Enhancements and Computational Evaluation of the Hit-and-Run Random Walk on Polyhedra

by

Jiajie Liang

B.Eng. Chemical Engineering (2005)
National University of Singapore

Submitted to the School of Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Computation for Design and Optimization

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2006

Author ....
School of Engineering
August 11, 2006

Certified by ....
Robert M. Freund
Theresa Seley Professor of Operations Research
Sloan School of Management
Codirector, Computation for Design and Optimization
Thesis Supervisor

Accepted by ....
Jaime Peraire
Professor of Aeronautics and Astronautics
Codirector, Computation for Design and Optimization

Enhancements and Computational Evaluation of the Hit-and-Run Random Walk on Polyhedra

by

Jiajie Liang

Submitted to the School of Engineering
on August 11, 2006, in partial fulfillment of the
requirements for the degree of
Master of Science in Computation for Design and Optimization

Abstract

The symmetry function of a convex set offers us numerous useful information about the set in relation to probabilistic theory and geometric properties. The symmetry function is a measure of how symmetric the convex set is, and for a point, intuitively it measures how symmetric the set is with respect to that point. We call a point of high symmetry value a *deep point*. A random walk is a procedure that starts from a particular point in $\Re^n$ and at each iteration, moves to a "neighboring" point according to some probability distribution that depends solely on the current point. The *Hit-and-Run* random walk on a convex set S picks a random line ℓ through the point, and at next iteration goes to a new point that is chosen uniformly on the chord $\ell \cap S$.

In this thesis, we analyze and investigate the effectiveness of the *Hit-and-Run* random walk to compute a deep point in a convex body, given a randomly generated convex set. The effectiveness is evaluated in terms of the role of the starting point and the likelihood that the random walk will enter the zone of high symmetry. Additionally, some known probabilistic properties of the symmetry function are tested using the random walk, from which the integrity of the code is also verified. The final portion of this thesis analyzes the behavioral properties of convex sets that have *non-Euclidean rounding*, which renders the random walk less efficient. Therefore the *pre-conditioned Hit-and-Run* random walk is performed, and the performance is quantitatively presented in a power law equation that predicts the pre-conditioning iterations required, given the dimension of the convex set, a starting point near a corner and the width of that corner.

Thesis Supervisor: Robert M. Freund
Title: Theresa Seley Professor of Operations Research
Sloan School of Management
Codirector, Computation for Design and Optimization

Acknowledgments

I would like to thank the following people for their help and guidance in the course of this thesis research.

First and formost, I would like to express my heart-felt gratitude and appreciation to my thesis advisor Robert M. Freund for being a truly great teacher. His wonderful guidance, cheerful motivation and ever friendly charming smile are crucial to my research. It is indeed my great pleasure to have him as my supervisor.

I am also very thankful to Alexandre Belloni, PhD in Operations Research, from whom I received many valuable insights and help, especially on using SDPT3.

My gratefulness also goes to my fellow colleagues in the Operations Research Center. I really appreciate the numerous helpful discussions and the fun-filled time spent together.

Last but not least, I must also thank my parents and friends for their love and support throughout my stay in MIT.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The symmetry function [1] of a point x in a convex set $S \in \Re^n$ is denoted as $sym(x, S)$ and is defined as:

$$sym(x, S) := max\{\alpha | \alpha \geq 0, x + \alpha(x - y) \in S \quad \forall y \in S\}$$

Correspondingly, the symmetry function of the convex set $sym(S)$ is defined as:

$$sym(S) := max_{x \in S} sym(x, S)$$

A point x^* that attains the maximum value is called a *symmetry point* of the convex set S, and we call points $x \in S$ that have high symmetry function values in relation to the convex set symmetry function value i.e. $sym(x, S) \approx sym(S)$ as *deep points.* We can see that if the convex set is fully symmetric, then $sym(S) = 1$, therefore $sym(S) \in (0, 1]$.

The symmetry function is a measure of how symmetric the convex set is, and for a point, it intuitively measures how symmetric the set is with respect to the point $x \in S$. It also provides us with numerous useful information about properties of the set. The symmetry value offers us intuitive understanding and quantitative measure of geometrical properties of the convex set such as volume, distance and rounding. Some probability theories of the convex set are also related to the symmetry function.

A geometric random walk process is one that starts at some point in $\Re^n$ and at each iteration, moves to a "neighboring" point according to some probability distribution, which is solely dependent on the current point. A geometric random walk simulates a Markov chain, and the steady state distribution of the points converges uniquely. Using different random walk algorithms such as *Grid Walk, Ball Walk*, and *Hit-and-Run*, different steady state distributions result [2]. In this thesis, the *Hit-and-Run* random walk is used in hope to generate a uniform steady state distribution. For a convex set S and a current point $x \in S$, the *Hit-and-Run* random walk starts off by choosing a random direction, from which a line ℓ is drawn passing through the current point x. We then perform two min-ratio tests to delimit the chord $\ell \cap S$. Finally a new point is chosen uniformly on the chord delimited above. After we perform enough *Hit-and-Run* random steps, the converging distribution is approximately uniform, in which we can gain some insights for computing *deep points.* A detailed discussion of the *Hit-and-Run* random walk implemented in this thesis can be found in section 2.3 on page 21.

We are particularly interested in *deep points* because they are good starting points for certain interior point algorithms [3]. The zone of *deep points* is non-trivial to compute especially when the dimension of the convex set is large or when the convex set is positioned in a highly *non-Euclidean* sense. In this thesis, we analyze and investigate the effectiveness of the *Hit-and-Run* random walk to compute a *deep point.* The analysis is performed on three basics which includes (i) the role of the starting point,(ii) the likelihood for the random walk to enter the zone of *deep points* and (iii) the relationship of the center of mass of uniform distribution to the symmetry point of the convex set. These results are presented in section 3.2 on page 31.

It is also our great interest to evaluate some known probabilistic properties involving the symmetry functions presented by Belloni and Freund [1] using the *Hit-and-Run* random walk. In this way, the integrity of our random walk set-up is also verified. This is outlined in section 3.1 on page 27.

Finally, the thesis also evaluates in detail how *non-Euclidean* rounding renders the random walk less efficient. It is highly likely for projective transformation, which is of great interests to some researchers [3] to result in *non-Euclidean* convex sets, while its origin is also probable to be situated near a corner of the set. Thus, the *Hit-and-Run* random walk based on random direction drawn from a Euclidean sphere will no longer work fine, and pre-conditioning is needed. In section 4.1 on page 39, we discuss in depth how to perform the pre-conditioning, and in section 4.3 on page 44, we present some quantitative heuristics that guides the number of pre-conditioning steps required in order to achieve fast uniform distribution convergence for the pre-conditioned random walk.

1.1 Thesis Outline

In Chapter 2, various theories involved in the *Hit-and-Ran* random walk analysis set-up are discussed in detail with the corresponding optimization model presented. Chapter 3 focuses on the verification of the integrity of the code by testing against some known probabilistic properties of the symmetry function, as well as the result for the effectiveness of the random walk. The pre-conditioning for *non-Euclidean* rounded convex sets and the quantitative power law results are the focus of chapter 4, while in chapter 5 we conclude the thesis with some discussions on future work.

