LOWER BOUNDS OF AREAS OF CONVEX COVERS FOR CLOSED UNIT ARCS

Thesis submitted at the University of Leicester
in partial fulfillment of the requirements for
the degree of Doctor of Philosophy of Mathematics

by

Sittichoke Som-am
Department of Mathematics
University of Leicester

2020

Contents

Declaration **ix**

Acknowledgment **x**

Abstract **xi**

1 Introduction **1**

Introduction **1**

1.1 Introduction and literature review 1

1.1.1 Lebesgue Universal covering problem 1

1.1.2 Lost in a Forest problem 3

1.1.3 Moser's worm problem 5

1.1.4 Covex cover 6

1.1.5 Convex arcs 10

1.1.6 Closed arcs 12

1.2 Structure and Results 18

2 Numerical minimization problem of Lipschitz function **19**

2.1 Basic definitions and theorems 19

2.2 Methodology 24

2.2.1 Quick non-rigorous method 24

2.2.2 Naive method with fixed grid 25

2.2.3 Grid-search algorithm with variable grid 26

2.2.4 Box-search algorithm 31

2.2.5 Heap sort algorithm 41

2.2.6 Modified heap sort 42

2.2.7 Discussion 49

3 The lower bound is 0.0975 **51**

3.1 Numerical method to find the best rectangle 51

3.2 Geometric analysis . . . 53
3.3 Computational results . . . 63
3.4 Main Theorem . . . 65

4 Systematic search **69**
4.1 Centrally symmetric objects . . . 69
4.1.1 Numerical method to find the bound for two centrally symmetric objects . . . 70
4.1.2 Numerical method to find the bound for three centrally symmetric objects . . . 71
4.1.3 Lemma 3.2 cannot be extended to $n \geq 3$ . . . 72
4.2 A systematic search for shape . . . 73
4.2.1 Numerical results for 2 objects . . . 74
4.2.2 Numerical results for 3 objects . . . 79

5 The lower bound is 0.1 **89**
5.1 Geometric analysis . . . 89
5.2 Computational results . . . 100
5.3 Main theorem . . . 100

6 Conclusions and Conjecture **103**
6.1 Conclusions . . . 103
6.2 Conjecture . . . 105

List of Figures

1.1 Pal's cover. 2
1.2 The cutting off area by Sprague and Hensen. 2
1.3 The smallest convex hull of a circle, a regular pentagon, and an equilateral triangle with diameter 1. 3
1.4 The smallest convex hull of a circle and regular Reuleaux polygons with $3, 5, 7$, and 9 sides. 4
1.5 Broadworm. 5
1.6 The Besicovitch path with $AB = BC = CD = \dfrac{\sqrt{27}}{\sqrt{28}}$. 5
1.7 Norwood and Poole's cover. 6
1.8 Wetzel's sector. 7
1.9 Gerriets's cover. 7
1.10 Norwood, Poole and Laidacker's cover. 8
1.11 Wang's cover 8
1.12 a $30°$ circular sector of unit radius. 9
1.13 a staple with leg $\frac{1}{4}$ and base $\frac{1}{2}$. 9
1.14 Fergoson's conjecture. 10
1.15 The small configuration found by Tanadkithirun. 10
1.16 The isosceles right triangle clipped at height 0.47140. 11
1.17 The cover for convex unit arcs with area 0.24656. 11
1.18 The cover for convex unit arcs with area 0.2464. 12
1.19 The cover for convex unit arcs. 12
1.20 The smallest rectangular cover. 13
1.21 The truncated regtangular cover. 13
1.22 The pentagonal cover. 14
1.23 The curvilinear rectangle. 14
1.24 The current closed arc cover which $t \approx 0.148$, $s \approx 0.142$ and $s_2 \approx 0.0617$. . . . 15
1.25 The smallest configuration of a circle with perimeter 1 and a half unit line segment. 16
1.26 The smallest configuration of a circle with perimeter 1 and $u \times v$ rectangle. 16

1.27 The smallest configuration of a circle with perimeter 1 and curvilinear rectangle. 16
1.28 The best configuration with area 0.0970439. 17

2.1 The bound B. 24
2.2 Grid points. 27
2.3 Grid points in 1st iteration for Example 2.31. 28
2.4 Grid points in 2nd iteration for Example 2.31. 29
2.5 Grid points in iteration 3 for Example 2.31. 30
2.6 The center point of box. 32
2.7 1st iteration of BSA in Example 2.35. 33
2.8 2nd iteration of BSA in Example 2.35. 33
2.9 3rd iteration of BSA in Example 2.35. 34
2.10 4th iteration of BSA in Example 2.35. 35
2.11 5th iteration of BSA in Example 2.35. 35
2.12 6th iteration of BSA in Example 2.35. 36
2.13 7th iteration of BSA in Example 2.35. 37
2.14 8th iteration of BSA in Example 2.35. 37
2.15 9th iteration of BSA in Example 2.35. 38
2.16 Graph of relationship between time and bound (log-log scale) in Table 2.3. . . 39
2.17 The complete binary tree with $n = 7$. 41
2.18 Step 1: `B[1]` is generated into `B[2]` and `B[3]` with `B[2] < B[3]`. 43
2.19 Step 2: `B[2]` is generated into `B[4]` and `B[5]` with `B[4] < B[5]`. 44
2.20 Step 3: `B[3]` is generated into `B[6]` and `B[7]` with `B[6]` $<$ `B[7]`. 44
2.21 sorting step in Step 3 45
2.22 Step 4: `B[4]` is generated into `B[8]` and `B[9]` with `B[9]` $<$ `B[8]`. 45
2.23 sorting step in Step 4 46
2.24 Step 5: `B[5]` is generated into `B[10]` and `B[11]` with `B[10]` $<$ `B[11]`. 46
2.25 sorting step in Step 5. 47
2.27 sorting step in Step 6 47
2.26 Step 6: `B[6]` is generated into `B[12]` and `B[13]` with `B[13] < B[12]`. 48
2.28 Graph of relationship between time and bound (log-log scale) in Table 2.3. . . 49

3.1 Optimal configuration for C_1, R, and T 52
3.2 The configuration X which depends on $x_1, y_1, x_2, y_2, \theta$ 54
3.3 $\mathcal{A}(X)$ is bounded by the area of $EKGH, S(F)/2$ and the semicircle C_1 . . . 55
3.4 $\mathcal{A}(X)$ is bounded by the area of convex hull of R', R_1, R_2, R_3, R_4 and P' . . 56
3.5 The ratio between $g(x_1)$ and x_1 when $x_1 \to +\infty$ 57
3.6 The ratio between $g(x_2)$ and x_2 when $x_2 \to +\infty$ 58

3.7 The ratio between $g(y_2)$ and y_2 when $y_2 \to +\infty$ . . . 58
3.8 The three triangles which increase the area of the convex hull with ϵ_2 . . . 60
3.9 The longest diameter between R_2 and F . . . 60
3.10 Polygon $\mathcal{H}(R, F, T)$ adjacent T_1, T_2, T_3 . . . 61
3.11 Six triangles which T rotated by angle ϵ_5 . . . 61
3.12 The maximum slope for C_1 is 0.1624 . . . 62
3.13 The maximum slope for C_2 is 0.20418 . . . 62
3.14 The maximum slope for C_3 is 0.32483 . . . 62
3.15 The maximum slope for C_4 is 0.38256 . . . 63
3.16 The maximum slope for C_5 is 0.03517 . . . 63
3.17 The convex hull of the configuration of the minimum area with 0.097627 acquired from the BSA . . . 66
3.18 The configuration of R and R' . . . 67

4.1 The maximum area of convex hull of A_1 and A_2' with the convex hull area of 0.0966693 . . . 71
4.2 The maximum area of convex hull of B_1, B_2 and B_3 which has the area of 0.100685 . . . 72
4.3 the optimal configuration for 3 centrally symmetric objects with the convex hull area of 0.100465 . . . 73
4.4 The optimal configuration of 2+3 points with the convex hull area of 0.072169. 76
4.5 The optimal configuration of 3+3 points with the convex hull area of 0.072375. 76
4.6 The optimal configuration of 3+4 points with the convex hull area of 0.085377. 76
4.7 The optimal configuration of 4+4 points with the convex hull area of 0.085377. 77
4.8 The optimal configuration of 3+5 points with the convex hull area of 0.087902. 77
4.9 The optimal configuration of 5+5 points with the convex hull area of 0.087902. 77
4.10 The optimal configuration of $9 + 19$ points with the convex hull area of 0.095790. . . . 78
4.11 The optimal configuration of $9 + 50$ points with the convex hull area of 0.0966051. . . . 78
4.12 The optimal configuration of $11 + 50$ points with the convex hull area of 0.0966053. . . . 79
4.13 The optimal configuration of $2 + 3 + 3$ points with the convex hull area of 0.072169. . . . 81
4.14 The optimal configuration of $3 + 3 + 3$ points with the convex hull area of 0.072419. . . . 81
4.15 The optimal configuration of $2 + 3 + 4$ points with the convex hull area of 0.087867. . . . 82

4.16 The optimal configuration of $3+3+4$ points with the convex hull area of 0.087887. . . . 82
4.17 The optimal configuration of $10+10+10$ points with the convex hull area of 0.093546. . . . 83
4.18 The optimal configuration of $2+4+50$ points with the convex hull area of 0.100403. . . . 83
4.19 The optimal configuration of $4+4+50$ points with the convex hull area of 0.100407. . . . 84
4.20 The optimal configuration of $7+7+50$ points with the convex hull area of 0.100417. . . . 84
4.21 The optimal configuration of 500-gon, a line and a triangle (green) with the convex hull area of 0.097043. . . . 85
4.22 The optimal configuration of 500-gon, a line and a quadrilateral (green) with the convex hull area of 0.1003. . . . 85
4.23 The optimal configuration of 500-gon, a line and a pentagon (green) with the convex hull area of 0.100304. . . . 86
4.24 The optimal configuration of 500-gon, a line and a hexagon (green) with the convex hull area of 0.100374. . . . 86
4.25 The optimal configuration of 500-gon, a line and 7-gon (green) with the convex hull area of 0.100386. . . . 87
4.26 The optimal configuration of 500-gon, a line and 8-gon (green) with the convex hull area of 0.100390. . . . 87
4.27 The optimal configuration of 500-gon, a line and 9-gon (green) with the convex hull area of 0.100418. . . . 88
4.28 The optimal configuration of 500-gon, a line and 10-gon (green) with the convex hull area of 0.100473. . . . 88

5.1 The configuration X. . . . 90
5.2 $EKL_1L_2$and l. . . . 91
5.3 The configuration of y_1^*, y_2^* and R. . . . 92
5.4 The line shows the optimum possible position of F, R, L. . . . 92
5.5 The ratio between $g(x_1)$ and $x_1$when $x_1 \to +\infty$. . . . 93
5.6 The ratio between $g(y_1)$ and y_1 when $y_1 \to +\infty$. . . . 94
5.7 The ratio between $g(x_2)$ and x_2 when $x_2 \to +\infty$. . . . 94
5.8 the ratio between $g(y_2)$ and y_2 when $y_2 \to +\infty$. . . . 95
5.9 The longest distance between R_2 and F. . . . 96
5.11 Four triangles with L rotated by angle ϵ_5. . . . 97
5.10 Polygon $\mathcal{H}(R, F, L)$ adjacent to L_1,L_2. . . . 97

5.12 The maximum slope for C_1 is 0.2535. 98
5.13 The maximum slope for C_2 is 0.4166. 98
5.14 The maximum slope for C_3 is 0.3482. 99
5.15 The maximum slope for C_4 is 0.4191. 99
5.16 The maximum slope for C_5 is 0.0256. 99
5.17 The convex hull of the configuration of the minimum area with 0.10044 acquired from the BSA. 101
5.18 The configuration of R and R'. 102

6.1 The convex hull of the configuration with the minimum area of 0.097627 acquired from the BSA. 104
6.2 The convex hull of the configuration with the minimum area of 0.10044 acquired from the BSA. 105
6.3 The optimal configuration of γ (green) and X (blue). 106
6.4 The conjecture' s cover for closed unit arcs and its area is about 0.104597 . 108
6.5 The configuration of cover γ and triangle X 108
6.6 The configuration of cover γ and quadrilateral X 109
6.7 The configuration of cover γ and pentagon X 109
6.8 The configuration of cover γ and hexagon X 110
6.9 The configuration of cover γ and 7-gon X 110
6.10 The configuration of cover γ and 8-gon X 111
6.11 The configuration of cover γ and 9-gon X 111
6.12 The configuration of cover γ and 10-gon X 112
6.13 The configuration of cover γ and 20-gon X 112
6.14 The graph of percentage of r and n 113
6.15 The graph of percentage of r and n 131

List of Tables

2.1 Grid search results in Example 2.32. 30
2.2 Grid search results in Example 2.33. 31
2.3 The computation time of Box-search method when the bound changes. . . . 39
2.4 The bound by MHS 49

3.1 Table shows the results for MHS. 65

4.1 The numerical series for 2 objects. 75
4.2 The numerical series for 3 objects 80
4.3 The numerical series for 3 objects when 2 objects are fixed. 85

6.1 The table of percentage of r and n 113
6.2 The table of percentage of r and n 131

Declaration

All sentences or passages quoted in this project dissertation from other people's work have been specifically acknowledged by clear cross referencing to author, work and page(s). I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this module and the degree examination as a whole.

