. Löwe. Integrated microstructured fuel processors for fuel cell applications. *Chemical Engineering Research & Design*, 83(A6):626–633, 2005.
- [175] Arthur D. Kuo. BIOPHYSICS: Harvesting Energy by Improving the Economy of Human Walking. *Science*, 309(5741):1686–1687, 2005.
- [176] G. L. La, J. Hertz, H. Tuller, and Y. Shao-Horn. Microstructural features of RF-sputtered SOFC anode and electrolyte materials. *Journal of Electroceramics*, 13(1-3):691–695, 2004.
- [177] Ajay Lakshmanan and Lorenz T. Biegler. Synthesis of optimal chemical reactor networks. *Industrial & Engineering Chemistry Research*, 35(4):1344–1353, 1996.
- [178] J. Larminie and A. Dicks. *Fuel Cell Systems Explained*. John Wiley & Sons, Ltd, Chichester, England, 2nd edition, 2003.
- [179] Hydrogen Peroxide LCSS. <http://www.hhmi.org/research/labsafe/lcss/>.

- [180] Panayiotis Lemonidis and Paul I. Barton. Interval methods for generalized semi-infinite programs. International Conference on Parametric Optimization and Related Topics (PARAOPT VIII), Cairo, Egypt, Nov 27-Dec 1 2005.
- [181] Panayiotis Lemonidis and Paul I. Barton. Global solution of generalized semi-infinite programs. *In preparation*, 2006.
- [182] Leo Liberti and Constantinos C. Pantelides. Convex envelopes of monomials of odd degree. *Journal of Global Optimization*, 25(2):157–168, 2003.
- [183] David Linden. *Handbook of Batteries*. McGraw-Hill, 2001.
- [184] Kirk Lingell, Robert Sievers, Brent Mattes, and Jan Pantolin. Micro-power AMTEC systems. *IEEE Aerospace and Electronic Systems Magazine*, 16(3):33–36, 2001.
- [185] Stefan Löbbecke. Baukastenkonzepte für die Mikroverfahrenstechnik: Das FAMOS-System. *Chemie Ingenieur Technik*, 76(5):581–583, 2004.
- [186] Stefan Löbbecke, T. Tuercke, W. Schweikert, F. Schnuerer, J. Antes, D. Boskovich, E. Marioth, and H. H. Krause. Black box microreactor? - Possibilities for a better control and understanding of microreaction processes by applying suitable analytics. In *IMRET 6, New Orleans*, pages 37–38, 2002.
- [187] Matthew W. Losey, Martin A. Schmidt, and Klavs F. Jensen. Microfabricated multiphase packed-bed reactors: Characterization of mass transfer and reactions. *Industrial and Engineering Chemistry Research*, 40(12):2555–2562, 2001.
- [188] G. Q. Lu, C. Y. Wang, T. J. Yen, and X. Zhang. Development and characterization of a silicon-based micro direct methanol fuel cell. *Electrochimica Acta*, 49(5):821–828, 2004.
- [189] Charles Mann. Near-term nanotech. *Technology Review*, page 22, July/August 2004.