Chapter 2

Theory and Optimization Model

2.1 Convex body and Boundness Test

2.1.1 Generation of the Convex Set

We wish to analyze the *Hit-and-Run* random walk on a convex set $S := \{x \in \Re^n | Ax \leq b\}$, where $A \in \Re^{m \times n}$. The convex set is generated randomly by generating each entry of matrix A and vector b independently, where every entry obeys a standard normal distribution. For manipulation convenience, we ensure that the origin lies in the strict interior of the convex set, and this is achieved by replacing each entry of b by its absolute value.

In the process of analysis, the dimension n increases. In order to to attain reasonable computational speed, we introduced sparsity of density 0.1 into the coefficient matrix A. The sparsity property is also in accordance with most real life applications, where non-zero entries are often very limited.

2.1.2 Convex Body Assumption

In order to compute efficiently the symmetry point and symmetry function, we must make an important assumption:

Convex Set S is a Convex Body, i.e. it is bounded with nonempty interior.

When S is a convex set but not a convex body, the notion of symmetry function is still defined. However, certain properties of the symmetry function will no longer be valid. Interested readers can refer to Belloni and Freund [1] for some general discussions.

2.1.3 Boundness Test

In order to fulfill the convex body assumption, the coefficient matrix A has to have full column rank, i.e. $rank(A) = n$. This is generally true given the generating technique used above. In addition to the full rank criterion, the boundness condition presents a larger challenge. Boundness of the convex set means that the *nonnegative combinations* of all the constraint vectors A_i (which is the row of coefficient matrix A) spans $\Re^n$. This is equivalent to the **inability** of finding a direction d that makes an obtuse angle with all the constraint vectors, i.e. $d|A_i d < 0 \quad \forall i$. Therefore the following LP is designed to test boundness:

$$\begin{aligned} \text{BT:} \quad \theta^* \quad = \quad & max_{\theta,d} \quad \theta \\ \text{s.t.} \quad & Ad + \theta e \leq 0 \\ & e^T Ad = -1 \end{aligned} \tag{2.1}$$

In the above formulation, Ad is normalized. In the case of unboundedness, every component of Ad will be non-positive, and the optimal value θ^* will be nonnegative. Thus if a negative θ^* is obtained, the convex set is a bounded convex body, or otherwise the coefficient matrix A will be re-generated.

It is easy to notice that as the number of constraints m increases in a particular dimension $\Re^n$, the chance of unboundedness diminishes. Using the above convex set generation scheme

Table 2.1: Unboundness on Coefficient Matrix Size with $n = 20$

Matrix Size	Percent Unbounded
$2n \times n$	36.33%
$3n \times n$	2.31%
$4n \times n$	0.01%

and the boundness test, coefficient matrix size of $2n \times n$, $3n \times n$ and $4n \times n$ are tested. Three runs of 3000 samples for each matrix size and their percentage unboundedness is tabulated.

From Table 2.1, it is clear that with the coefficient matrix size of $4n \times n$ (and predictably larger size), the likelihood of unboundedness is very low. However choosing such a large matrix size will severely affect the computation speed as the dimension n ramps up. On the other hand coefficient matrix size of $3n \times n$ offers reasonably good boundedness percentage and yet renders the least amount of future computation. Therefore as a guideline, coefficient matrix A will be generated using the size of $3n \times n$ for all future experiments.

2.2 Computation of Symmetry Function and Symmetry Point

In order to investigate the effectiveness of the *Hit-and-Run* random walk on the ability and likelihood to enter the zone of *deep points*, the symmetry function of the convex set and of any point must be readily available.

There are two methods established by Belloni and Freund [1] for computing the symmetry function $sym(S)$ of a bounded polyhedron S, where S is given as the intersection of m inequalities: $S := \{x | Ax \leq b\}$.The first method is based on a theorem of alternative and results in the following linear program:

$$\begin{aligned} \text{TOA:} \quad & min_{y,\Pi,\gamma} \quad \gamma \\ \text{s.t.} \quad & \Pi A = -A \\ & \Pi b + Ay - b\gamma \leq 0 \\ & \Pi \geq 0 \end{aligned} \tag{2.2}$$

where $sym(S) := \frac{1}{\gamma^*}$ and $x^* := \frac{y^*}{1+\gamma^*}$, with (y^*, γ^*) being the optimal solution.

In formulation 2.2, there are in total m^2+m+mn constraints and m^2+n+1 variables. Because of the m^2+m inequalities, an interior point algorithm would require $O(m^6)$ operations per Newton step, and is undoubtedly undesirable.

The second method is based on proposition 1 of [1] and engenders the following linear programs:

$$\begin{aligned} DF_i: \quad \delta_i^* \quad := \quad & max_x - A_i x \\ \text{s.t.} \quad & Ax \leq b \end{aligned} \tag{2.3}$$

m linear program are performed to obtain $\delta_i^*, i = 1, \ldots, m$.

The symmetry value of any point $x \in S$ can then be easily calculated as:

$$sym(x, S) = min_{i=1,\ldots,m} \left\{ \frac{b_i - A_i x}{\delta_i^* + A_i x} \right\} \tag{2.4}$$

An additional linear program is needed to obtain the symmetry value of the convex polyhedron and the symmetry point. Define δ^* as $(\delta_1^*, \ldots, \delta_m^*)$ and use the data from previous m linear program:

$$\begin{aligned} \text{Symmetry:} \quad & max_{x,\check{\theta}} \quad \check{\theta} \\ \text{s.t.} \quad & Ax + \check{\theta}(\delta^* + b) \leq b \end{aligned} \tag{2.5}$$

where $sym(S) := \frac{\check{\theta}^*}{1-\check{\theta}^*}$ and x^* is the symmetry point.

The above method employs $m+1$ linear programs, each with n or $n+1$ variables and m

constraints, in which SDPT3 solver is used to obtain the symmetry function value fairly efficiently.

2.3 Hit-and-Run Procedure and Min-Ratio Test

2.3.1 Hit-and-Run Random Walk

The *Hit-and-Run* random walk on a convex set S starts from a feasible current point. A random point is then picked uniformly on a sphere around the current point. Connecting these two points to generate a line ℓ and two min-ratio tests are done to delimit the feasible portion of the line. Finally a new point will be picked uniformly along the feasible chord and the iterations continue. The *Hit-and-Run* random walk simulates a Markov Chain and it uniquely converges to uniform distribution given enough iterations performed.

In this thesis's set up, the *Hit-and-Run* random walk is implemented in the following way:

1. Start from a feasible, strictly interior point $\bar{x}$ such that $A\bar{x} < b$.

2. Generate n independent random number $d_1, \ldots, d_n$, each from the standard normal distribution.

3. Let $d = (d_1, \ldots, d_n)$ be normalized, i.e., $d \leftarrow d/\|d\|$, and the direction vector d is now uniformly distributed on a Euclidean sphere, i.e. $d \sim N(0, I)$.

4. Perform two min-ratio tests to obtain the interval $[\beta^L, \beta^U]$, where $A(\bar{x} + \beta^U d) \leq b, \quad \beta^U \geq 0$ and $A(\bar{x} + \beta^L d) \leq b, \quad \beta^L \leq 0$.