Name:

Signed:

Date:

Acknowledgment

I would like to thank my research supervisor, Dr. Bogdan Grechuk, for giving me the opportunity to do research and helping me to solve many problems in this research. It was a great privilege and honor to work and study under his guidance. I would also like to thanks him for his friendship, empathy and great sense of humor.

I would like to extend my sincere gratitude to the exceptional School of Mathematics and Actuarial Science which is a part of the College of Science and Engineering, University of Leicester. I would like to give special thanks to Prof. Ruslan Davidchack and Dr. Oleg Karpenkov who gave me many suggestions in my thesis. Thank you to Charlotte Langley who is a programme administrator for helping me to submit everything.

In addition I thank my sponsor Royal Thai Government scholarship for their support of everything in UK.

Last of all, I would like to thank my family, my friends and everyone else who helped contribute to this project especially my best friend Asif Essak who helped me to correct my thesis.

Abstract

Moser's worm problem is the unsolved problem in geometry which asks for the minimal area of a region S on the plane which can cover all curves of unit length, assuming that curves may be rotated and translated to fit inside the region. This thesis studies a version of this problem when region S is *convex* and unit curves to be covered are *closed*. For example, region S should be able to cover a circle of length 1, a square of side length $1/4$, a line interval of length $1/2$, and so on. An example of such cover S is the circle of diameter 1, whose area is about 0.7854, but the problem is to find S with minimal area. Recently, Wichiramala constructed a hexagon with this property and area about 0.11023, and this is the current record. On the other hand, it is known that the area of S cannot be less than 0.096694.

In this work, we improve the lower bound for area of convex cover S for closed unit arcs from 0.096694 to 0.0975 and then to 0.1 by finding the smallest areas of convex hulls of three carefully chosen closed unit arcs. We do this by combining geometric arguments with numerical methods such as the box-search algorithm. First, we show that the minimal area of a convex hull of a circle with radius $\dfrac{1}{2\pi}$, a rectangle with perimeter 1 and the equilateral triangle of side $\dfrac{1}{3}$ is at least 0.0975. Next, we perform a systematic search for triples of closed unit arcs which leads to an even better bound. The result of our search suggests to consider circle with radius $\dfrac{1}{2\pi}$, the rectangle with sides 0.1727 and 0.3273, and the line of length $\frac{1}{2}$. As the main result of this work, we prove that the minimal area of a convex hull of these three closed unit arcs is at least 0.1. This gives 0.1 as the lower bound for the area of S.

Chapter 1

Introduction

1.1 Introduction and literature review

In 1914, Lebesgue [34] asked what is the smallest set S in the plane which contains a congruent copy of each set C of diameter 1. The diameter of a set is the least upper bound of the distances between all pairs of points in the set. This problem is known as "Lebesgue Universal covering problem", and it is still unsolved. In 1966, Leo Moser [36] asked the same question for curves C of length 1, which is known as "Moser's worm problem", and also remains open today. There are many variants of these problems, such as, for example:

1. restrict the allowed cover sets S (e.g. S should be triangle, rectangle, convex, etc.).
2. restrict the sets C to be covered (e.g. closed curves, convex sets, etc.).

Below we give a literature review of the progress on the Lebesgue Universal covering problem, Moser's worm problem, their variants, and related problems.

1.1.1 Lebesgue Universal covering problem

The Lebesgue Universal covering problem is called "Universal cover's problem" In 1910, Jung [29] showed that a circle of radius $\dfrac{1}{\sqrt{3}}$ is a universal cover, whose area is 1.047, and the unit square is also a universal cover with area 1. In 1920, Pal [41] demonstrated that a clipped regular hexagon contained the unit circle in Figure 1.1, which has an area of 0.8454. In 1936, Sprague [49] reduced the Pal's cover by cutting off two circular arcs at another corner decreasing the upper bound to 0.8441377. Later, Hansen [24] improved the upper bound by a small amount of 1.8738×10^{-11} by chopping two clipped corners from Pal's cover in 1992, see Figure 1.2. Moreover, Duff [8] constructed the non-convex universal cover by reducing Sprague's cover, which has an area of 0.84413570. In 2015, Baez, Bagdasaryan, and Gibbs [2] removed Hansen's cover to be a new upper bound by creating a Java applet, which gives an area of 0.8441153. Gibbs [16] applied the same method to improve the upper bound to 0.844094 in 2018 which is a current upper bound.

Furthermore, Burr [40] added this problem to the unsolved problems for amateurs, meaning that this problem is easy to understand but difficult to solve.

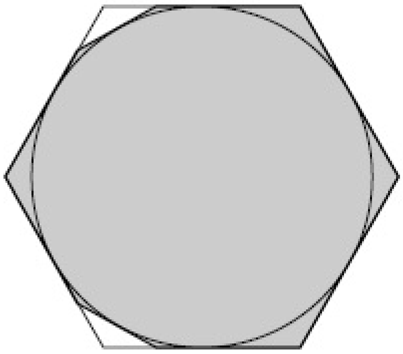

Figure 1.1: Pal's cover.

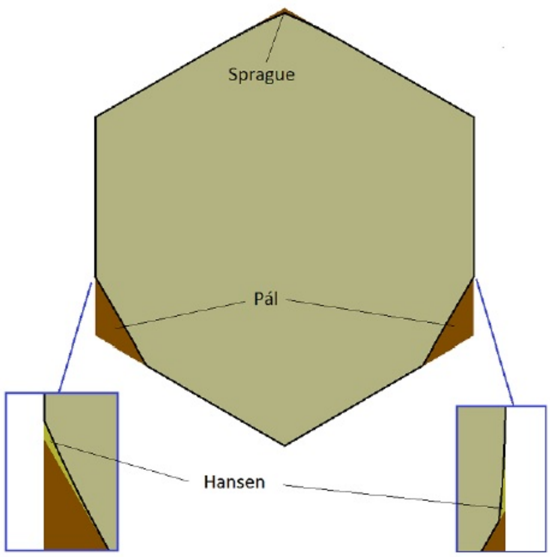

Figure 1.2: The cutting off area by Sprague and Hensen.

On the other hand, progress has also been made on the lower bound problem. Firstly, Pal [41] found the smallest convex hull of a circle and a triangle of diameter 1 which has an area of 0.8257. Later, Elekes [10] determined the minimum area of the convex hull of a circle and all regular 3^i gons with diameter 1, which increased the bound to 0.8271. In 2005, Brass and Sharifi [5] improved the lower bound to 0.832 which is the current optimal lower bound by using the computational method to find the minimum area of the convex hull of a circle, a regular pentagon, and an equilateral triangle with diameter 1, see Figure 1.3. In 2018, Gibbs [16] used simulated annealing method to find the smallest area of

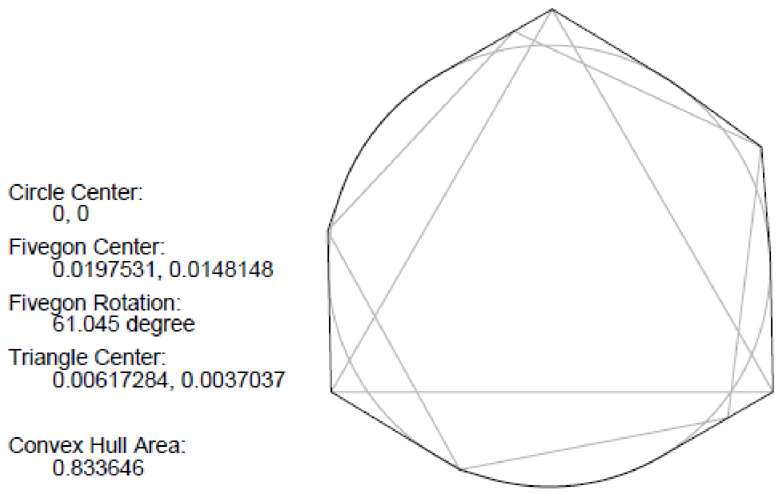

Figure 1.3: The smallest convex hull of a circle, a regular pentagon, and an equilateral triangle with diameter 1.

convex hull of a circle and regular Reuleaux polygons with $3, 5, 7$, and 9 sides, see Figure 1.4. Its area was 0.836991. However, he did not give a rigorous proof for the lower bound. To summarise, if α_1 is the smallest area of convex cover for this problem, then we have $0.832 \leq \alpha_1 \leq 0.844094$.

We can see that the upper bound has continuously reduced for this problem by cutting off the previous covers, whereas the lower bound was not improved for a long time, slowing the challenging nature of the problem.

1.1.2 Lost in a Forest problem

In 1956, R. Bellman asked what is the shortest path which a hiker who is lost in a forest should follow to escape the forest if he knows the shape and dimensions of the forest. It is equivalent to a swimmer lost in a sea [40]. This problem is still open. Let a forest F be a closed and convex region. A path γ is called an *escape path* of F if γ cannot be fitted in F without intersecting boundary of F. The *escape length* $l(F)$ of F is the length of escape path which is the shortest for F. For example, if F is bounded of diameter δ, a line of length δ is an escape path and $l(F) \leq \delta$. In general, $l(F) \leq l(\gamma)$ and it is optimal if $l(F) = l(\gamma)$, where $l(\gamma)$ is a length of arc γ. Now, several mathematicians have found solutions for specifics types of forests, as follow

- In 1955, Gross [23] showed that the shortest escape path for circular disk or square is diameter.
- In 1957, Isbell [26] showed that the shortest escape path for a half-plane forest and

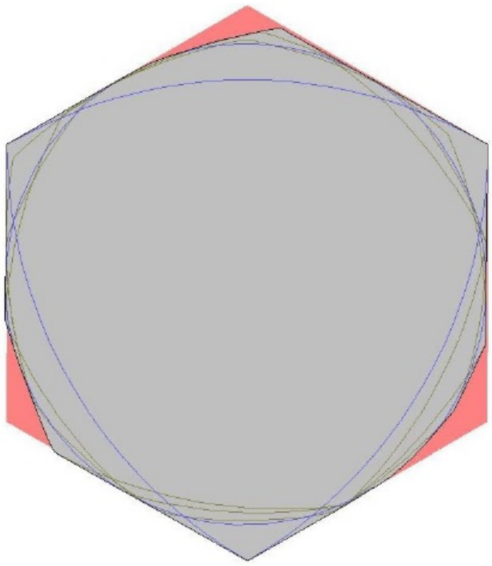

Figure 1.4: The smallest convex hull of a circle and regular Reuleaux polygons with $3, 5, 7,$ and 9 sides.

the distance from hiker and edge is d is a straight line of length d and then move in a circular path with radius d and centered at starting point.

- If the forest is an unbounded region which has the infinite strip between two parallel lines with distance d, then what is the best path? In 1961, Zalgaller [59] found the path which escaped the unbounded forest and Schaer [46] also found the same solution in 1968 but he did not know that Zalgaller found the solution. This path is called "broadworm". It consists of four straight lines and two circular curves with $\alpha \approx 0.290046$, $\gamma \approx 0.480931$, $\beta \approx 0.318888$ and $a_0 \approx 1.043590$ see Figure 1.5. Thus, the escape length is $b_0 d$, where $b_0 \approx 2.278294$. In addition, Adhikari and Pitman [1] discovered this path in 1989 and called "caliper".

- In 1963, Graham [20] asked what is the shortest escape path for an equilateral triangle of side 1. Besicovitch [4] conjectured the solution in 1966. It was composed of three equal line segments, see Figure 1.6.

- In 1973, Poole and Gerriets [15] showed that the shortest escape path for a $60°$ rhombus with a longer diagonal of L is the line of length L.

- In 1974, Wetzel [54] showed that the escape length of a circular sector with angle θ and radius $r = (L/2)\csc(\theta/2)$, where L is diameter of this sector, is L for $\theta \geq 60°$.

- Let X be a compact and convex set. If (i) X contains $AB=$ diameter of X and (ii) a $60°$ rhombus with a longer diagonal of AB is contained in X, X is called *rhombus diametral* set. In 2004, Finch and Wetzel [12] showed that the escape length of a *rhombus diametral* is its diameter. In particular, every regular n-gon for $n > 3$ is a rhombus diametral. So, the shortest escape path for the regular n-gon for $n > 3$ is its diameter.