- [190] Christos T. Maravelias and Ignacio E. Grossmann. Optimal resource investment and scheduling of tests for new product development. *Computers & Chemical Engineering*, 28(6-7):1021–1038, 2004.
- [191] Hellen L. Maynard and Jeremy P. Meyers. Miniature fuel cells for portable power: Design considerations and challenges. *Journal of Vacuum Science Technologies*, 20(4):1287–1297, 2002.
- [192] R. Metkemeijer and P. Achard. Ammonia as a feedstock for a hydrogen fuel cell; reformer and fuel cell behaviour. *Journal of Power Sources*, 49(1-3):271–282, 1994.
- [193] Jeremy P. Meyers and Helen L. Maynard. Design considerations for miniaturized PEM fuel cells. *Journal of Power Sources*, 109(1):76–88, 2002.
- [194] A. B. Mhadeshwar and D. G. Vlachos. Hierarchical multiscale mechanism development for methane partial oxidation and reforming and for thermal decomposition of oxygenates on Rh. *Journal of Physical Chemistry B*, 109(35):16819–16835, 2005.
- [195] Maurice Mignotte. *Mathematics for Computer Algebra*. Springer, New York, 1992.
- [196] Alexander Mitsos and Paul I. Barton. Publications on portable power generation, <http://yoric.mit.edu/download/reports/micropowerpub.pdf>. Technical report, Massachusetts Institute of Technology, 2006.
- [197] Alexander Mitsos and Paul I. Barton. Issues in the development of global optimization algorithms for bilevel programs with a nonconvex inner program. *Submitted to: Journal of Global Optimization*, February 13, 2006.
- [198] Alexander Mitsos, Benoît Chachuat, Mehmet Yunt, Ignasi Palou-Rivera, Michael M. Hencke, and Paul I. Barton. <http://yoric.mit.edu/micropower>.

- [199] Alexander Mitsos, Michael M. Hencke, and Paul I. Barton. <https://yoric.mit.edu/micropower>, 2004.
- [200] Alexander Mitsos, Michael M. Hencke, and Paul I. Barton. Product engineering for man-portable power generation based on fuel cells. *AIChE Journal*, 51(8):2199–2219, 2005.
- [201] Alexander Mitsos, Ignasi Palou-Rivera, and Paul I. Barton. Alternatives for micropower generation processes. *Industrial and Engineering Chemistry Research*, 43(1):74–84, 2004.
- [202] Geoff D. Moggridge and Edward L. Cussler. An introduction to chemical product design. *Chemical Engineering Research and Design*, 78(A1):5–11, 2000.
- [203] James T. Moore and Jonathan F. Bard. The mixed integer linear bilevel programming problem. *Operations Research*, 38(5):911–921, 1990.
- [204] R. Moore. *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, PA, 1979.
- [205] Jeffrey D. Morse, Alan F. Jankowski, Robert T. Graff, and P. Hayes. Novel proton exchange membrane thin-film fuel cell for microscale energy conversion. *Journal of Vacuum Science & Technology A-Vacuum Surfaces and Films*, 18(4), 2000.
- [206] Hydrogen Peroxide MSDS. <http://www.bu.edu/es/labsafety/ESMSDSs/MSHydPeroxide.html>.
- [207] Katta G. Murty. Computational complexity of parametric linear programming. *Mathematical Programming*, 19(2):213–219, 1980.
- [208] George L. Nemhauser and Laurence A. Wolsey. *Integer and Combinatorial Optimization*. Wiley-Interscience, New York, 1999.
- [209] R. Neubert. Approximation of solution curves of underdetermined systems of nonlinear equations. *Computing*, 59(4):285–306, 1997.

- [210] Z. Ni, E. G. Seebauer, and R. I. Masel. Effects of microreactor geometry on performance: Differences between posted reactors and channel reactors. *Industrial & Engineering Chemistry Research*, 44(12):4267–4271, 2005.
- [211] Ole M. Nielsen. *A Thermophotovoltaic Micro-Generator for Portable Power Applications*. PhD thesis, Massachusetts Institute of Technology, 2006.
- [212] Ole M. Nielsen, Leonel R. Arana, Chelsey D. Baertsch, Klavs F. Jensen, and Martin A. Schmidt. A thermophotovoltaic micro-generator for portable power applications. In *Transducers 03 The 12th International Conference on Solid State Sensors, Actuators and Microsystems*,, pages 714–717, 2003.
- [213] D. G. Norton, E. D. Wetzel, and D. G. Vlachos. Thermal management in catalytic microreactors. *Industrial & Engineering Chemistry Research*, 45(1):76–84, 2006.
- [214] Dan G. Norton and Dionisios G. Vlachos. Combustion characteristics and flame stability at the microscale: A CFD study of premixed methane/air mixtures. *Chemical Engineering Science*, 58(21):4871–4882, 2003.
- [215] Dan G. Norton and Dionisios G. Vlachos. A CFD study of propane/air microflame stability. *Combustion and Flame*, 138(1-2):97–107, 2004.
- [216] Yoshiaki Ohtake and Naonori Nishida. A branch-and-bound algorithm for 0-1 parametric mixed integer programming. *Operations Research Letters*, 4(1):41–45, 1985.
- [217] K. B. Oldham and J. C. Myland. *Fundamentals of Electrochemical Science*. Academic Press, Inc., San Diego, 1994.
- [218] Oluwayemisi O. Oluwole, Binita Bhattacharjee, John E. Tolsma, Paul I. Barton, and William H. Green Jr. Rigorous valid ranges for optimally reduced kinetic models. *Combustion and Flame*, 146(1-2):348–365, 2006.