5. Choose α uniformly in the interval $[\beta^L, \beta^U]$, i.e. $\alpha \sim U[\beta^L, \beta^U]$.

6. Update the current point by $\bar{x} = \bar{x} + \alpha d$.

7. Iteration repeats from step 1.

In the above implementation, the random direction is picked on a Euclidean sphere. However in the actual case, the convex set may not be round enough in a Euclidean sense. Therefore the speed for convergence to uniform distribution may be slowed. This will be further studied in section 4.1 on page 39.

2.3.2 Min-Ratio Test

The min-ratio test is performed twice in each random walk iteration, and it is also used in the determination of the feasible starting corner discussed in section 2.4.1 on page 23 and finding the distance ϵ of a particular point away from its nearest corner discussed in section 2.4.2 on page 24. Therefore its detailed implementation is illustrated below together with some numerical issues addressed:

1. Setting $\beta^U := \infty$ and $\beta^L := -\infty$.

2. Repeat from $i = 1, \ldots, m$:

   - Evaluate $Num := b_i - A_i\bar{x}$ and $Den := A_i d$
   - If $|Num| < 10^{-14}$ then set $Num = 0$
   - If $|Den| < 10^{-14}$ then set $Den = 0$
   - If $Den > 0$ and $\frac{Num}{Den} < \beta^U$, then update $\beta^U := \frac{Num}{Den}$
   - If $Den < 0$ and $\frac{Num}{Den} > \beta^L$, then update $\beta^L := \frac{Num}{Den}$

2.4 Starting Point Consideration

In analyzing the effectiveness of the *Hit-and-Run* random walk, one of the important aspects is the role of the starting point. Using the above illustrated convex set generation technique, the origin is easily the most intuitive starting point to consider. Other than the origin, finding an initial feasible point is non-trivial and the following algorithm is used to obtain such a point.

2.4.1 Finding a Feasible Corner

It is of great interest to consider a starting point that is jammed inside a corner or near a corner, therefore finding an initial feasible corner of a polyhedron is highly useful. The following implementation is designed to find a feasible corner:

1. Set the corner index set to null, i.e. $indexset = [\ \ ]$.

2. Start from the origin and generate a normalized random direction $\bar{d}$.

3. Perform the min-ratio test to obtain the interval $[\beta^L, \beta^U]$ and its corresponding index number (The row number of coefficient matrix A) that attains these limits, noted as $[index_L, index_U]$.

4. If $index_U$ is updated,
   then $indexset := indexset \cup index_U$ and $\bar{A} := A_{index_U}$
   else $indexset := indexset \cup index_L$ and $\bar{A} := A_{index_L}$.

5. While $|indexset| < n$, repeat

   - Generate a normalized random direction d.

   - Obtain the projection vector d_{proj} of d onto the null space of $\bar{A}$ using
     $d_{proj} := \left[I - \bar{A}^T(\bar{A}\bar{A}^T)^{-1}\bar{A}\right]d$.

   - Perform min-ratio test to obtain the interval $[\beta^L, \beta^U]$ as well as $[index_L, index_U]$. The index i where $A_i d = 0$ in the min-ratio test is also recorded in $indexzeroset$.

   - If $index_U \neq 0$, and $index_U \notin indexset$
     then $indexset := indexset \cup index_U$ and $\bar{A} := \bar{A} \cup A_{index_U}$
     else if $index_L \neq 0$, and $index_L \notin indexset$
     then $indexset := indexset \cup index_L$ and $\bar{A} := \bar{A} \cup A_{index_L}$
     else pick first element j in $indexzeroset$ that $j \notin indexset$ and
     let $indexset := indexset \cup j$, $\bar{A} := \bar{A} \cup A_j$.

The above algorithm starts from the origin and first moves to the boundary of the convex set. Once hitting the boundary, it moves along that boundary defined by all previous boundaries until it hits another boundary. The process repeats until the cardinality of the indexset defining the corner becomes n, and the final point will be a feasible corner.

2.4.2 Obtaining a Starting Point Near a Corner

After locating the initial feasible corner, we move away slightly from the corner towards the direction of origin to obtain the initial starting point. The relative distance of the starting point away from the corner is measured by a self-defined quantity ϵ as follows:

1. Treating the starting point as the current point.
2. The direction vector $d = currentpoint - cornerpoint$.
3. Perform the min-ratio test to obtain the interval $[\beta^L, \beta^U]$.
4. Then $\epsilon := \frac{|\beta^L|}{|\beta^L|+|\beta^U|}$.

From Figure 2-1, $\epsilon := \frac{AB}{AC}$

2.5 Analysis Set Up

With the convex body generated randomly and the symmetry function calculation ready, the *Hit-and-Run* random walk is started from various starting points, including origin and points near/jammed into a corner. M random walk steps are performed and each point recorded and their point symmetry value calculated. The points' distribution statistics such as mean point and covariance matrix are also recorded as:

$$\begin{aligned} \bar{\mu} &= \tfrac{1}{M}\sum x_i \\ \bar{V} &= \tfrac{1}{M-1}\sum (x_i-\bar{\mu})(x_i-\bar{\mu})^T \end{aligned} \quad (2.6)$$

With these, the verification, analysis and investigations are demonstrated in chapter 3.

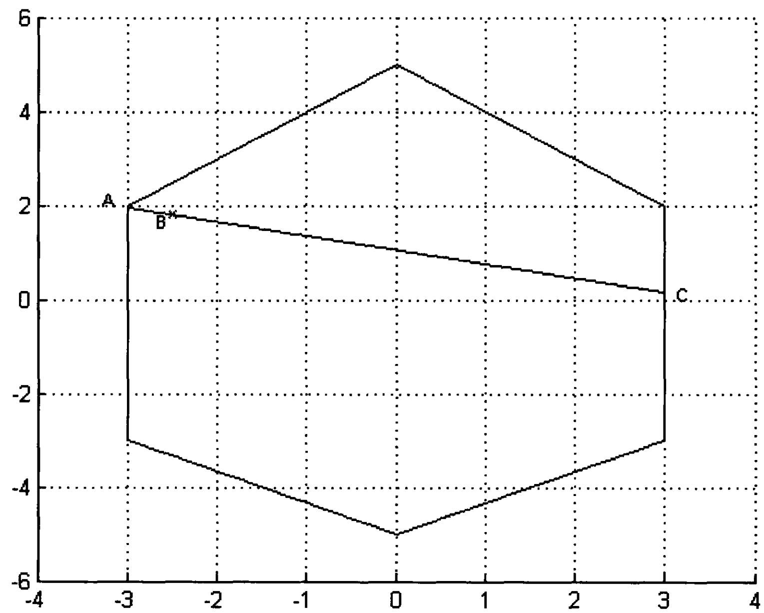

Figure 2-1: Self defined relative distance from a corner: ϵ

Chapter 3

Random Walk Integrity and Investigations

3.1 Integrity Testing

Using the convex set generation technique and the *Hit-and-Run* random walk implementation described in chapter 2, we can obtain a series of M points, in which each point's symmetry function value can be calculated using equation 2.4. It is known that given enough *Hit-and-Run* random walk iterations, i.e. a large enough M, the distribution will converge to the uniform distribution. Under uniform distribution, the symmetry function obeys some probabilistic properties that are established by Belloni and Freund [1].