- In 2006, Besicovitch's conjecture was proved by Coulton and Movshovich [7] and they showed that this path is the shortest path for some isosceles triangles [37]. Ward [53] gave a good review for this problem in 2008.
- In 2016, Gibbs [18] [17] used numerical method to find the shortest escape path for convex polygons and isosceles triangle.

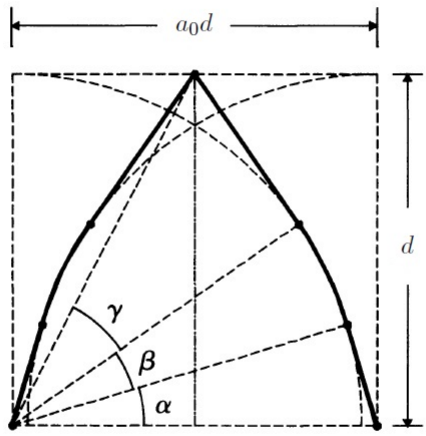

Figure 1.5: Broadworm.

Figure 1.6: The Besicovitch path with $AB = BC = CD = \dfrac{\sqrt{27}}{\sqrt{28}}$.

Moreover, Williams [58] added Lost in a Forest problem to his list of unsolved million dollar math problems. This means that this problem is of high significance in mathematics.

1.1.3 Moser's worm problem

In 1966, Leo Moser [36] asked the question "What is the smallest set which accumulates every unit arc in $\mathbb{R}^2$". This question is called "Moser's worm problem". A unit arc or worm is a continuous rectifiable curve of unit length [44]. It is easy to see that Moser's worm problem is a Universal cover problem for unit arcs. Although this problem is currently unsolved, mathematicians try to find a small cover and show that a strategy exists so that this cover contains all unit arcs. For example, a disk with radius 1 and area π is a cover for this problem. Let α be the smallest area of convex cover for unit arcs. The current upper bound of minimal area of S is 0.260437 which is created by Norwood and Poole [38]. It was constructed by laying the unit segment on the X-axis which the midpoint was at $(0, 0)$. By numerical method, $T = (0, 0.01528)$ is optimal point which attains the minimum area. Next, point T was fixed on Y-axis and an arc was drawn through T. The endpoints of the parabola which completed the top half of the cover using B'. The bottom half was created B reflection in the X-axis of the top half, see Figure 1.7. Note that S is non-convex. From the lower bound perspective, we only know it is strictly positive [35]. Hence, $0 < \alpha \leq 0.260437$.

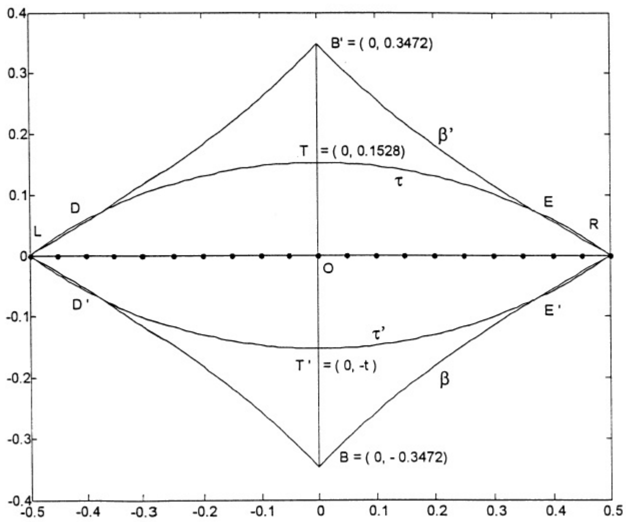

Figure 1.7: Norwood and Poole's cover.

1.1.4 Covex cover

The current smallest cover for this problem is a non-convex set. Thus, an interesting problem is to restrict the cover to be convex. Laidecker and Poole [33] used Blachke Selection Theorem to show the existence of solution for convex cover, i.e., there exist the smallest convex cover which covers every unit arc . As this problem is difficult, a complete solution has not been found. Therefore, we only know the bound of the area for convex covers of unit arcs, i.e. for an α_0 which is the smallest area of convex cover for unit arcs, there exists $a, b \in \mathbb{R}$ such that $a \leq \alpha_0 \leq b$, where a and b are said to be the lower and upper bounds, respectively. We will first discuss the upper bound b. Clearly, the circle with radius $\frac{1}{2}$ which has area of 0.78539 can cover every unit arcs by translating the mid point of arc to the center. Mower's worm problem can be reformulated as a problem to find the convex forest of largest area for which the shortest escape path has length 1. It follows that this region is a cover for unit arcs. For example, if we scale the Besicovitch path to 1, we obtain that an equilateral triangle of side $\frac{\sqrt{28}}{\sqrt{27}}$ is a cover for unit arcs. Its area is 0.44905. Hence, the escape path which is of length one for a specific shape gives an upper bound for Moser's worm problem. In 1975, A. Meir [55] showed that every unit curve can be place inside a semicircle with diameter 1 which has area of 0.3927. Later, Schare and Wetzel [45] found the smallest rectangle and triangle cover which are the square with diagonal 1 and an equilateral triangle of side 1. However, the area of each cover is larger than Meir's cover. In 1973, Wetzel [54] proved that the circular sector with radius r and vertex angle θ contains all unit curves if $r \geq \frac{\csc\theta}{2}$, see Figure 1.8. Moreover, this cover has a minimum area 0.34501 at $r = \frac{csc\theta}{2}$ where $\theta \approx 1.16556$, which is better than Meir's cover.

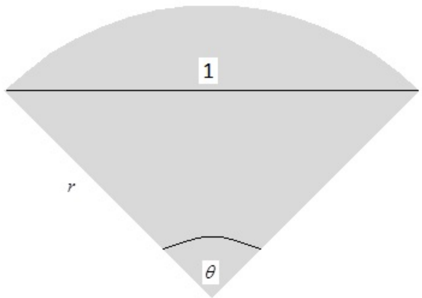

Figure 1.8: Wetzel's sector.

Next, Gerriets [14] constructed a smaller cover which has area 0.32140. This cover consists of a semi-ellipse with semi-minor axis $\frac{1}{4}$ and major axis 1 and an isosceles triangle with base 1 and height $\frac{1}{4}$, as shown in Figure 1.9.

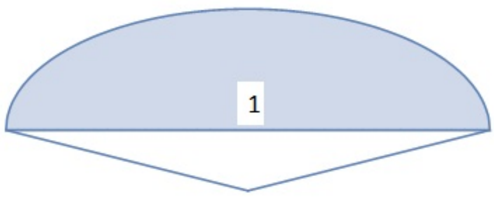

Figure 1.9: Gerriets's cover.

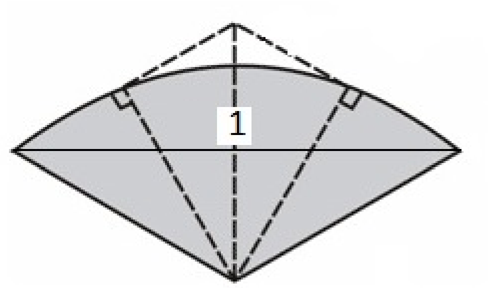

Figure 1.10: Norwood, Poole and Laidacker's cover.

In 1975, Gerriets and Poole [15] established the new cover which was a rhombus with major diagonal 1 and minor diagonal $\frac{1}{\sqrt{3}}$. Its area is 0.2887. Furthermore, this rhombus was truncated to attain a region with the area 0.2861. 17 years later, Norwood, Poole and Laidacker [39] improved this truncated rhombus to be a smaller cover with area 0.27524, see Figure 1.10. In addition, in 2003 Norwood and Poole [38] adapted this cover to be convex with the area of 0.2738086, see Figure 1.7. Then, in 2006 Wang [52] improved Norwood and Poole's cover. Wang's cover has the area 0.2709119, see Figure 1.11. In 2019, Pansaksa and Wichiramala [42] showed that a 30° circular sector of unit radius is a cover for this problem which has an area of 0.2618 confirming Wetzel's conjecture [55] from 1972. This sector is the best current cover for this problem which leads $b = 0.2618$, see Figure 1.12.

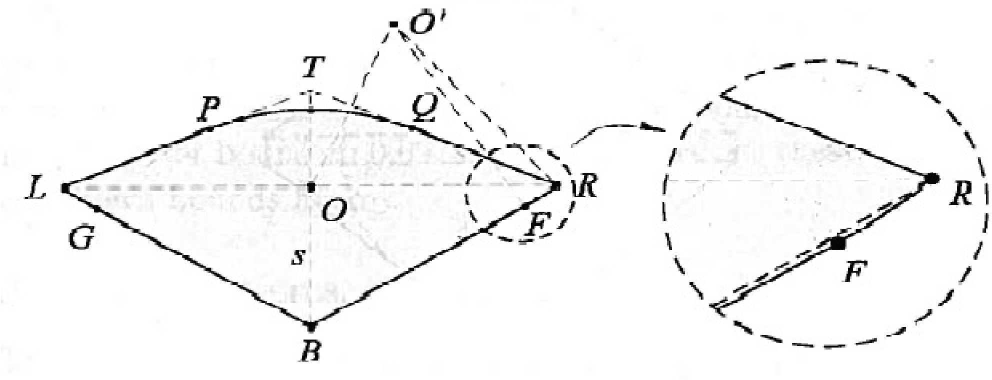

Figure 1.11: Wang's cover

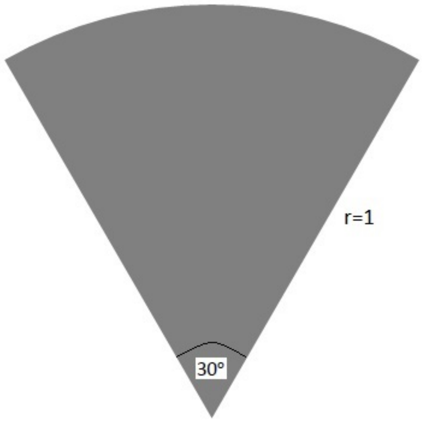

Figure 1.12: a 30° circular sector of unit radius.

On the other hand, the lower bound for this problem is also an interesting problem, which is to find the smallest convex hull area for some arcs. In 1968, Schare and Wetzel [46] determined the first lower bound by considering the convex hull of a unit segment and the unit broadworm in Figure 1.5 with area 0.21946. In 2002, Ferguson had conjectured that the minimum area of convex hull of a unit segment, a "V" shape of two sides of an equilateral triangle of side $\frac{1}{2}$, and a square staple with sides $\frac{1}{3}$ is 0.2388. In 2005, Tanadkithirun [50] disproved the conjecture by finding a small configuration of these arcs with area 0.2275896 see Figure 1.15. In 2009, Khandhawit and Sriswasdi [31] used grid-search algorithm to prove that Tanadkithirun's configuration is the optimal configuration. In 2013, Khandhawit, Sriswasdi and Pagonakis [30] used min-max strategy to show that the smallest convex hull of a staple with leg $\frac{1}{4}$ and base $\frac{1}{2}$, see Figure 1.13, a unit line, a "V" shape of two side of an equilateral triangle of side $\frac{1}{2}$ and unit broadworm is 0.235539. It is a current lower bound which leads $a = 0.235539$. Hence, $0.235539 \leq \alpha_0 \leq 0.2618$.

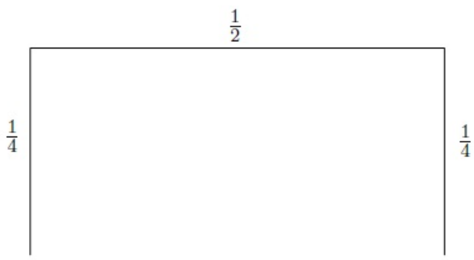

Figure 1.13: a staple with leg $\frac{1}{4}$ and base $\frac{1}{2}$.

We can see that to find the smallest area of convex hull of three arcs is not a simple task. For instance, the conjecture of Ferguson which claims that symmetric arrangement of the arcs gives the minimum area of convex hull is false, see Figure 1.14 and 1.15. It follows that symmetry argument cannot be applied to this problem. Hence, the rearrangement of three objects is very complicated. So, finding the solution to the lower bound problem for three or more arcs is limited. However, we can apply lost in the forest method to find the solution to the upper bound problem. It may be concluded that to find the minimum area of convex hull for three or more arcs is more challenging than the upper bound problem.

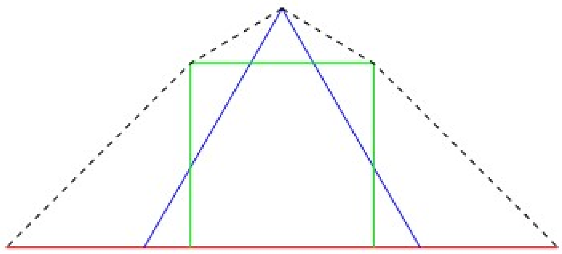

Figure 1.14: Fergoson's conjecture.