- [219] D. R. Palo, J. D. Holladay, R. T. Rozmiarek, C. E. Guzman-Leong, Y. Wang, J. Hu, Y.-H. Chin, R. A. Dagle, and E. G. Baker. Development of a soldier-portable fuel cell power system Part I: A bread-board methanol fuel processor. *Journal of Power Sources*, 108(1-2):28–34, 2002.
- [220] Katerina P. Papalexandri and Efstratios N. Pistikopoulos. Generalized modular representation framework for process synthesis. *AIChE Journal*, 42(4):1010–1032, 1996.
- [221] Taeshin Park and Paul I. Barton. State event location in differential-algebraic models. *ACM Transactions on Modelling and Computer Simulation*, 6(2):137–165, 1996.
- [222] Ashok S. Patil, Terry G. Dubois, Nicholas Sifer, Elizabeth Bostic, Kristopher Gardner, Michael Quah, and Christopher Bolton. Portable fuel cell systems for america’s army: Technology transition to the field. *Journal of Power Sources*, 136(2):220–225, 2004.
- [223] Robert H. Perry and Donald W. Green. *Perry’s Chemical Engineering Handbook*. <http://www.knovel.com>, 2002.
- [224] Anastasios Pertsinidis. *On the Parametric Optimization of Mathematical Programs with Binary Variables and its Applications in Chemical Engineering Process Synthesis*. PhD thesis, Carnegie Mellon, 1991.
- [225] Anastasios Pertsinidis, Ignacio E. Grossmann, and Gregory J. McRae. Parametric optimization of MILP programs and a framework for the parametric optimization of MINLPs. *Computers & Chemical Engineering*, 22(Suppl):205–212, 1998.
- [226] Max S. Peters, Klaus D. Timmerhaus, and Ronald E. West. *Plant Design and Economics for Chemical Engineers*. McGraw Hill, New York, fifth edition, 2003.

- [227] Anton J. Pfeiffer, Tamal Mukherjee, and Steinar Hauan. Design and optimization of compact microscale electrophoretic separation systems. *Industrial and Engineering Chemistry Research*, 43(14):3539–3553, 2004.
- [228] T. Pignet and L. D. Schmidt. Selectivity of NH_3 oxidation on Platinum. *Chemical Engineering Science*, 29(5):1123–1131, 1974.
- [229] T. Pignet and L. D. Schmidt. Kinetics of NH_3 oxidation on Pt, Rh and Pd. *Journal of Catalysis*, 40(2):212–225, 1975.
- [230] Efstratios N. Pistikopoulos, Vivek Dua, Nikolaos A. Bozinis, Alberto Bemporad, and Manfred Morari. On-line optimization via off-line parametric optimization tools. *Computers and Chemical Engineering*, 24(2-7):183–188, 2000.
- [231] Efstratios N. Pistikopoulos, Vivek Dua, Nikolaos A. Bozinis, Alberto Bemporad, and Manfred Morari. On-line optimization via off-line parametric optimization tools. *Computers and Chemical Engineering*, 26(2):175–185, 2002.
- [232] Efstratios N. Pistikopoulos, Vivek Dua, and Jun-Hyung Ryu. Global optimization of bilevel programming problems via parametric programming. In Christodoulos A. Floudas and Panos M. Pardalos, editors, *Frontiers in Global Optimization*, pages 457–476. Santorini, Greece: Kluwer Academic Publishers, 2003.
- [233] David J. Quiram, I.-Ming Hsing, Aleksander J. Franz, Klavs F. Jensen, and Martin A. Schmidt. Design issues for membrane-based, gas phase microchemical systems. *Chemical Engineering Science*, 55(16):3065–3075, 2000.
- [234] Robert C. Reid, John M. Prausnitz, and Bruce E. Poling. *The Properties of Gases and Liquids*. McGraw Hill, New York, 1987.
- [235] Xiaoming Ren, Thomas E. Springer, Thomas A. Zawodzinski, and Shimshon Gottesfeld. Methanol transport through nafion membranes. *Journal of the Electrochemical Society*, 147(2):466–474, 2000.