They state that if X is a random uniformly distributed vector on any given convex body $S \in \Re^n$, then:

$$\text{For any } M \geq 1, \quad Pr\left(sym(X,S) \geq \frac{sym(S)}{M}\right) \geq \left(1 - \frac{2}{M+1}\right)^n \tag{3.1}$$

and

$$\mathbb{E}[sym(X,S)] \geq \frac{sym(S)}{2n+1} \tag{3.2}$$

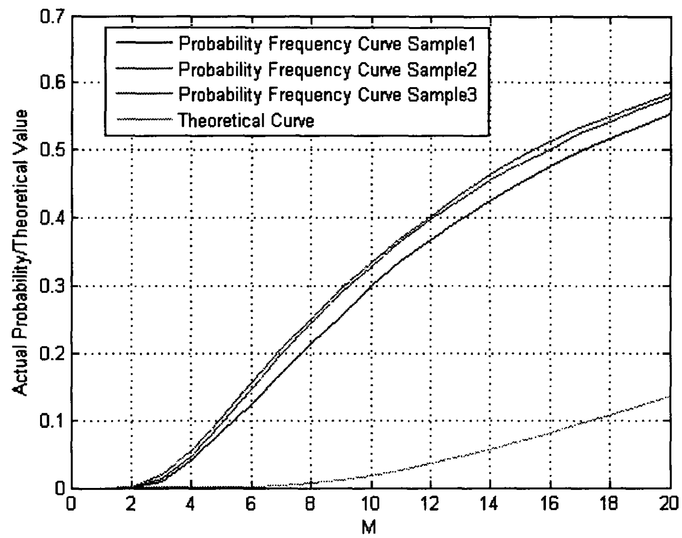

Figure 3-1: Probability Property Verification in term of $sym(S)/M$

They further state that for symmetric sets, i.e. $sym(S) = 1$:

$$\text{For any } M \geq 1, \quad Pr\left(sym(X,S) \geq \frac{1}{M}\right) = \left(1 - \frac{2}{M+1}\right)^n \tag{3.3}$$

and

$$\mathbb{E}[sym(X,S)] \leq \frac{1}{2(n+1)} + \frac{1}{(n+1)(n+2)} \tag{3.4}$$

These properties are excellent tools for verification of the integrity of the random walk procedure as well as the codes. Using the analysis set up presented in section 2.5, three different randomly generated convex bodies are tested against property 3.1 and in each case three runs are tested. The starting point chosen in these experiments are the origin and 100,000 steps of random walk are performed. The result shows that in all three cases, the above property is satisfied strictly. In fig 3-1, $Pr\big(sym(X,S) \geq \frac{1}{M}\big)$ is plotted against different M values, and the theoretical lower bound $\big(1 - \frac{2}{M+1}\big)^n$ is also plotted for one of the three convex body. It can be seen that the actual probability curve is at all time well above the lower bound curve.

Table 3.1: Expected Point Symmetry Property Verification

Matrix Size	Run ID	$\mathbb{E}[sym(X,S)]$	$\frac{sym(S)}{2n+1}$
	1	0.0147	0.0087
30×10	2	0.0162	0.0087
	3	0.0158	0.0087
	1	0.0097	0.0018
90×30	2	0.0103	0.0018
	3	0.0092	0.0018
	1	0.0045	0.00082
150×50	2	0.0048	0.00082
	3	0.0047	0.00082

The results for property 3.2 for the same three random convex bodies are tabulated in table 3.1. It is easily shown that the property does remain valid for all cases.

In order to verify the properties for the symmetric cases, three random symmetric convex bodies in the form of

$$S = \{x | Ax \le b, -Ax \le b\}$$

are generated. Again, three random walk runs each of 100,000 steps starting from the origin are performed for each of the three symmetric convex bodies. The results show that properties 3.3 and 3.4 are excellently satisfied. It can be shown from fig 3-2, in which $Pr(sym(X,S) \ge \frac{1}{M})$ is again plotted against different M values, and the theoretical lower bound $(1 - \frac{2}{M+1})^n$ also plotted for one of the three symmetric convex body. All the four lines (three actual probability curves and one theoretical bound curve) are very close to each other, indicating an excellent match to property 3.3.

In table 3.2, it is clearly demonstrated that property 3.4 holds for all cases.

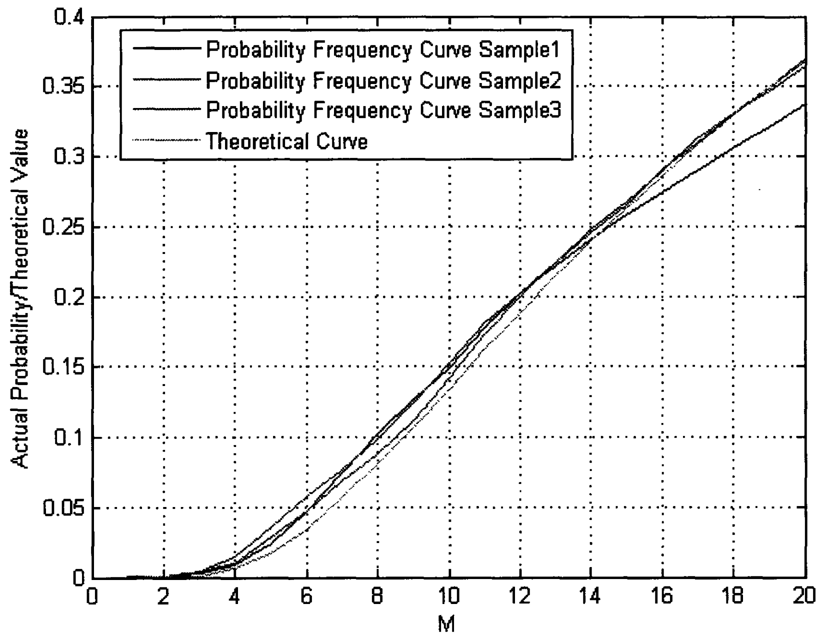

Figure 3-2: Probability Property Verification for Symmetric Convex Body in term of $sym(S)/M$

From the above verification, it is clear that both the random walk set up/codes and the probabilistic properties established by Belloni and Freund are indeed correct.

Table 3.2: Expected Point Symmetry Property Verification for Symmetric Convex Body

Matrix Size	Run ID	$\mathbb{E}[sym(X,S)]$	$\frac{1}{2(n+1)}+\frac{1}{(n+1)(n+2)}$
60×10	1	0.506	0.530
	2	0.519	0.530
	3	0.530	0.530
180×30	1	0.0170	0.0171
	2	0.0171	0.0171
	3	0.0168	0.0171
300×50	1	0.0101	0.0102
	2	0.0102	0.0102
	3	0.0101	0.0102

3.2 Hit-and-Run Random Walk Investigations

In the previous section, the integrity of the *Hit-and-Run* random walk set up is verified and in this section, a series of investigations of the effectiveness of the *Hit-and-Run* random walk are carried out. The analysis is primarily targeted at (i) the role of starting point (ii) the likelihood for the *Hit-and-Run* random walk to enter the zone of *deep points* and (iii) the relationship of the center of mass of uniform distribution to the symmetry point of the convex set.

In order to evaluate the effectiveness of the *Hit-and-Run* random walk in the above three aspects, the following experiments are designed to perform the task:

1. Five random convex sets are generated with dimension 10, 20, 30, 40 and 50.

2. For each convex set four starting points, namely: the origin and points with $\epsilon = 0.1, 0.01$ and 0.001 to a common corner are tested.