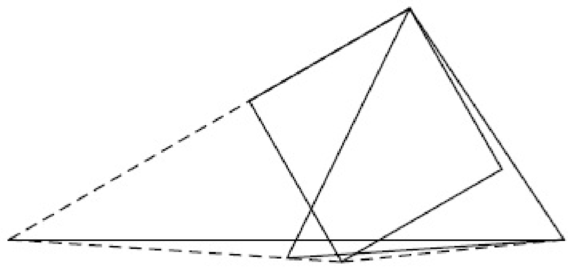

Figure 1.15: The small configuration found by Tanadkithirun.

1.1.5 Convex arcs

Let α_2 be the smallest area of convex cover for convex unit arcs which is a simple arc that lie on the boundary of their convex hull. Because the current lower bound for Moser's worm problem is the smallest convex hull of a staple with leg $\frac{1}{4}$ and base $\frac{1}{2}$, a unit line, a "V" shape of two side of an equilateral triangle of side $\frac{1}{2}$ and unit broadworm [30] which are convex unit arcs, it is a lower bound for convex arcs as well. For upper bound, Besicovitch [4] in 1965 proved that an equilateral triangle of side 1 and area 0.433213 accumulated all convex arcs. In 1970, Wetzel [55] demonstrated that an isosceles right triangle with hypotenuse 1 and area 0.25. Moreover, he showed that all convex arcs were in this triangle cut off at height 0.47140, whose area is 0.24918, as shown in Figure 1.16. Later, Jonson, Poole, and Wetzel [27] trimmed an isosceles right triangle with hypotenuse 1 by 2 symmetric parabolas to construct a cover which has an area of 0.24656, see Figure 1.17.

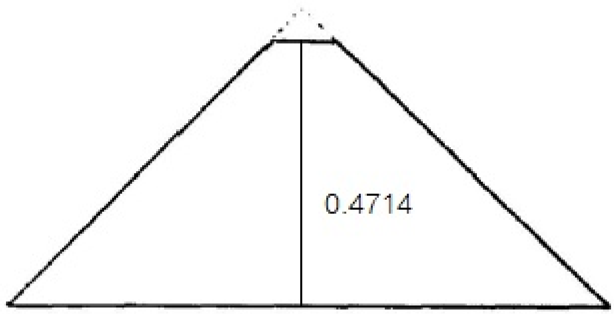

Figure 1.16: The isosceles right triangle clipped at height 0.47140.

In 2005, Wichiramala [57] cropped an isosceles right triangle with hypotenuse 1 at height 0.44, whose area is 0.2464, see Figure 1.18. The present optimal upper bound is 0.24170. It was shown by Wichiramala [56] in 2010. This cover is the quadrilateral $ABCD$ in Figure 1.19 which has $AB = 1$, $\alpha = 31.77°$ and $\beta = 68.9294°$. Therefore, $0.23223 \leq \alpha_2 \leq 0.24170$.

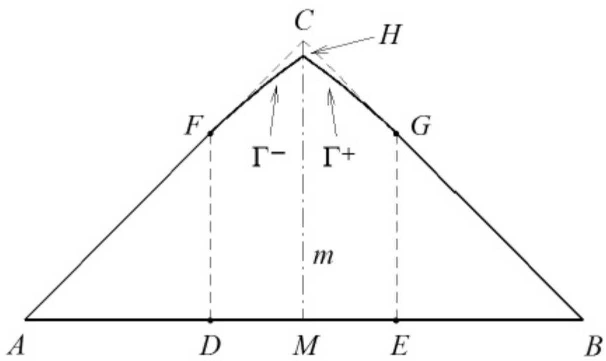

Figure 1.17: The cover for convex unit arcs with area 0.24656.

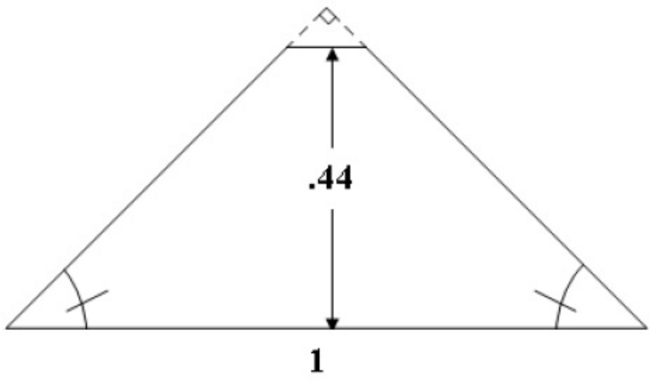

Figure 1.18: The cover for convex unit arcs with area 0.2464.

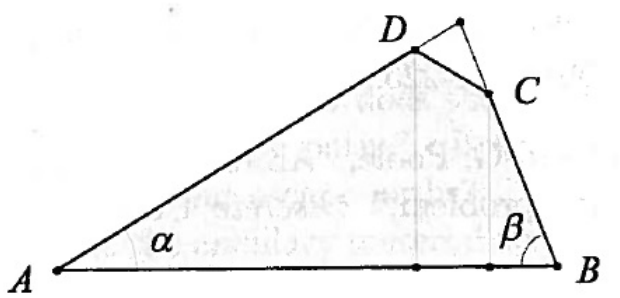

Figure 1.19: The cover for convex unit arcs.

1.1.6 Closed arcs

Let α_3 be the smallest area of convex cover for closed unit arcs, such as its endpoints coincide. In 1957, Eggleston [9] proved that the triangle covers all closed arcs if and only if it covers a circle of radius $\dfrac{1}{\pi}$. It follows that the smallest triangular cover is an equilateral triangle with sides $\dfrac{\sqrt{3}}{\pi}$ which has an area of 0.13162. Later, Jones and Schare [28] showed that the rectangle accommodates all closed arcs if and only if its diagonal is at least $\dfrac{1}{2}$ in 1973. It leads to be the smallest rectangular cover proved by Schare and Wetzel [45], which has an area of 0.122738 in Figure 1.20. In 2006, Furedi and Wetzel [13] decreased the upper bound to 0.117493 by demonstrating the closed arcs cannot be in four corner triangles of the smallest rectangular cover at the same time. Hence, this rectangle was cut at least one corner triangle, which was the isosceles triangle with leg 0.1025 see in Figure

1.21.

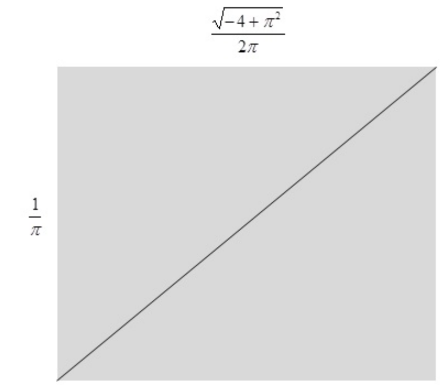

Figure 1.20: The smallest rectangular cover.

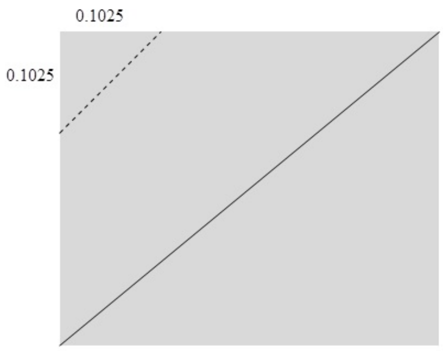

Figure 1.21: The truncated regtangular cover.

They also construct the small covers for closed curves which are the pentagon and curvilinear rectangle in 2011. They showed that the pentagon in Figure 1.22 contained all closed arcs which its area is 0.112242 and reduced this pentagon to curvilinear rectangle in Figure 1.23 which has area 0.11213. In 2018, Wichiramala [56] showed that the opposite corner of this pentagon can be clipped to be irregular hexagon and its area is 0.11023, see Figure 1.24.

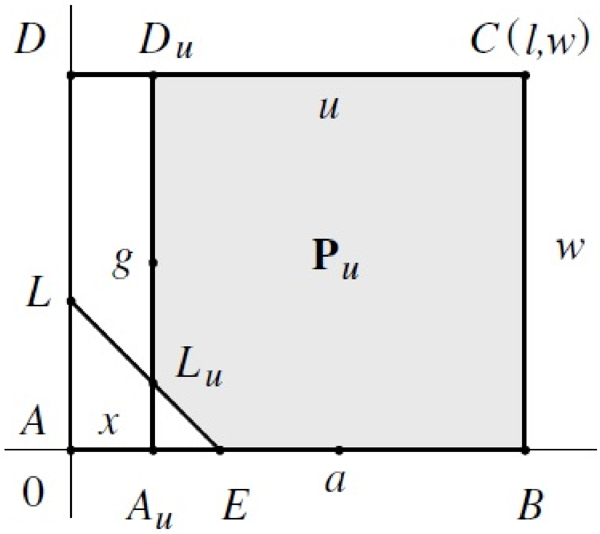

Figure 1.22: The pentagonal cover.

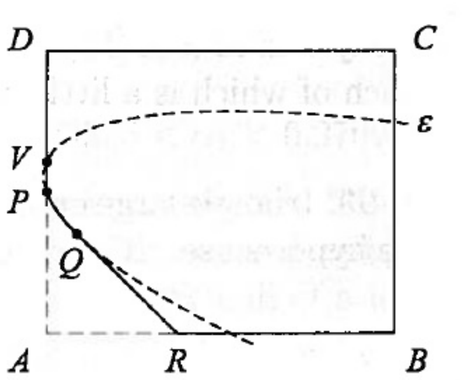

Figure 1.23: The curvilinear rectangle.

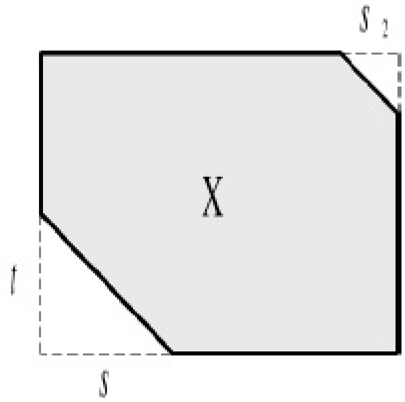

Figure 1.24: The current closed arc cover which $t \approx 0.148$, $s \approx 0.142$ and $s_2 \approx 0.0617$.

For the lower bound, In 1973, Chakerian and Klamkin [6] applied Fary and Redei's theorem [11] which states that the area of convex hull of two centrally symmetric convex objects attains the minimum when the center of two objects coincides to find the first lower bound by using a segment and a circle. Its area is 0.0963275, see Figure 1.25. In 2006, Furedi and Wetzel [13] improved the lower bound to 0.0966675 by using $u \times v$ rectangle instead of the line segment, which attained the minimum value when $v = 0.0130843$ see in Figure 1.26. Five years later, Furedi and Wetzel [13] modified $u \times v$ rectangle to a curvilinear rectangle see in Figure 1.27. By Fary and Redei's theorem, they obtain the minimum area of convex hull of the curvilinear rectangle and a circle of perimeter 1, which is 0.096694. In 2010, My master's thesis [48] raised the lower bound to 0.096905 which is the smallest area of convex hull of a circle of perimeter 1, a $\frac{1}{2}$ line segment, and an equilateral triangle of side $\frac{1}{3}$, see Figure 1.28. Although there is no theorem to find the smallest area of convex hull for three arcs, it can be determined by grid-search algorithm which is similar to Khandhawit and Sriswasdi's work [31]. Thus, the best bounds before this thesis were $0.096905 \leq \alpha_3 \leq 0.11023$. **We have improved the lower bound from 0.096905 to 0.1 in this thesis.**

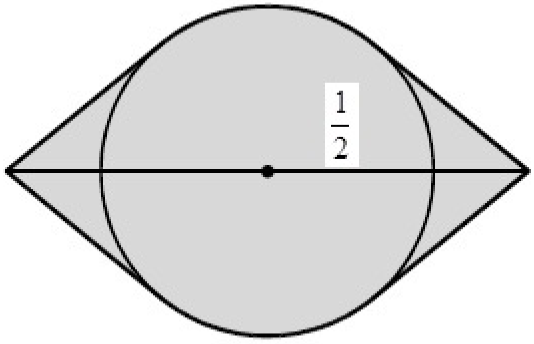

Figure 1.25: The smallest configuration of a circle with perimeter 1 and a half unit line segment.

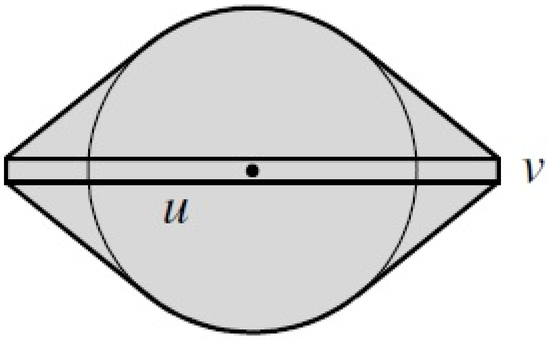

Figure 1.26: The smallest configuration of a circle with perimeter 1 and $u \times v$ rectangle.