- [236] Pierre Reuse, Albert Renken, Katha Haas-Santo, Oliver Görke, and Klaus Schubert. Hydrogen production for fuel cell application in an autothermal micro-channel reactor. *Chemical Engineering Journal*, 101(1-3):133–141, 2004.
- [237] Y. W. Rhee, S. Y. Ha, and R. I. Masel. Crossover of formic acid through nafion membranes. *Journal of Power Sources*, 117(1-2):35–38, 2003.
- [238] Werner C. Rheinboldt and John V. Burkardt. A locally parameterized continuation process. *ACM Transactions on Mathematical Software*, 9(2):215–235, 1983.
- [239] Werner C. Rheinboldt and John V. Burkardt. A program for a locally parameterized continuation process. *ACM Transactions on Mathematical Software*, 9(2):236–241, 1983.
- [240] C. Rice, S. Ha, R. I. Masel, P. Waszczuk, A. Wieckowski, and Tom Barnard. Direct formic acid fuel cells. *Journal of Power Sources*, 111(1):83–89, 2002.
- [241] C. Rice, S. Ha, R. I. Masel, and A. Wieckowski. Catalysts for direct formic acid fuel cells. *Journal of Power Sources*, 115(2):229–235, 2003.
- [242] Irven Rinard. Miniplant design methodology. In *IMRET 2, New Orleans*, pages 299–312, 1998.
- [243] Lawrence C. Rome, Louis Flynn, Evan M. Goldman, and Taeseung D. Yoo. Generating electricity while walking with loads. *Science*, 309(5741):1725–1728, 2005.
- [244] Walter Rudin. *Principles of Mathematical Analysis*. McGraw-Hill, Singapore, third edition, 1976.
- [245] Nikhil Saha and I. H. Rinard. Miniplant design methodology: A case study manufacture of hydrogen cyanide. In *IMRET 4, Atlanta*, pages 327–333, 2000.