3. 3 runs are performed for each starting point with $20,000$ steps of *Hit-and-Run* random walk.

4. The maximum point symmetry value are recorded, i.e. $max(sym(x_i, S))$

5. The maximum point symmetry value of the moving mean points are also recorded, i.e. $max(sym(\bar{\mu}_i, S))$, where $\bar{\mu}_i = \frac{1}{i}\sum_{j=1}^{i} x_j$

6. The relative symmetry value are calculated as $relsym(x_i, S) := \frac{max(sym(x_i,S))}{sym(S)}$ and $relsym(\bar{\mu}_i, S) := \frac{max(sym(\bar{\mu}_i,S))}{sym(S)}$.

The results are tabulated in table 3.3 below:

Table 3.3: *Hit-and-Run* Random Walk Analysis

Dimension	$sym(S)$	Start Point	Run ID	max $sym(x_i, S)$	rel $sym(x_i, S)$	max $sym(\bar{\mu}_i, S)$	rel $sym(\bar{\mu}_i, S)$
30×10	0.1880	origin	1	0.1296	68.9%	0.1511	80.4%
			2	0.1324	70.4%	0.1546	82.2%
			3	0.1314	69.9%	0.1447	77.0%
		$\epsilon = 0.1$	1	0.1312	69.8%	0.1501	79.8%
			2	0.1292	68.7%	0.1552	82.6%
			3	0.1249	66.4%	0.1434	76.3%
		$\epsilon = 0.01$	1	0.1132	60.2%	0.1394	74.1%
			2	0.1293	68.8%	0.1402	74.6%
			3	0.1024	54.5%	0.1332	70.1%
		$\epsilon = 0.001$	1	0.0945	50.3%	0.1123	59.7%
			2	0.0453	24.1%	0.1027	54.6%
			3	0.0924	49.1%	0.1094	58.2%
60×20	0.1306	origin	1	0.0934	71.5%	0.1102	84.4%
			2	0.0947	72.5%	0.1025	78.5%
			3	0.0841	64.4%	0.1091	83.5%
		$\epsilon = 0.1$	1	0.0798	61.1%	0.931	71.3%
			2	0.0912	69.8%	0.0945	72.4%
			3	0.0843	64.5%	0.1102	84.4%
		$\epsilon = 0.01$	1	0.0132	10.1%	0.0234	17.9%
			2	0.0145	11.1%	0.0211	16.2%

Dimension	$sym(S)$	Start Point	Run ID	max $sym(x_i, S)$	rel $sym(x_i, S)$	max $sym(\bar{\mu}_i, S)$	rel $sym(\bar{\mu}_i, S)$
			3	0.0101	7.7%	0.0145	11.1%
		$\epsilon = 0.001$	1	0.0094	7.2%	0.0099	7.6%
			2	0.0093	7.1%	0.0101	7.7%
			3	0.0035	2.7%	0.0103	7.9%
90×30	0.1090	origin	1	0.0718	65.9%	0.0877	80.5%
			2	0.0794	72.8%	0.0865	79.4%
			3	0.0737	67.6%	0.0881	80.8%
		$\epsilon = 0.1$	1	0.0423	38.8%	0.0469	43.0%
			2	0.0527	48.3%	0.0513	47.1%
			3	0.0601	55.1%	0.0613	56.2%
		$\epsilon = 0.01$	1	0.0009	0.8%	0.0011	1.0%
			2	0.0008	0.7%	0.0011	1.0%
			3	0.0009	0.8%	0.0015	1.4%
		$\epsilon = 0.001$	1	0.0006	0.6%	0.0008	0.7%
			2	0.0006	0.6%	0.0009	0.8%
			3	0.0005	0.5%	0.0009	0.8%
120×40	0.0945	origin	1	0.0681	72.1%	0.0761	80.5%
			2	0.0671	71.0%	0.0734	77.7%
			3	0.0622	65.8%	0.0741	78.4%
		$\epsilon = 0.1$	1	0.0283	29.9%	0.0311	32.9%
			2	0.0391	41.4%	0.0399	42.2%
			3	0.0293	31.0%	0.0301	31.9%
		$\epsilon = 0.01$	1	0.0007	0.7%	0.0007	0.7%
			2	0.0004	0.4%	0.0006	0.6%
			3	0.0006	0.6%	0.0006	0.6%
		$\epsilon = 0.001$	1	0.0002	0.2%	0.0003	0.3%
			2	0.0004	0.4%	0.0004	0.4%

Dimension	$sym(S)$	Start Point	Run ID	max $sym(x_i, S)$	rel $sym(x_i, S)$	max $sym(\bar{\mu}_i, S)$	rel $sym(\bar{\mu}_i, S)$
			3	0.0001	0.1%	0.0001	0.1%
150 × 50	0.0799	origin	1	0.0571	71.5%	0.0652	81.6%
			2	0.0521	65.2%	0.0635	79.5%
			3	0.0558	69.8%	0.0574	71.8%
		$\epsilon = 0.1$	1	0.0002	0.3%	0.0004	0.5%
			2	0.0003	0.4%	0.0004	0.5%
			3	0.0003	0.4%	0.0003	0.4%
		$\epsilon = 0.01$	1	0.0001	0.1%	0.0002	0.3%
			2	< 0.0001	$< 0.1\%$	0.0001	0.1%
			3	0.0001	0.1%	0.0001	0.1%
		$\epsilon = 0.001$	1	< 0.0001	$< 0.1\%$	< 0.0001	$< 0.1\%$
			2	< 0.0001	$< 0.1\%$	< 0.0001	$< 0.1\%$
			3	< 0.0001	$< 0.1\%$	< 0.0001	$< 0.1\%$

From the above results we can draw the following conclusion:

1. The role of the starting point:

   - If the starting point is not jammed in a corner, the uniform distribution can be achieved rather quickly. (illustrated by the origin case)
   - If the starting point is closer to a corner, the random walk needs to take more steps to move towards the region of relatively high symmetry. (illustrated by $\epsilon = 0.1$ case)
   - If the starting point is jammed into a corner, the *Hit-and-Run* random walk will stay in the vicinity of the corner for a long time, and needs a tremendous amount of time to move out. (illustrated by $\epsilon = 0.01$ and 0.001 cases)
   - As dimension goes up, the likelihood of moving out of a corner for the *Hit-and-run* random walk decreases.

2. The likelihood of the *Hit-and-run* random walk to enter the zone of *deep points*:

    - It is very unlikely for the *Hit-and-run* random walk to enter the zone of high relative symmetry, i.e. *deep points* that have point symmetry value of at least 90% of $sym(S)$.

    - It shows that the zone of high symmetry points for randomly generated convex bodies is extremely small.

3. Center of mass and the symmetry point of the convex set:

    - It is shown that in general the center of mass of the convex set is not the symmetry point of the set.

    - The relationship of center of mass to the symmetry point could be further explored in future work.

Chapter 4

Pre-Conditioning and Non-Euclidean Rounding

It is obvious from the *Hit-and-Run* random walk analysis in chapter 3 that when starting from a jammed corner: (i) the random walk procedure is less effective,(ii) the likelihood for random walk to move out of a poor corner is low, (iii) a lot of iterations are needed for the random walk to depart from the corner, and (vi) the situation gets worse when the convex body dimension is large.