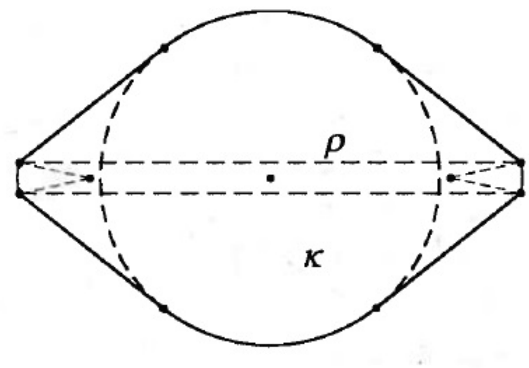

Figure 1.27: The smallest configuration of a circle with perimeter 1 and curvilinear rectangle.

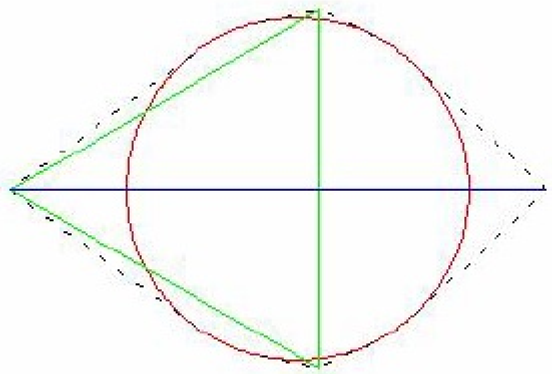

Figure 1.28: The best configuration with area 0.0970439.

We can see that the lower bound problem for three and more objects is very complicated because there is no theorem to guarantee the minimum values such as Ferguson's conjecture. Brass and Sharifi [5] used a computational method to prove a lower bound for Universal cover problem in 2005. This method is called "Brass grid search method". They improved the lower bound for this problem by finding the smallest area of convex hull for three sets. In 2007, Khandhawit and Sriswasdi [31] used this method to improve the lower bound for all unit arcs by considering the smallest convex hull of a unit segment, a "V" shape of two sides of an equilateral triangle of length $\frac{1}{2}$, and a square staple with side $\frac{1}{3}$. It has an area of at least 0.227498. For closed arcs, in 2010, Som-am [48] used the Brass grid search method to improve the former lower bound, 0.0966675, by recognizing convex hull of a line segment, a circle and an equilateral triangle. In 2018, Gibbs [16] used simulated annealing method to find the smallest area of convex hull of a circle and regular Reuleaux polygon with $3, 5, 7$, and 9 side for universal cover's problem. Its area was 0.836991. However, he did not prove the lower bound. It can be shown that we should use a numerical method to solve this problem. In addition, the lower bound for closed arcs has not been improved for a long time hence this problem is very difficult. Therefore, this research focuses on improving the lower bound of convex cover for closed arcs by using three objects. For four and more objects, we do not mention in this work because it is very hard to solve as [16].

1.2 Structure and Results

- In Chapter 2, we discuss the methods for finding numerically a lower bound for a minimum of Lipschitz function f on a compact set. The problem of finding the lower bound of area of convex cover for closed unit arcs is studied starting from Chapter 3.

- In Chapter 3, we rigorously prove the first lower bound which is 0.0975 by considering an area of convex hull of a circle, a rectangle and an equilateral triangle. First, we use a numerical method to find the best rectangle which gives an area of convex hull for these arcs as large as possible with size 0.0375×0.4625. Next, we combine a geometric method and a numerical method from Chapter 2 to prove the bound. Finally, we will show that the convex cover for these arcs has area at most 0.09763 [21].

- In Chapter 4, we use numerical method to find the lower bound for two centrally symmetric objects which has an area of 0.0966693 by applying the result of [11]. It is close to the current bound for 2 objects which is 0.096694. Furthermore, we use a systematic search for 2 and 3 objects and find that possible optimal area is 0.10044 by considering the area of convex hull of a circle, a line and a rectangle of size 0.1727×0.3273. There is no rigorous proof in this chapter.

- In Chapter 5, we rigorously prove that the lower bound for the same problem is improved to 0.1, based on the configuration which is found numerically is Chapter 4. Next, we will show that the convex cover for a circle, a line and a rectangle has area between 0.1 and 0.1005. This shows that $0.1 \leq \alpha_3$, but this set of objects cannot be used to prove the bound $0.1005 \leq \alpha_3$ or better. Thus, $0.1 \leq \alpha_3 \leq 0.11023$ [22].

- In Chapter 6, we give the summary and conjecture for upper bound which has an area of 0.1046.

Chapter 2

Numerical minimization problem of Lipschitz function

The lower bound problem is to find the smallest area of convex hull of some unit closed arcs. It is related to a minimization problem. We consider the problem of minimizing function $f : \mathbb{R}^n \to \mathbb{R}$ on set $A \subset \mathbb{R}^n$. That is,

$$\min_{x \in A} f(x) \tag{2.1}$$

f is called the **objective function** and A is the set of **feasible solutions**. In general, there is no guarantee that f has a minimum value. In Section 2.1 we give definitions and theorems to guarantee that (2.1) has a solution.

2.1 Basic definitions and theorems

Definition 2.1 ([47], p.140)**.** Let $A \subset \mathbb{R}^n$. A function $f : A \to \mathbb{R}$ is said to be continuous at a point $x_0 \in A$ if given any $\epsilon > 0$, there exists a $\delta > 0$ such that $\|f(x) - f(x_0)\| < \epsilon$ whenever $x \in A$ and $\|x - x_0\| < \delta$. Equivalently, f is continuous at x_0 if and only if $\lim_{x \to x_0} f(x) = f(x_0)$.

Definition 2.2 ([47], p.140)**.** Let $A \subset \mathbb{R}^n$. A function $f : A \to \mathbb{R}$ is continuous on A if it is continuous at all $x \in A$.

Definition 2.3 ([25], p.9)**.** Let $A \subset \mathbb{R}^n$. A function $f : A \to \mathbb{R}$ is Lipschitz continuous in A if there exist constant $C > 0$ such that for all $x, y \in A$ we have

$$|f(x) - f(y)| \leq C \|x - y\|_1$$

where $\|x - y\|_1 = \sum_{i=1}^n |x_i - y_i|$.[1]

[1] more standard norm is $\|x - y\|_2 = \sqrt{\sum_{i=1}^n |x_i - y_i|^2}$ but in $\mathbb{R}^n$ all norms are equivalent.

Definition 2.4. Let $A \subset \mathbb{R}^n$ and $C_1, C_2, \dots, C_n$ be some positive constants. A function $f : A \to \mathbb{R}$ is $(C_1, C_2, \dots, C_n)$-Lipschitz continuous on A, if for all $x = (x_1, x_2, \dots, x_n), x' = (x_1, x_2, \dots, x_i + \delta, \dots, x_n) \in A$, for all $\delta > 0$, inequality

$$|f(x) - f(x')| \leq C_i \delta \tag{2.2}$$

holds for all $i \in \{1, 2, \dots, n\}$.

The following theorem proves that it is sufficient to check (2.2) only for small δ.

Theorem 2.5. *Let $A \subset \mathbb{R}^n$. Let $f : A \to \mathbb{R}$ and $C_i > 0$ are some constants such that for all $x = (x_1, x_2, \dots, x_n), x' = (x_1, x_2, \dots, x_i + \delta, \dots, x_n) \in A$ if there exists $\delta_0 > 0$ such that*

$$|f(x) - f(x')| \leq C_i \delta \tag{2.3}$$

for all $i \in \{1, 2, \dots, n\}$ and for all $\delta < \delta_0$. Then (2.3) holds for all $\delta > 0$, and thus f is $(C_1, C_2, \dots, C_n)$-Lipschitz continuous on A.

Proof. Let $f(x_1, x_2, \dots, x_i, \dots, x_n) = g(x_i)$. Let $\delta > 0$. There exists $N = \lfloor \frac{\delta}{\delta_0} \rfloor$ such that $\delta < N\delta_0$. Let $\delta_1 = \frac{\delta}{N} < \delta_0$. We have $|g(x_i + \delta_1) - g(x_i)| \leq C_i \delta_1$ by (2.3). Thus,

$$\begin{aligned} |g(x_i) - g(x_i + N\delta_1)| &\leq \sum_{j=1}^{N} |g(x_i + j\delta_1) - g(x_i + (j-1)\delta_1)| \\ &\leq \sum_{j=1}^{N} C_i \delta_1 = C_i(N\delta_1) \end{aligned}.$$

□

Theorem 2.6. *Every $(C_1, C_2, \dots, C_n)$-Lipschitz continuous is Lipschitz continuous function with $C = \max\{C_1, C_2, \dots, C_n\}$. Conversely, every Lipschitz continuous function is $(C, C, \dots, C)$-Lipschitz continuous.*

Proof. ($\implies$), Let $x = (x_1, x_2, \dots, x_n), y = (x_1 + \delta_1, x_2 + \delta_2, \dots, x_n + \delta_n) \in A$. Thus,

$$\begin{aligned} |f(x) - f(y)| &\leq \sum_{i=1}^{n} |f(x_1 + \delta_1, x_2 + \delta_2, \dots, x_i + \delta_i, x_{i+1}, \dots, x_n) \\ &\quad - f(x_1 + \delta_1, x_2 + \delta_2, \dots, x_{i-1} + \delta_{i-1}, x_i, \dots, x_n)| \\ &\leq \sum_{i=1}^{n} C_i \delta_i \\ &\leq C \sum_{i=1}^{n} \delta_i \\ &= C \|x - y\|_1 . \end{aligned}$$

($\Longleftarrow$), Let $x_1, x_2 \in \mathbb{R}^n$. Set $x_1 - x_2 = \delta e_i$, where e_i is a unit vector in ith component. Hence, $|f(x_2 + \delta e_i) - f(x_2)| \leq C|\delta e_i| = C\delta$ for all $i \in \{1, 2, \ldots, n\}$. $\square$

Definition 2.7 ([51], p.5)**.** Euclidean norm of $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ is $\|x\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$.

Definition 2.8 ([51], p.43)**.** A function $f : A \subset \mathbb{R}^n \to \mathbb{R}$ is differentiable at x_0 if there exits a $1 \times n$ matrix D such that for all $\epsilon > 0$, there is $\delta > 0$ such that $x \in A$ and $\|x_0 - x\| < \delta$ implies

$$\|f(x_0) - f(x) - D(x_0 - x)\| < \epsilon \|x_0 - x\| .$$

Equivalently, f is differentiable at $x_0 \in A$ if $\lim\limits_{x \to x_0} \dfrac{\|f(x) - f(x_0) - D(x - x_0)\|}{\|x - x_0\|} = 0$.

Definition 2.9 ([51], p.44)**.** If f is differentiable (or smooth) at all points in A, then f is said to be differentiable on A.

Definition 2.10 ([51], p.174)**.** A function $f : A \subset \mathbb{R}^n \to \mathbb{R}$ is convex if $\forall x_1, x_2 \in A, \forall \lambda \in [0, 1]$ such that $f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2))$.

In this thesis, we work with $(C_1, C_2, \ldots, C_n)$-Lipschitz continuous function f. So, f is continuous. Next, we will consider set $A \subset \mathbb{R}^n$ and definition 2.10-2.18 are in [51].

Definition 2.11. Let $x \in A \subset \mathbb{R}^n$. The open ball with radius r center at a point $p \in A$ is defined by $B(p, r) = \{x \in A |\, \|x - p\| < r\}$, where $\|.\|$ is Euclidean norm.

Definition 2.12. A set $A \subset \mathbb{R}^n$ is open if for all $x \in A$ there exits an $\epsilon > 0$ such that $B(x, \epsilon) \subset A$.

Definition 2.13. A set $A \subset \mathbb{R}^n$ is closed if its complement A^C is open.

Definition 2.14. A set $A \subset \mathbb{R}^n$ is bounded if there exits a constant $r > 0$ such that $A \subset B(0, r)$.

Theorem 2.15 ([51], p.23)**.** *A set $A \subset \mathbb{R}^n$ is a compact if and only if it is closed and bounded.*

Theorem 2.16 (Weierstrass's Extreme Value Theorem)**.** *Every continuous function on a compact set attains its extreme values on that set.*

Definition 2.17. A vector $x^* \in A \subset \mathbb{R}^n$ is a feasible solution of (2.1) if $x^* \in A$.

Definition 2.18. A vector $x^* \in A \subset \mathbb{R}^n$ is a local optimal solution of (2.1) if there exits a $\delta > 0$ such that $f(x^*) \leq f(x)$ for all $x \in \{x \in A |\, \|x - x^*\| \leq \delta\}$.

Definition 2.19. A vector $x^* \in A \subset \mathbb{R}^n$ is a global optimal solution of (2.1) if $x^* \in A$ and $f(x^*) \leq f(x)$ for all $x \in A$.