- [246] Kemal H. Sahin and Amy R. Ciric. A dual temperature simulated annealing approach for solving bilevel programming problems. *Computers and Chemical Engineering*, 23(1):11–25, 1998.
- [247] Nick Sahinidis and Mohit Tawarmalani. BARON. <http://www.gams.com/solvers/baron.pdf>, 2005.
- [248] Yuusuke Sato, Kei Matsuoka, Eiichi Sakaue, and Kyoji Hayashi. Material and heat management in a DMFC for portable usage. In *AIChE Annual Meeting*, 2005, Cincinnati, OH.
- [249] R. S. Segall. Using branch-and-bound to solve bi-level geometric-programming problems: A new optimization model. *Applied Mathematical Modelling*, 14(5):271–274, 1990.
- [250] R. S. Segall. An update on bi-level geometric-programming: A new optimization model. *Applied Mathematical Modelling*, 17(4):219–222, 1993.
- [251] Rüdiger U. Seydel. *Practical Bifurcation and Stability Analysis: From Equilibrium to Chaos*. Springer-Verlag, New York, 2nd edition, 1994.
- [252] Zongping Shao, Sossina M. Haile, Jeongmin Ahn, Paul D. Ronney, Zhongliang Zhan, and Scott A. Barnett. A thermally self-sustained micro solid oxide fuel-cell stack with high power density. *Nature*, 435(9):795–798, 2005.
- [253] Kiyotaka Shimizu, Yo Ishizuka, and Jonathan F. Bard. *Nondifferentiable and Two-Level Mathematical Programming*. Kluwer Academic Publishers, Boston, 1997.
- [254] J. Shu, B. P. A. Grandjean, A. Vanneste, and S. Kallaguine. Catalytic palladium-based membrane reactors - A review. *Canadian Journal of Chemical Engineering*, 69(5):1036–1060, 1991.
- [255] C. Sigal and C. G. Vayenas. Ammonia oxidation to nitric oxide in a solid electrolyte fuel cell. *Solid State Ionics*, 5:567–570, 1981.

- [256] Darlene K. Slattery and Michael D. Hampton. Complex hydrides for hydrogen storage. Proceedings of the 2002 U.S. DOE Hydrogen Program Review <http://www.eere.energy.gov/hydrogenandfuelcells/pdfs/32405b34.pdf>.
- [257] William Robert Smith and Ronald W. Missen. *Chemical Reaction Equilibrium Analysis: Theory and Algorithms*. John Wiley & Sons, New York, 1982.
- [258] Chunshan Song. Fuel processing for low temperatures and high temperature fuel cells. Challenges, and opportunities for sustainable development in the 21st century. *Catalysis Today*, 77(1-2):17–49, 2002.
- [259] V. T. Srikar, Kevin T. Turner, Tze Yung Andre Ie, and S. Mark Spearing. Structural design considerations for micromachined solid-oxide fuel cells. *Journal of Power Sources*, 125(1):62–69, 2004.
- [260] Oliver Stein, Jan Oldenburg, and Wolfgang Marquardt. Continuous reformulations of discrete-continuous optimization problems. *Computers & Chemical Engineering*, 28(10):1951–1966, 2004.
- [261] Oliver Stein and Georg Still. On generalized semi-infinite optimization and bilevel optimization. *European Journal of Operational Research*, 142(3):444–462, 2002.
- [262] Oliver Stein and Georg Still. Solving semi-infinite optimization problems with interior point techniques. *SIAM Journal on Control and Optimization*, 42(3):769–788, 2003.
- [263] Dharmashankar Subramanian, Joseph F. Pekny, and Gintaras V. Reklaitis. A simulation-optimization framework for addressing combinatorial and stochastic aspects of an R&D pipeline management problem. *Computers & Chemical Engineering*, 24(2-7):1005–1011, 2000.
- [264] F. Takei, N. F. Cooray, K. Yoshida, H. Yoshida, K. Ebisu, S. Suzuki, and N. Sawatari. Development of prototype micro fuel cells for mobile electronics. *Fujitsu Scientific & Technical Journal*, 41(2):191–200, 2005.