Before we discuss the fundamental reason for the poor performance, it is beneficial to introduce first the idea of rounding, i.e. approximation of convex set S by another convex set P. If there exists a point $x \in S$ such that $\beta P \subset S - x \subset P$, we say P is a β-approximation of S. In the special case where P is an ellipsoid centered at the origin, then we say βP provides a $\frac{1}{\beta}$-rounding of S. The classic Löwner-John theorem [4] states that using ellipsoid, a $\frac{1}{n}$-approximation of any general convex body is guaranteed. The theorem states:

> The minimum-volume ellipsoid E centered at the Löwner-John center x^L which contains convex set S, provides a $\sqrt{n}$-rounding of S when S is symmetric and an n-rounding of S when S is non-symmetric.

Furthermore, Lovász and Vempala [5] prove:

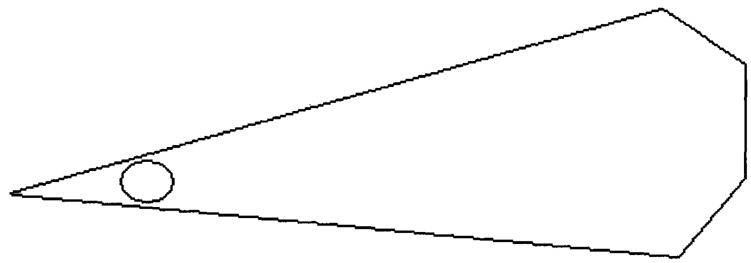

Figure 4-1: Demonstration for Wrong Sphere Rounding

Let X be a random variable uniformly distributed on a convex body $S \subset \Re^d$. Then

$$B_\Sigma\Big(\mu, \sqrt{(d+2)/d}\Big) \subset S \subset B_\Sigma\Big(\mu, \sqrt{d(d+2)}\Big),$$

where μ denotes center of mass and Σ denotes covariance matrix. $B_\Sigma(x, r)$ represents the ball centered at x with radius r in the norm $\|\cdot\|_\Sigma$ and $\|v\|_\Sigma := \sqrt{v^T \Sigma^{-1} v}$.

The lemma reiterates the existence of the n-rounding by an appropriate ellipsoid, which facilitates the explanation of the poor performance of the *Hit-and-run* random walk starting jammed in a corner. The existence of the poor corner means that the Euclidean sphere is no longer the right shape for best rounding (see Fig 4-1), rather an ellipsoid with the right shape covariance matrix $V \neq I$ should be the suitable one to consider. Thus ideally, if sampling of the random direction for the *Hit-and-Run* random walk can be drawn from the correct shape covariance matrix V of the ellipsoid, the *Hit-and-run* random walk will be as effective from a corner starting point as from a spacious central point (see Fig 4-2).

However, for a randomly generated convex body, the true ellipsoid shape covariance matrix is unknown. Some estimation must be done to get an approximate for it and this is the pre-conditioning steps performed. After the estimated covariance matrix is obtained, it is applied to the *Hit-and-run* random walk and this is called pre-conditioning.

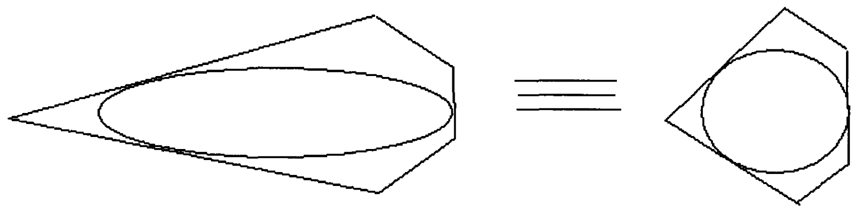

Figure 4-2: Demonstration for Correct Ellipsoid Rounding

4.1 Pre-conditioning and Conditioning

In order to estimate the true ellipsoid covariance matrix V of any randomly generated convex body, a limited number of the ordinary *Hit-and-Run* random walk steps starting from a corner are performed. From these points, the sample mean $\bar{\mu}$ and sample covariance matrix $\bar{V}$ are calculated. The sample covariance matrix is used as an approximation for the true ellipse covariance matrix. The number of pre-conditioning random walk iterations will depend on the dimension of the convex body, the degree of poorness of the corner, and how jammed in the point is inside the corner. A quantitative heuristic is presented in section 4.3 on page 44.

With the sample ellipse covariance matrix $\bar{V}$ estimated, the following is observed:
Originally without condition:

$$d \sim N(0, I)$$

Pre-Conditioning target:

$$\bar{d} \sim N(0, \bar{V})$$

The positive definite matrix $\bar{V}$ can be written as:

$$\bar{V} = M^T M$$

It is easy to see

$$d \sim N(0, I) = N\big(0, M^{-T}\bar{V}M^{-1}\big)$$

Therefore

$$d = M^{-1}\bar{d}$$

and

$$\bar{d} = Md$$

As a result, the *Hit-and-Run* random walk is conditioned as follow:

1. Generate $d \sim N(0, I)$ as described before.

2. Obtain the conditioned direction $\bar{d} = Md$.

3. Continue the min-ratio test and update the point.

With this conditioned *Hit-and-Run* random walk, getting out from the jammed corner becomes much faster and convergence to the uniform distribution is also faster.

4.2 Pre-conditioning Analysis Set up

4.2.1 Parameters Required

In order to investigate the number of pre-conditioning random walk iterations needed to produce a satisfactory sampled ellipse covariance matrix, some parameters are required.

Given:

The dimension of the convex body n.

Easily obtained:

An initial starting point near a corner whose distance from the corner is ϵ obtained using the method described in section 2.4.1 on page 23 and section 2.4.2 on page 24.

Obtained with some calculation:

Symmetry value of the convex set $sym(S)$ using formulation 2.3 on page 20.

A new parameter w called ***width*** is used to measure the size of the corner/cone and it is defined as follows:

$$w := max_{x,r}\left\{\frac{r}{\|x-c\|}\right\}$$

where c denotes the corner and $B(x,r) \subset S$ is a ball inside S centered at x with radius r.

The ***relative width*** W is then defined as

$$W := w \times \sqrt{n}$$

which measures the size of the corner "independent" of dimension. (Corner of nonnegative orthant has relative width 1.) This relative width W provides a rough scalar measure of the degree of *non-Euclidean* rounding.

Mathematically it is obtained starting from

$$Ax \leq b$$

Find a basis $\bar{B}$ such that

$$A = \begin{bmatrix} \bar{B} \\ \bar{M} \end{bmatrix} \quad b = \begin{bmatrix} \bar{b} \\ \bar{g} \end{bmatrix}$$

where

$$c = \bar{x} = \bar{B}^{-1}\bar{b} \text{ is a feasible corner and } \bar{M}\bar{x} < \bar{g}.$$

Let

$$f_i = \|\bar{B}_i\|$$

and define

$$r := -\bar{B}^{-1} f$$

in which r satisfies:

1. $\bar{B}r = -f < 0$

2. $dist\Big(r, \{x|\bar{B}_i x = 0\}\Big) = 1$

Therefore the width of the corner/cone $\{x|\bar{B}x \leq 0\}$ is

$$\frac{1}{\|r\|}$$

and the relative width is:

$$W = \frac{\sqrt{n}}{\|r\|}$$

Interestingly, the width of the corner can be adjusted to the desired value through a linear transformation which keeps the symmetry value constant. The linear transformation applied is:

$$P = I + \theta ee^T$$

and

$$x = Py$$

in which the new convex body becomes

$$\bar{B}Py \leq \bar{b}$$

$$\bar{M}Py \leq \bar{g}$$

Proposition: When θ increases, the relative width $\hat{W}$ decreases.