We consider (2.1) with $(C_1, C_2, \ldots, C_n)-$ Lipschitz continuous f on a compact $A \subset \mathbb{R}^n$. By Weierstrass's Extreme Value Theorem, (2.1) has an global optimal solution. There are many methods to find the local minimum such as Line search method, Steepest Descent method, Newton method, Quasi-Newton method, Conjugate Gradient method, and etc., but most of them require derivative function. Although, the local search method can generate many local minimum points, it does not guarantee that it has a global optimum. Some global optimization methods require special property such as convexity to find exact global optimum solution. In general, the global optimal solution for (2.1) is impossible to solve exactly, hence we use numerical approximation. In (2.1), if f is locally Lipschitz continuous (but no guarantee of convexity or smoothness), we can find the lower bound of $f(x^*)$ by applying the definition of $(C_1, C_2, \ldots, C_n)-$ Lipschitz continuous function with the following theorem.

Theorem 2.20. *If f is $(C_1, C_2, \ldots, C_n)-$ Lipschitz continuous function on $A \subset \mathbb{R}^n$, then*

$$f(x_1 + \delta_1, x_2 + \delta_2, \ldots, x_n + \delta_n) \geq f(x_1, x_2, \ldots, x_n) - \sum_{i=1}^{n} C_i \delta_i \tag{2.4}$$

for all $\delta_i > 0$ and all $x = (x_1, x_2, \ldots, x_n) \in A$.

Proof. Since f is $(C_1, C_2, \ldots, C_n)-$ Lipschitz continuous function, we have

$$|f(x_1, x_2, \ldots, x_i + \delta_i, \ldots, x_n) - f(x_1, x_2, \ldots, x_n)| \leq C_i \delta_i$$

for all $\delta_i > 0$, $i = 1, 2, \ldots, n$. Thus,

$$\begin{aligned} |f(x_1, x_2, \ldots, x_i + \delta_i, \ldots, x_n) - f(x_1, x_2, \ldots, x_n)| &\leq \sum_{i=1}^{n} |f(x_1 + \delta_1, x_2 + \delta_2, \ldots, x_i + \delta_i, x_{i+1}, \ldots, x_n) \\ &\quad - f(x_1 + \delta_1, x_2 + \delta_2, \ldots, x_{i-1} + \delta_{i-1}, x_i, \ldots, x_n)| \\ &\leq \sum_{i=1}^{n} C_i \delta_i \end{aligned}$$

$$f(x_1 + \delta_1, x_2 + \delta_2, \ldots, x_n + \delta_n) \geq f(x_1, x_2, \ldots, x_n) - \sum_{i=1}^{n} C_i \delta_i.$$

□

Definition 2.21. A box is a set A in form $A = \begin{cases} a_1 \leq x_1 \leq b_1, \\ a_2 \leq x_2 \leq b_2 \\ \vdots \\ a_k \leq x_k \leq b_k \end{cases}$ for some constants $a_i < b_i, i = 1, 2, \ldots, k$. Let $n_1, n_2, \ldots, n_k$ be integers greater than 1. A point $x^* =$

$(x_1^*, x_2^*, \ldots, x_k^*) \in A$ is called grid point if $x_i^* = a_i + \delta_i(j-1), j = 1, 2, \ldots, n_i$, where $\delta_i = \frac{b_i - a_i}{n_i - 1}$ for all $i = 1, 2, \ldots, n$. The set of all grid points is called a grid and, numbers $\delta_1, \delta_2, \ldots, \delta_n$ are called size of the grid.

A box in form $A = \begin{cases} a_1 \le x_1 \le b_1, \\ a_2 \le x_2 \le b_2 \\ \quad\vdots \\ a_k \le x_k \le b_k \end{cases}$ can be written in form $A = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_k, b_k]$.

Definition 2.22. Let a box $A = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_k, b_k]$. The point $x^* = (\frac{b_1+a_1}{2}, \frac{b_2+a_2}{2}, \ldots, \frac{b_k+a_k}{2})$ is called the center of box A.

In our work, we want to prove the lower bound B for minimum such that

$$f(x) \ge B, \forall x \in A. \tag{2.5}$$

Let F be a finite subset of A. We do it by checking inequality

$$f(x) - \sum_{i=1}^{n} C_i \delta_i \ge B, \text{for all } x \in F, \text{ for some } \delta_i > 0. \tag{2.6}$$

This allows to prove (2.5) by checking (2.6) in a finite number of points. If (2.6) holds on a grid, it implies that (2.5) holds for all A by the following theorem.

Theorem 2.23. *Let $A = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_n, b_n]$ and $x = (x_1, x_2, \ldots, x_n)$. Let $f : A \to \mathbb{R}$ be $(C_1, C_2, \ldots, C_n)$-Lipschitz continuous on A. Let G be a grid with size $\delta_1, \delta_2, \ldots, \delta_n$. If (2.6) holds for all points in G then $f(x) \ge B, \forall x \in A$.*

Proof. For every $x = (x_1, x_2, \ldots, x_n)$, there exist point $a = (a_1, a_2, \ldots, a_n)$ in G such that $a_1 \le x_1 \le a_1 + d_1$, $a_2 \le x_2 \le a_2 + d_2, \ldots, a_n \le x_n \le a_n + d_n$. Then, by (2.4),

$$f(x) \ge f(a) - C_1|x_1 - a_1| - C_2|x_2 - a_2| - \ldots - C_n|x_n - a_n| \ge f(a) - C_1 d_1 - C_2 d_2 - \ldots - C_n d_n \ge B$$

because (2.6) holds for all points in the grid. □

This method cannot be used to find the global minimum exactly, but it can be used to prove some lower bound B for it, see Figure 2.1.

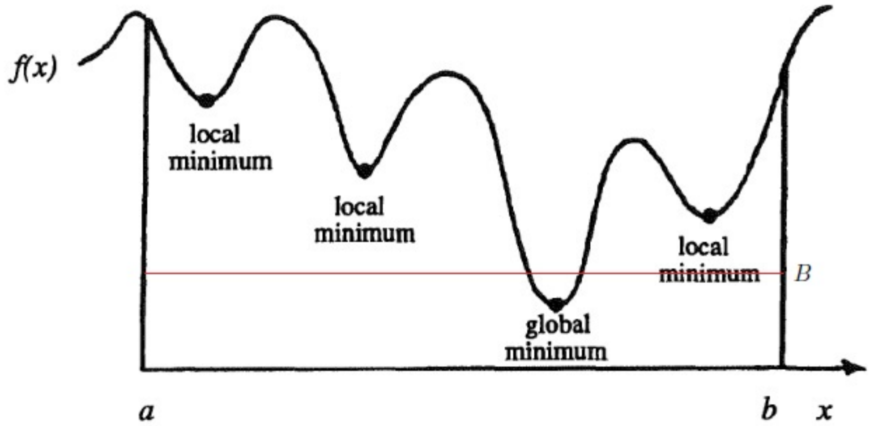

Figure 2.1: The bound B.

Next, we consider the methodology to solve the minimization problem of Lipschitz function numerically.

2.2 Methodology

Presently, we consider five methods for finding a global minimum of a function, which are quick non-rigorous method, naive method with fixed grid, grid search with variable grid, BSA and MHS. BSA and MHS are constructed by us and we will use them to prove the bound in our research.

2.2.1 Quick non-rigorous method

Let $A \subset \mathbb{R}^n$ be a compact set. Let $f : A \to \mathbb{R}$ be a continuous function. If $n = 1$, we use `fminbnd` function in Matlab to find the minimum of single variable function on fixed interval. This function can find the minimum value automatically.

Example 2.24. Let us consider the case $n = 1$. Let $f(x) = x^2$. Find the minimum for f on $[-1, 3]$.

We use `fminbnd` function to find the minimum value by setting `fun=@(x)x.^2` and `[x,fal]=fminbnd(fun,lb,ub)`. Inputs are function (`fun`), lower bound `lb=-1`, and upper-bound `ub=3`. Outputs are a minimum point `x` and a minimum value `fval`. The minimum value by the program is 9.4371×10^{-33} and $x = -9.7145 \times 10^{-17}$. It takes 0.052 seconds to find the solution.

For multivariable function ($n > 1$), we use `fmincon` function to find the minimum value on A.

Example 2.25. Let us consider the case $n = 2$. Let $f(x, y) = x^2 + y^2$. Find the minimum for f on $[-1, 3] \times [-2, 4]$.

We run `fmincon` function by setting `fun=@(x)x(1).^2+x(2).^2` and `[x,fal] =fmincon(fun,x0,A,b,Aeq,beq,lb,ub)` which minimizes `fun` subject to the linear equalities `Aeq*x = beq` and `A*x` $\leq$ `b`. Inputs are function (`fun`), initial points `x0=[1,1]`, lower bound `lb=[-1,-2]`, and upperbound `ub=[3,4]`. We do not have the linear inequalities and equalities. So, setting `A=[],b=[],Aeq=[],beq=[]`. Outputs are a minimum point `x` and a minimum value `fval`. The minimum value by the program is 3.08085×10^{-17} and $x = [-0.0476 \times 10^{-8}, -0.6153 \times 10^{-8}]$. It takes 0.1194 seconds to complete. We see that the numerical minimization function in Matlab can find a numerical solution quickly. However, Matlab does not provide a rigorous proof that the point it returns is indeed a minimum.

2.2.2 Naive method with fixed grid

Grid search method approximates the minimum value of the unknown function which is defined on a compact subset of $\mathbb{R}^n$ [32]. If function f is Lipschitz on compact subset of $\mathbb{R}^n$, the global minimum can be approximated by this method [43]. That is, the grid points are divided by the boundaries and grid sizes in each dimension. This method evaluates every grid points in each dimensions at the same time.

Let $A \subset \mathbb{R}^n$ be a compact set. Let $f : A \to \mathbb{R}$ be $(C_1, C_2, \ldots, C_n)$-Lipschitz continuous function. We consider a function f to prove the bound by checking (2.6) and apply Theorem 2.23. Consider example for $n = 1$. Let $f(x)$ be $(C)-$Lipschitz continuous on $[a, b]$. Let $x_i = a + \frac{i}{k}(b - a), i \in \{0, 1, \ldots, k\}, k \in \mathbb{N}$. Assume that we have checked (2.6) for $x = x_i$, $i = 0, \ldots, k$. By Theorem 2.23, $f(x) > B, \forall x \in [a, b]$, B is a lower bound which we want to prove.

Example 2.26. Let us consider the case $n = 1$. Let $f(x) = x^2$ on $[-1, 3]$ and the bound $B = -12$. Show that f is (6)-Lipschitz continuous function and $f(x) > -12, \forall x \in [-1, 3]$.

Solution. *First, we want to show that f is (6)-Lipschitz continuous function. Let $-1 \leq x \leq 3$ and $-1 \leq x+\delta \leq 3$. We have $|(x+\delta)^2-x^2| = |2x\delta+\delta^2| = \delta|2x+\delta| \leq \delta(|x|+|x+\delta|) \leq (3 + 3)\delta = 6\delta$. Thus, f is (6)-Lipschitz continuous function. Next, we fix grid size $d = 1$. Thus, there are 5 grid points which are $-1, 0, 1, 2, 3$ to check the inequality*

$$f(x_i) - 6d \geq -12 \tag{2.7}$$

It is easy to check that every grid point satisfies (2.7). By Theorem 2.23, $f(x) > -12, \forall x \in [-1, 3]$.

Example 2.27. Let us consider the case $n = 2$. Let $f(x, y) = x^2 + y^2$,$-1 \le x \le 3$ and $-2 \le y \le 4$ and a bound $B = -35$. Show that f is $(6, 8)$-Lipschitz continuous function and $f(x) > -35, \forall x \in [-1, 3] \times [-2, 4]$.

Solution. *First, we want to show that f is $(6, 8)$-Lipschitz continuous function. Let $-1 \le x \le 3$ and $-1 \le x + \delta \le 3$. We have $|(x + \delta)^2 + y^2 - (x^2 + y^2)| = |2x\delta + \delta^2| = \delta|2x + \delta| \le \delta(|x| + |x + \delta|) \le (3 + 3)\delta = 6\delta$. Similarly, $|(x + \delta)^2 + y^2 - (x^2 + y^2)| \le 8\delta$. Thus, f is $(6, 8)$-Lipschitz continuous function on $[-1, 3] \times [-2, 4]$. Next, we fix grid size $d_1 = 2$ and $d_2 = 3$. Thus, there are 9 grid points which are $(-1, -2), (1, -2), (3, -2), (-1, 1), (1, 1), (3, 1), (-1, 4), (1, 4), (3, 4)$ to check the inequality*

$$f(x_1, x_2) - 6d_1 - 8d_2 \ge -35 \tag{2.8}$$

It is easy to check that every grid point satisfies (2.8). By Theorem 2.23, $f(x) > -35, \forall x \in [-1, 3] \times [-2, 4]$.