- [265] Yanghua Tang, Kevin Stanley, Jonathan Wu, Dave Ghosh, and Jiujun J. Zhang. Design consideration of micro thin film solid-oxide fuel cells. *Journal of Micromechanics and Microengineering*, 15(9):S185–S192, 2005.
- [266] Mohit Tawarmalani and Nikolaos V. Sahinidis. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming*. Nonconvex Optimization and its Applications. Kluwer Academic Publishers, Boston, 2002.
- [267] Mohit Tawarmalani and Nikolaos V. Sahinidis. Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming*, 99(3):563–591, 2004.
- [268] W. E. TeGrotenhuis, D. L. King, K. P. Brooks, B. J. Golladay, and R. S. Weng. Optimizing microreactors by trading-off equilibrium and reaction kinetics through temperature managing. In *IMRET 6, New Orleans*, pages 18–28, 2002.
- [269] John Tolsma and Paul I. Barton. DAEPACK: An open modeling environment for legacy models. *Industrial and Engineering Chemistry Research*, 39(6):1826–1839, 2000.
- [270] John E. Tolsma. *DAEPACK DSL48SE Manual Version 1.0, Hybrid Discrete/Continuous Numerical Integration and Parametric Sensitivity Analysis*. MIT, March 2001.
- [271] John E. Tolsma and Paul I. Barton. <http://yoric.mit.edu/daepack/daepack.html>.
- [272] John E. Tolsma, Jerry Clabaugh, and Paul I. Barton. <http://yoric.mit.edu/abacuss2/abacuss2.html>.
- [273] John E. Tolsma, Jerry A. Clabaugh, and Paul I. Barton. Symbolic incorporation of external procedures into process modeling environments. *Industrial and Engineering Chemistry Research*, 41(16):3867–3876, 2002.
- [274] Luís N. Vicente and Paul H. Calamai. Bilevel and multilevel programming - A bibliography review. *Journal of Global Optimization*, 5(3):291–306, 1994.

- [275] František Štěpánek and Miloš Marek. Optimal design and operation of a separating microreactor. *Chemical Engineering Science*, 54(10):1493–1498, 1999.
- [276] František Štěpánek and Miloš Marek. Optimization of reaction-separation networks via mass integration on the μ -scale. In *IMRET 3, Frankfurt Germany*, pages 243–252, 1999.
- [277] Stein W. Wallace. Decision making under uncertainty: Is sensitivity analysis of any use? *Operations Research*, 48(1):20–25, 2000.
- [278] André Weber and Ellens Ivers-Tiffée. Materials and concepts for solid oxid fuel cells (SOFCs) in stationary and mobile applications. *Journal of Power Sources*, 127(1):273–283, 2004.
- [279] James Wei. Molecular structure and property: Product engineering. *Industrial and Engineering Chemistry Research*, 41(8):1917–1919, 2002.
- [280] Christiano Wibowo and Ka M. Ng. Product-centered processing: Manufacture of chemical-based consumer products. *AIChE Journal*, 48(6):1212–1230, 2002.
- [281] H. P. Williams. *Model Building in Mathematical Programming*. John Wiley and Sons, Chichester, third edition, 1990.
- [282] A. Wojcik, H. Middleton, I. Damopoulos, and J. Van Herle. Ammonia as a fuel in solid oxide fuel cells. *Journal of Power Sources*, 118(1/2):342–348, 2003.
- [283] W. M. Yang, S. K. Chouand, C. Shu, Z. W. Li, and H. Xue. Combustion in micro-cylindrical combustors with and without a backward facing step. *Applied Thermal Engineering*, 22(16):1777–1787, 2002.
- [284] Mehmet Yunt, Benoît Chachuat, Alexander Mitsos, and Paul I. Barton. Designing man-portable power generation systems for varying power demand. *In preparation*.
- [285] Yimin Zhu, Su Y. Ha, and Richard I. Masel. High power density direct formic acid fuel cells. *Journal of Power Sources*, 130(1-2):8–14, 2004.

- [286] S. Zuhe, A. Neumaier, and M. C. Eiermann. Solving minimax problems by interval-methods. *BIT*, 30(4):742–751, 1990.
- [287] A. Züttel, S. Rentsch, P. Fischer, P. Wenger, P. Sudan, Ph. Mauron, and Ch. Emmenegger. Hydrogen storage properties of LiBH_4 . *Journal of Alloys and Compounds*, 356-357:515–520, 2003.
- [288] A. Züttel, P. Wenger, P. Sudan, P. Mauron, and S. Orimo. Hydrogen density in nanostructured carbon, metals and complex materials. *Materials Science and Engineering*, B108(1-2):9–18, 2004.