Proof. Let

$$\hat{f}_i = \|(\bar{B}P)_i\| = \|\bar{B}_i P\| = \|\bar{B}_i + \big(\theta(\bar{B}_i)^T e\big)e\|$$

$$\hat{r} = -(\bar{B}P)^{-1}\hat{f} = -P^{-1}\bar{B}^{-1}\hat{f} = -\left(I - \frac{\theta ee^T}{1+n\theta}\right)\bar{B}^{-1}\hat{f}$$

Thus the new relative width becomes:

$$\hat{W} = \frac{\sqrt{n}}{\|\hat{r}\|} = \frac{\sqrt{n}}{\| - (I - \frac{\theta ee^T}{1+n\theta})\bar{B}^{-1}\hat{f}\|}$$

When $\theta \gg 0$ increases, $\|\hat{f}\|$ also increases.

But

$$\left(I - \frac{\theta ee^T}{1+n\theta}\right) \rightarrow \left(I - \frac{ee^T}{n}\right)$$

gets independent of θ.

Therefore $\|\hat{r}\|$ increases as θ increases and the relative width $\hat{W}$ decreases. □

4.2.2 Performance Metric For Satisfactory Conditioning

Before the pre-conditioning investigation can be carried out, the performance metric for satisfactory conditioning must first be determined. Our primary objective for the *Hit-and-Run* random walk is to achieve uniform distribution speedily. The consequences of uniformity as suggested by Belloni and Freund are the following:

Suppose X is a uniformly distributed vector on $S \in \Re^n$ for $n \geq 2$, then

$$\Pr\left(sym(X,S) \geq \frac{sym(S)}{n}\right) \geq \left(1 - \frac{2}{n+1}\right)^n \geq \frac{1}{9}.$$

This is an excellent indicator for uniformity for the purpose of investigation. Thus in all the experiments, a ***successful run*** is defined as one that achieves uniformity within 1000 steps of post-conditioned *Hit-and-Run* random walk, where the uniformity is indicated by satisfying the above inequality relationship, i.e., the percentage of points within a factor of n of $sym(S)$ is at least 11.1%.

4.2.3 Experimental Design

The experiments are designed as follows:

1. Five random convex sets generated with dimension n = 10, 20, 30, 40 and 50.

2. For each convex set, five starting points with $\epsilon = 0.1, 0.05, 0.01, 0.005$ and 0.001 are tested.

3. For each starting point, four relative width with $\hat{W} = 0.2, 0.1, 0.05$ and 0.01 are experimented.

4. With the appropriate increment of the number of pre-conditioning *Hit-and-Run* random walk iterations in each case, the smallest number of iteration that achieves ***successful run*** in at least 7 runs out of 10 runs are recorded.

4.3 Power Law Heuristic for Pre-conditioning

The experiments are carried out and the results are tabulated in table 4.1.

Table 4.1: Pre-conditioning Investigation

Dimension	$sym(S)$	$\hat{W}$	ϵ	Iterations needed to meet performance metric
		0.2	0.1	70
		0.2	0.05	80
		0.2	0.01	100
		0.2	0.005	120
		0.2	0.001	130
		0.1	0.1	130
		0.1	0.05	170

Dimension	$sym(S)$	$\hat{W}$	ϵ	Iterations needed to meet performance metric
30×10	0.2064	0.1	0.01	220
		0.1	0.005	250
		0.1	0.001	260
		0.05	0.1	220
		0.05	0.05	250
		0.05	0.01	400
		0.05	0.005	500
		0.05	0.001	550
		0.01	0.1	700
		0.01	0.05	1100
		0.01	0.01	1700
		0.01	0.005	2000
		0.01	0.001	2200
60×20	0.1291	0.2	0.1	80
		0.2	0.05	90
		0.2	0.01	130
		0.2	0.005	140
		0.2	0.001	170
		0.1	0.1	140
		0.1	0.05	150
		0.1	0.01	280
		0.1	0.005	300
		0.1	0.001	350
		0.05	0.1	230
		0.05	0.05	250
		0.05	0.01	420
		0.05	0.005	540
		0.05	0.001	580
		0.01	0.1	750

Dimension	$sym(S)$	$\hat{W}$	ϵ	Iterations needed to meet performance metric
		0.01	0.05	1200
		0.01	0.01	1900
		0.01	0.005	2350
		0.01	0.001	2900
		0.2	0.1	250
		0.2	0.05	300
		0.2	0.01	390
		0.2	0.005	430
		0.2	0.001	550
		0.1	0.1	440
		0.1	0.05	520
		0.1	0.01	800
		0.1	0.005	880
90×30	0.1071	0.1	0.001	1020
		0.05	0.1	760
		0.05	0.05	900
		0.05	0.01	1400
		0.05	0.005	1600
		0.05	0.001	2000
		0.01	0.1	2700
		0.01	0.05	3400
		0.01	0.01	5000
		0.01	0.005	5900
		0.01	0.001	8500
		0.2	0.1	700
		0.2	0.05	820
		0.2	0.01	1100
		0.2	0.005	1300
		0.2	0.001	1800

Dimension	$sym(S)$	$\hat{W}$	ϵ	Iterations needed to meet performance metric
120×40	0.0978	0.1	0.1	1250
		0.1	0.05	1500
		0.1	0.01	2100
		0.1	0.005	2500
		0.1	0.001	3350
		0.05	0.1	2050
		0.05	0.05	2400
		0.05	0.01	3350
		0.05	0.005	4000
		0.05	0.001	5600
		0.01	0.1	7200
		0.01	0.05	8900
		0.01	0.01	12000
		0.01	0.005	14000
		0.01	0.001	19000
150×50	0.0825	0.2	0.1	1500
		0.2	0.05	1700
		0.2	0.01	2500
		0.2	0.005	2800
		0.2	0.001	3500
		0.1	0.1	2900
		0.1	0.05	3500
		0.1	0.01	4500
		0.1	0.005	5000
		0.1	0.001	5500
		0.05	0.1	4500
		0.05	0.05	5000
		0.05	0.01	7200
		0.05	0.005	8200

Dimension	$sym(S)$	$\hat{W}$	ϵ	Iterations needed to meet performance metric
		0.05	0.001	8800
		0.01	0.1	15000
		0.01	0.05	21000
		0.01	0.01	28000
		0.01	0.005	32000
		0.01	0.001	39000

It is of great interest to develop a quantitative heuristic that will predict the number of pre-conditioning iterations needed given the dimension of the convex body, the relative width of the corner and the distance of the starting point from the corner. With all the results obtained, a multiple linear regression is performed and the following power law relationships are obtained.

Given:

n : The dimension of the convex body.

$\hat{W}$: The relative width of the corner.

ϵ : The distance of the starting point to the corner

Supplement:

sym : The symmetry value of the convex body $sym(S)$.

Target:

$Iter$: The pre-conditioning steps required.