2.2.3 Grid-search algorithm with variable grid

This method is similar to naive method with fixed grid, but if there is some points which does not satisfy (2.6), we would decrease grid sizes to be a new fixed grid and check (2.6) again and so on. The program will be terminated when all grid points satisfy (2.6). Thus, we have many grid points to check the inequality. For example, if $k = 100$ and f is a five dimensional function, we will have 100^5 points to run. If we want to prove the best bound which is close to minimum value, the program will take time to run it.

In this method, we will reduce the boundaries and grid sizes in order to obtain the minimum value which is close to the exact minimum value. Although we cannot find the exact solution, we can prove the lower bound of Lipschitz function by apply the following theorem:

Definition 2.28. Cell is a set of form $S = \{x = (x_1, x_2, \ldots, x_n) : 0 \le x_i - x_i^* \le \delta_i, i = 1, 2, \ldots, n\}$, where $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$ is a grid point. A cell is a polytope in $\mathbb{R}^n$ with 2^n vertices.

Theorem 2.29. *Let $A = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_n, b_n]$ and $x = (x_1, x_2, \ldots, x_n)$. Let $f : A \to \mathbb{R}$ be $(C_1, C_2, \ldots, C_n)$-Lipschitz continuous on A. Let G be a grid with size $\delta_1, \delta_2, \ldots, \delta_n$. Let S be a cell in G. If (2.6) holds for at least one vertex of S then $f(x) \ge B$ for all $x \in S$.*

Proof. For every $x = (x_1, x_2, \ldots, x_n) \in S$, and every vertex $a = (a_1, a_2, \ldots, a_n)$ of S. we have $|x_1 - a_1| \le \delta_1$, $|x_2 - a_2| \le \delta_2, \ldots, |x_n - a_n| \le \delta_n$. Then, by (2.4),

$$f(x) \ge f(a) - C_1|x_1 - a_1| - C_2|x_2 - a_2| - \ldots - C_n|x_n - a_n| \ge f(a) - C_1\delta_1 - C_2\delta_2 - \ldots - C_n\delta_n \ge B$$

. $\square$

First, we set the big grid size and check (2.6). If there is some point satisfying (2.6), we remove the boundaries that have this point as a vertex by Theorem 2.29. Next, we decrease the grid size (divided by 2) on new boundaries and check (2.6) again and so on. We will do it until all grid points satisfy (2.6), see Figure 2.2. By Theorem 2.23, $f(x) > B$, $\forall x \in A$.

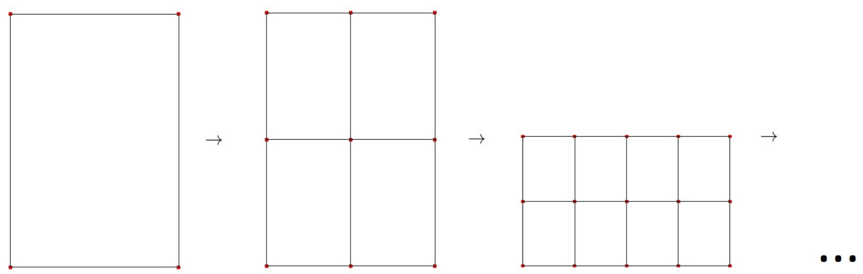

Figure 2.2: Grid points.

Example 2.30. Let us consider the case $n = 1$. Let $f(x) = x^2$ on $[-1, 3]$ and $B = -12$. Show that $f(x) > -12$, $\forall x \in [-1, 3]$.

Solution. *By Example 2.24, f is (6)-Lipschitz continuous function. Let $x_i = -1 + 4i, i \in \{0, 1\}$. We need to check inequality*

$$f(x_i) - 6d \geq -12 \tag{2.9}$$

to hold for both grid points x_0 and x_1. Since $f(x_0) - 6d = f(-1) - 6(4) = -23 < -12$ and $f(x_1) - 6d = f(3) - 6(4) = -15 < -12$, we have to decrease the grid size.

In the second iteration, Let $x_i = -1 + 2i, i \in \{0, 1, 2\}$. We have $f(-1) - 6(2) = -11 > -12$, $f(1) - 6(2) = -11 > -12$ and $f(3) - 6(2) = -3 > -12$. Thus, (2.9) holds for all grid points. By Theorem 2.23, $f(x) > -12$ $\forall x \in [-1, 3]$.

Example 2.31. Let us consider the case when $n = 2$. Let $a_1 \leq x \leq b_1$ and $a_2 \leq y \leq b_2$. We have $x_i = a_1 + id_1$, $y_j = a_2 + jd_2$, $d_1 = \dfrac{b_1 - a_1}{n}$, and $d_2 = \dfrac{b_2 - a_2}{m}$, $i \in \{0, 1, 2, 3, \ldots, n\}$ and $j \in \{0, 1, 2, 3, \ldots, m\}$. First, we determine $f(x_i, y_j)$ and check the condition $f(x_i, y_j) - C_1 d_1 - C_2 d_2 \geq B$. If this inequality holds, B is the lower bound. When it does not hold, we subdivide the new domain which does not satisfy the inequality and reduce d_1 and d_2. In each new domain, we find x_i, y_j, d_1, d_2 and check the condition recursively. We use a concrete function $f(x, y) = x^2 + y^2$,$-1 \leq x \leq 3$, $-2 \leq y \leq 4$ and $B = -23$. Show that $f(x) > -23$, $\forall x \in [-1, 3] \times [-2, 4]$.

Solution. *From Example 2.25, f is $(6,8)$-Lipschitz continuous function. We need to check inequality*

$$f(x_i, y_j) - 6d_1 - 8d_2 \geq -23, \textit{where } d_1, d_2 \textit{ are grid size.} \tag{2.10}$$

to hold for all grid points $(x_i, y_j) = (-1 + id_1, -2 + jd_2), i = 0, 1, j = 0, 1$

We will use grid-search method to prove the bound in (2.10). We have 4 points which are $(-1,-2), (-1,4), (3,4)$ and $(3,-2)$ with $d_1 = 4, d_2 = 6$

First iteration, we check the four points above: $f(-1,-2) - C_1(d_1) - C_2(d_2) = -67 < B, f(-1,4) - C_1(d_1) - C_2(d_2) = -55 < B, f(3,4) - C_1(d_1) - C_2(d_2) = -47 < B$ and $f(3,-2) - C_1(d_1) - C_2(d_2) = -59 < B$. All points do not satisfy the inequality (2.10). So, in the second iteration, we have to decrease the grid sizes on $-1 \leq x \leq 3$ and $-2 \leq y \leq 4$ see Figure 2.3.

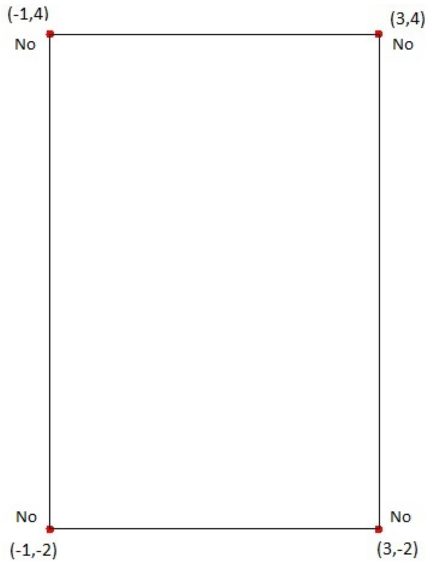

Figure 2.3: Grid points in 1st iteration for Example 2.31.

In the second iteration, Let $x_1 = -1 + id_1,\ y_1 = -2 + jd_2, d_1 = 2, d_2 = 3\ , i, j \in \{0, 1, 2\}$. We have $f(-1,-2) - C_1d_1 - C_2d_2 = -31 < B, f(-1,1) - C_1d_1 - C_2d_2 = -34 < B, f(-1,4) - C_1d_1 - C_2d_2 = -19 > B, f(1,-2) - C_1d_1 - C_2d_2 = -31 < B, f(1,1) - C_1d_1 - C_2d_2 = -34 < B, f(1,4) - C_1d_1 - C_2d_2 = -19 > B, f(3,-2) - C_1d_1 - C_2d_2 = -23 \geq B, f(3,1) - C_1d_1 - C_2d_2 = -26 < B$, and $f(3,4) - C_1d_1 - C_2d_2 = -11 > B$. There are only $(-1,4), (1,4), (3,-2)$ and $(3,4)$ which satisfy the inequality (2.10). By Theorem 2.29, we must decrease d_1 and d_2 further in $-1 \leq x \leq 1$ and $-2 \leq y \leq 1$, see Figure 2.4.

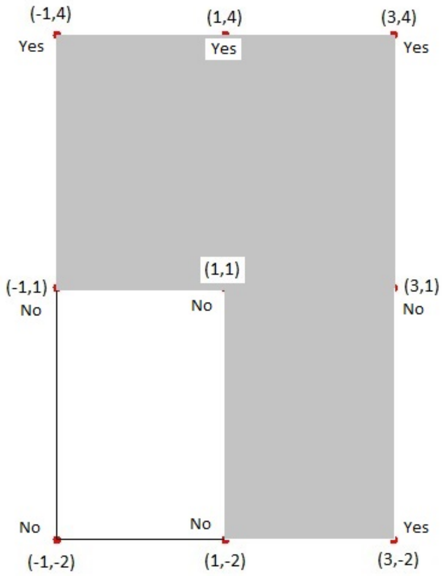

Figure 2.4: Grid points in 2nd iteration for Example 2.31.

In the third iteration, Let $x_1 = -1 + id_1$, $y_1 = -2 + jd_2, d_1 = 1, d_2 = 1.5$ $, i \in \{0, 1, 2\}$ *and* $j \in \{0, 1, 2\}$. *We have* $f(-1, -2) - C_1d_1 - C_2d_2 = -13 > B, f(-1, 0.5) - C_1d_1 - C_2d_2 = -16.75 > B, f(-1, 1) - C_1d_1 - C_2d_2 = -16 > B, f(0, -2) - C_1d_1 - C_2d_2 = -14 > B, f(0, -0.5) - C_1d_1 - C_2d_2 = -17.75 > B, f(0, 1) - C_1d_1 - C_2d_2 = -17 > B, f(1, -2) - C_1d_1 - C_2d_2 = -13 > B, f(1, -0.5) - C_1d_1 - C_2d_2 = -16.75 > B$ *and* $f(1, 1) - C_1d_1 - C_2d_2 = -16 > B$. *The inequality (2.10) holds for all points with* $minf(x_1, y_1) = f(0, -0.25) = 0.25$ *see Figure 2.5.*

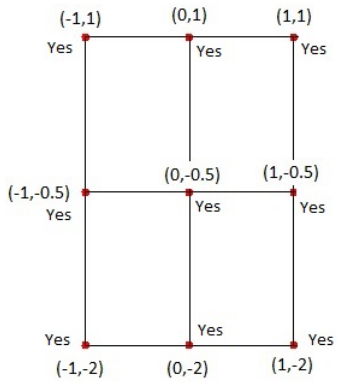

Figure 2.5: Grid points in iteration 3 for Example 2.31.

By Theorem 2.23, $f(x) > -23$ $,\forall x \in [-1, 3] \times [-2, 4]$.

Example 2.32. $f(x, y) = x^2 + y^2$ $,-1 \le x \le 3,\ -2 \le y \le 4$ $B = -0.01$
In every iteration, we apply Theorem 2.29 to cut the domain. At iteration 13, the program terminates, giving a minimum value of 0.0076. Refer to Table 2.1

Iteration	$min(f(x, y))$	$max\ x$	$min\ x$	$max\ y$	$min\ y$
1	5	-1	3	-2	4
2	5	-1	3	-2	4
3	5	-1	3	-2	4
4	3.6	-1	2.5	-2	2.5
5	2.1285	-1	1.375	-1.4375	1.375
6	1.0977	-1	1	-0.9688	1
7	0.5461	-0.7188	0.7188	-0.7344	0.7188
8	0.2689	-0.5156	0.5156	-0.5	0.5078
9	0.1301	-0.3594	0.3594	-0.3594	0.3535
10	0.06	-0.2422	0.2422	-0.2422	0.2441
11	0.0251	-0.1582	0.1582	-0.1582	0.1563
12	0.0076	-0.0869	0.0869	-0.0869	0.0869
13	0.0076	0	0	0	0

Table 2.1: Grid search results in Example 2.32.

By Theorem 2.23, $f(x) \ge -0.01$ for all $x, y \in [-1, 3] \times [-2, 4]$.

Example 2.33. Let us consider the case when $n = 3$. Let $f(x, y, z) = x^2 + y^2 + z^2$ $,-1 \le x \le 3,\ -2 \le y \le 4,\ -3 \le z \le 3$ and $B = -0.1$. Show that f is $(6, 8, 6)$-Lipschitz continuous function and $f(x) \ge -0.1$ for all $x, y, z \in [-1, 3] \times [-2, 4] \times [-3, 3]$.