Law 1 with $R^2 = 0.973$:

$$Iter = 3.874e^{0.08n}\hat{W}^{-0.837}\epsilon^{-0.200}$$

Law 2 with $R^2 = 0.991$:

$$Iter = 91.713e^{0.126n}\hat{W}^{-0.837}\epsilon^{-0.200}(sym)^{2.11}$$

Law 3 with $R^2 = 0.906$:

$$Iter = 0.0914n^{1.896}\hat{W}^{-0.837}\epsilon^{-0.200}$$

Law 4 with $R^2 = 0.962$:

$$Iter = 2.762n^{7.295}\hat{W}^{-0.837}\epsilon^{-0.200}(sym)^{9.836}$$

All the above laws exhibit excellent regression fit to the data, the T stats shown in Table 4.2 are all significantly large to indicate a strong fit. All laws see identical power coefficient for relative width $\hat{W}$ and distance from a corner ϵ, while the first and third ones do not include the variable $sym(S)$. In **Law** 1 **and** 2, exponential constants for dimension are obtained, with **Law** 2's exponential coefficient slightly larger. We know from [6] that the increase in number of iterations for *Hit-and-Run* random walk should be at most polynomial in the dimension. The reason of higher R-square value for the two laws involving exponential coefficient to dimension is probably due to a better regional fit that does not represent the entire trend correctly. Therefore **Law** 3 **and Law** 4, where dimension is in polynomial degree are considered better. Furthermore, we know that the variables $\hat{W}$, ϵ and n are independent of each other, however $sym(S)$ is dependent on n. As n increases, $sym(S)$ decreases. Therefore although **Law** 4 provides a slightly better fit in terms of R-square value as compare to **Law** 3, care must be taken when applying outside the dimension range tested. Furthermore, in real life applications, the calculation of $sym(S)$ is not always possible or practicable, thus **Law** 3 is definitely the better and handy one to use in real applications.

From these laws, one can easily observe that as dimension n goes up, the pre-conditioning

Table 4.2: T stats for multiple linear regression

Law ID	Dimension n	Relative Width $\hat{W}$	Distance ϵ	$sym(S)$
1	44.4	-36.0	-12.7	N.A.
2	36.3	-61.1	-21.6	13.5
3	22.5	-19.4	-6.9	N.A.
4	15.9	-30.4	-10.7	11.9

iteration numbers increase very fast. The effect of relative width and thus the degree of *non-Euclidean* rounding has an intuitively stronger influence on the iteration number needed as compare to the distance of the point from the corner. In overall, a large number of iterations are expected in high dimension, *non-Euclidean* cases and powerful computers are needed to fulfill the task.

4.4 Use of Power Law Heuristic in Real Life Application

In real life applications, we are usually given a convex body $S := \{x \in \Re^n | Ax \leq b\}$, where $A \in \Re^{m \times n}$, and a feasible initial point x^0 such that $Ax^0 < b$. In order to use the above power law heuristic, with the dimension n already given, one has to obtain the relative width $\hat{W}$ and distance ϵ. They can be obtained by the following method:

1. : Identify a basis A_B such that $\|A_B x^0 - b_B\|$ is small.

2. : Obtain the corner x^c by solving $A_B x^c = b_B$.

3. : The relative width $\hat{W}$ can then be calculated using the method described in section 4.2.1 on page 40.

4. : The distance ϵ can then be calculated using methods described in section 2.4.2 on page 24.

Chapter 5

Conclusion

5.1 Summary

In this thesis, a series of experiments are used to analyze and investigate the effectiveness of the *Hit-and-Run* random walk to compute *deep points* in a given polyhedron. A deep point is one that has high symmetry function value, where symmetry measures how symmetric the set is with respect to that point. The *Hit-and-run* random walk iteratively moves from point to point. In each iteration a random direction is first selected and a line is drawn along that direction and passes through the current point. A new random point is then picked on the chord located within the convex set. All the convex sets in these analysis are assumed to be bounded with non-empty interior and this is achieved by performing the boundness test designed.

The original *Hit-and-run* random walk draws the random direction from the Euclidean sphere and results have shown that when starting from the relatively spacious origin, the unconditioned *Hit-and-run* random walk performs satisfactorily. Though the likelihood of entering the zone of high symmetry is very low, and the center of mass of the uniform distribution are in general different from the symmetry point, the random walk is proven to be effective in obtaining uniform distributions on the convex body. The above claims are supported strongly by the fact that all the probabilistic properties involving the symmetry function established by Belloni and Freund are experimentally verified by the *Hit-and-Run* random walk routine.

However, it is of great concern that when starting from a poor corner of a convex set with *non-Euclidean* rounding, the *Hit-and-Run* random walk requires a lot of steps and the situation gets worse when the dimension of the convex body increases. In order to improve the performance of the random walk in such cases, pre-conditioning is applied to the procedure. Instead of drawing the random direction from the Euclidean sphere, the estimated ellipsoid covariance matrix is used for picking the random direction. The approximation of the covariance matrix is done through pre-conditioning steps of the original *Hit-and-Run* random walk, where the sampled covariance matrix of those pre-conditioning points are taken to be the estimation. When the pre-conditioned *Hit-and-Run* random walk is performed, a marked improvement is observed in the speed of convergence to uniformity.

In order to quantify the number of pre-conditioning steps needed, experiments are performed and some heuristics are obtained in the form of power laws. The most practical one is $Iter = 0.0914 n^{1.896} \hat{W}^{-0.837} \epsilon^{-0.200}$, where $Iter$ represents the number of pre-conditioning steps needed, n means the dimension of the convex set, $\hat{W}$ measures the size of the corner and ϵ indicates the relative distance of the point from the corner. Using this empirical relationship, one can roughly gauge the number of pre-conditioning steps needed so that uniformity can be achieved within 1000 iterations of conditioned *Hit-and-Run* random walk.

5.2 Future Work

There are some interesting points relating to the topic worthy of future in-depth research. Some of them are listed below:

1. When performing the *Hit-and-Ran* random walk, a point is picked from the entire chord length feasible to the convex body. If instead of taking the whole chord length at every iteration, a gradual reduction in the length is applied, how will the convergence to uniformity affected?

2. In performing the *Hit-and-Ran* random walk, a large amount of computation is needed especially in larger dimension. This provides opportunity for parallel computing, in which the solving speed could hopefully be enhanced.

Bibliography

[1] Alexandre Belloni and Robert M. Freund. On the symmetry function of a convex set. *Mathematical Programming.* to appear.

[2] Santosh Vempala. Geometric random walks: A survey. *Combinatorial and Computational Geometry, MSRI Publications*, 52:573–612, 2005.

[3] Alexandre Belloni and Robert M. Freund. Projective pre-conditioners for improving the behavior of a homogeneous conic linear system. *MIT Operations Research Center Working Paper OR-375-05.* Submitted to Mathematical Programming.

[4] F. John. Extremum problems with inequalities as subsidiary conditions. *in Studies and Essays, Presented to R. Courant on His 60th Birthday, Interscience, New York*, pages 187–204, 1948.

[5] László Lovász and Santosh Vempala. The geometry of logconcave functions and an $O^*(n^3)$ sampling algorithm. *Microsoft Technical Report.*

[6] László Lovász and Santosh Vempala. Hit-and-run from a corner. In *Proc. of the 36th ACM Symposium on the Theory of Computing (STOC '04)*, Chicago, 2004.