Solution. *First, we want to show that* f *is* $(6, 8, 6)$*-Lipschitz continuous function. Let* $-1 \le x \le 3, -1 \le x + \delta_1 \le 3$. *We have* $|(x+\delta_1)^2 + y^2 + z^2 - (x^2 + y^2 + z^2)| = |2x\delta_1 + \delta_1^2| = \delta_1|2x + \delta_1| \le \delta_1(|x| + |x + \delta_1|) \le (3 + 3)\delta_1 = 6\delta_1$. *Similarly,* $|x^2 + (y + \delta_2)^2 + z^2 - (x^2 +$

$y^2+z^2)| \leq 8\delta_2$ *and* $|x^2+y^2+(z+\delta_3)^2-(x^2+y^2+z^2)| \leq 6\delta_3$ *Thus,* f *is* $(6,8,6)$*-Lipschitz continuous function. We need to check inequality*

$$f(x_i, y_j, z_k) - 6d_1 - 8d_2 - 6d_3 \geq -0.1$$

, where d_1, d_2, d_3 *are grid sizes. to hold for all grid points* $(x_i, y_j, z_k) = (-1+id_1, -2+jd_2, -3+kd_3), i = 0,1, j = 0,1, k = 0,1$ *and* $d_1 = 4, d_2 = 6, d_3 = 6$*. We apply Theorem 2.29 to cut the domain. At iteration* 12*, the program terminates, giving a minimum value of* 0.0053*. Refer to Table 2.2. See code in Supplementary Material. By Theorem 2.23,* $f(x) \geq -0.1$ *for all* $x, y, z \in [-1,3] \times [-2,4] \times [-3,3]$.

Iteration	$min(f(x,y,z))$	$max\ x$	$min\ x$	$max\ y$	$min\ y$	$max\ z$	$min\ z$
1	5	-1	3	-2	4	-3	3
2	5	-1	3	-2	4	-3	3
3	5	-1	3	-2	4	-3	3
4	5	-1	3	-2	3.25	-2	3
5	5	-1	2.5	-2	2.5	-2	2.5
6	3.0664	-1	1.75	-1.625	1.75	-1.625	1.75
7	1.4727	-1	1.25	-1.25	1.1875	-1.25	1.1875
8	0.7415	-0.8438	0.8438	-0.8281	0.8594	-0.8281	0.8594
9	0.3194	-0.5625	0.5625	-0.5469	0.5547	-0.5469	0.5547
10	0.1107	-0.3281	0.3281	-0.3242	0.3320	-0.3242	0.3320
11	0.0053	-0.0703	0.0703	-0.0723	0.0684	-0.0723	0.0684
12	0.0053	0	0	0	0	0	0

Table 2.2: Grid search results in Example 2.33.

In $\mathbb{R}^n$, let $a_k \leq x_k \leq b_k$, $k \in \{1,2,3,\ldots,n\}$. We have $z_k = a_k + id_k$, $d_k = \dfrac{b_k - a_k}{n_k}$, $i \in \{0,1,2,3,\ldots,n_k\}$. Next, we examine the inequality proposed in (2.6); $f(z_1, z_2, \ldots, z_n) - \sum_{i=1}^{n} C_i d_i \geq B$. If this inequality holds, the proof will be done by Theorem 2.23. Else, we apply Theorem 2.29 to subdivide the new domain which does not satisfy the inequality and reduce d_i and check the inequality again. The algorithm stops when the inequality holds.

Although grid search is very simple to implement, the number of function evaluations grows exponentially with the number of dimensions. We notice that it becomes very inefficient when the dimension is increased. Next, we will construct a new method which reduces the number of function evaluations.

2.2.4 Box-search algorithm

Box-search algorithm (BSA) is a method to find the minimum value of a Lipschitz function which is defined on a compact subset of $\mathbb{R}^n$. We pick the point located at a center of box referred to as the box point. Now, we check if the inequality (2.6) holds, the box is excluded and we are done by Theorem 2.34 as follow

Theorem 2.34. *Let $A = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_n, b_n]$, Let box $S = [c_1, d_1] \times [c_2, d_2] \times \ldots \times [c_n, d_n]$ be a subbox of A of size $(\delta_1, \delta_2, \ldots, \delta_n)$, where $\delta_i = \frac{d_i - c_i}{2}$ is size of S for all $i = 1, 2, \ldots, n$ and $x^* = (\frac{c_1+d_1}{2}, \frac{c_2+d_2}{2}, \ldots, \frac{c_n+d_n}{2})$ is the center of S. Let $f : A \to \mathbb{R}$ be a $(C_1, C_2, \ldots, C_n)$-Lipschitz continuous function. If (2.6) holds for x^* in S then $f(x) \geq B, \forall x \in S$.*

Proof. For every $x = (x_1, x_2, \ldots, x_n) \in S$. Then, by (2.4),

$$\begin{aligned} f(x) &\geq f(x^*) - C_1|x_1 - (\frac{c_1 + d_1}{2})| - C_2|x_2 - (\frac{c_2 + d_2}{2})| - \ldots - C_n|x_n - (\frac{c_n + d_n}{2})| \\ &\geq f(x^*) - C_1\delta_1 - C_2\delta_2 - \ldots - C_n\delta_n \geq B \end{aligned} \tag{2.11}$$

because (2.6) holds for x^* in S. □

If the condition in Theorem 2.34 holds, then B is a lower bound for a minimum, and center x^* of S is an approximation of a minimum point. If not, we subdivide the largest length into two resulting boxes and check the inequality (2.6) again, for each of two boxes. We do the process recursively until all boxes are excluded, see Figure 2.6. We then compare the points in the boxes and pick the minimum value to be an approximation minimum point.

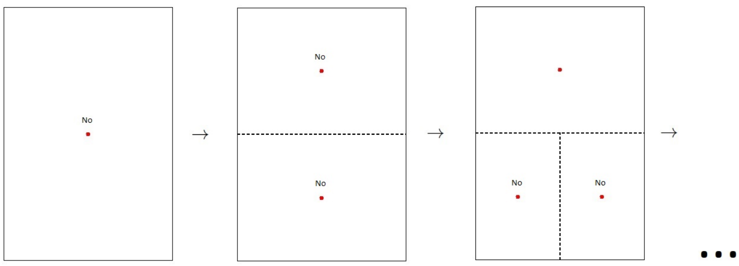

Figure 2.6: The center point of box.

Example 2.35. Recall Example 2.31, with $f(x, y) = x^2 + y^2$. Let $a_1 \leq x_1 \leq b_1$ and $a_2 \leq x_2 \leq b_2$. We have $a = \dfrac{a_1 + b_1}{2}$, $b = \dfrac{a_2 + b_2}{2}$, $d_1 = \dfrac{b_1 - a_1}{2}$, and $d_2 = \dfrac{b_2 - a_2}{2}$. First, we determine $f(x_1, y_1)$ and check the condition $f(x_1, y_1) - C_1d_1 - C_2d_2 \geq B$. If it holds, B is lower bound of f by Theorem 2.34, if not, we choose the maximum length of box length and subdivide it into two boxes recursively as before.

Let $z = (x, y)$ be the midpoint of box $\mathcal{B} = [-1, 3] \times [-2, 4]$. We have $z = (1, 1)$ and $d_1 = 4, d_2 = 6$. Thus, the first iteration gives $f(z) - C_1d_1 - C_2d_2 = -70 < B$, see Figure

2.7.

In the second iteration, we select the largest length $d_2 = 6$ and split it into 2 boxes

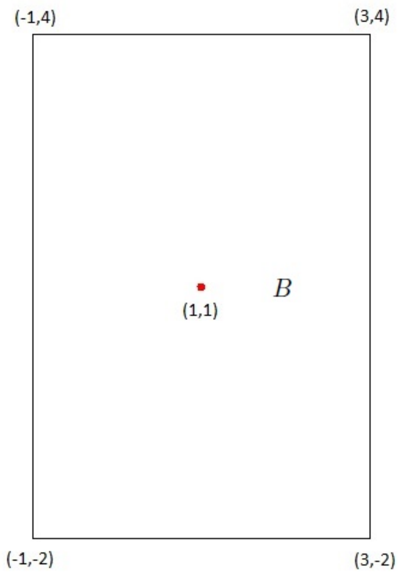

Figure 2.7: 1st iteration of BSA in Example 2.35.

$\mathcal{B}_1 = [-1, 3] \times [-2, 1]$ and $S_2 = [-1, 3] \times [1, 4]$.
$\mathcal{B}_1 : z = (1, -0.5)$ and $d_1 = 4, d_2 = 3$. We have $f(z) - C_1d_1 - C_2d_2 = -46.75 < B$.
$\mathcal{B}_2 : z = (1, 2.5)$ and $d_1 = 4, d_2 = 3$. We have $f(z) - C_1d_1 - C_2d_2 = -40.75 < B$, see Figure 2.8.

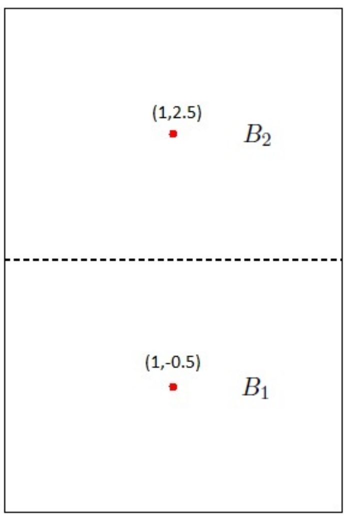

Figure 2.8: 2nd iteration of BSA in Example 2.35.

In the third iteration, we choose the largest length $d_1 = 4$ and split $\mathcal{B}_1$ into 2 boxes $\mathcal{B}_3 = [-1, 1] \times [-2, 1]$ and $\mathcal{B}_4 = [1, 3] \times [-2, 1]$.
$\mathcal{B}_3 : z = (0, -0.5)$ and $d_1 = 2, d_2 = 3$. We have $f(z) - C_1 d_1 - C_2 d_2 = -35.75 < B$.
$\mathcal{B}_4 : z = (2, -0.5)$ and $d_1 = 2, d_2 = 3$. We have $f(z) - C_1 d_1 - C_2 d_2 = -31.75 < B$, see Figure 2.9.

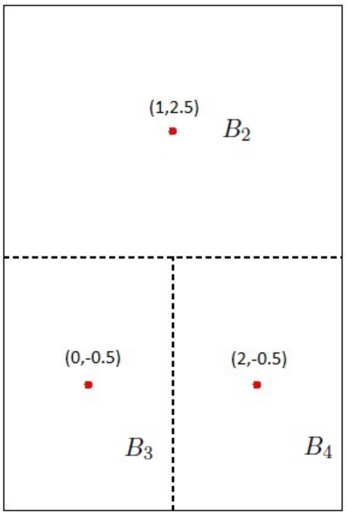

Figure 2.9: 3rd iteration of BSA in Example 2.35.

In the fourth iteration, we choose the largest length $d_2 = 3$ and split $\mathcal{B}_3$ into 2 boxes $\mathcal{B}_5 = [-1, 1] \times [-2, -0.5]$ and $\mathcal{B}_6 = [-1, 1] \times [-0.5, 1]$.
$\mathcal{B}_5 : z = (0, -1.25)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -22.43 > B$. Thus, $\mathcal{B}_5$ is excluded.
$\mathcal{B}_6 : z = (0, 0.25)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -23.94 < B$, see Figure 2.10.

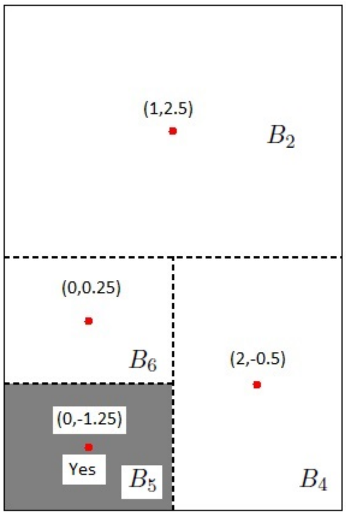

Figure 2.10: 4th iteration of BSA in Example 2.35.

In the fifth iteration, we choose the largest length $d_2 = 2$ and split $\mathcal{B}_6$ into 2 boxes $\mathcal{B}_7 = [-1, 0] \times [-0.5, 1]$ and $\mathcal{B}_8 = [0, 1] \times [-0.5, 1]$.
$\mathcal{B}_7 : z = (-0.5, 0.25)$ and $d_1 = 1, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -17.69 > B$. Thus, $\mathcal{B}_7$ is excluded.
$\mathcal{B}_8 : z = (0.5, 0.25)$ and $d_1 = 1, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -17.69 > B$. Thus, $\mathcal{B}_8$ is excluded, see Figure 2.11.

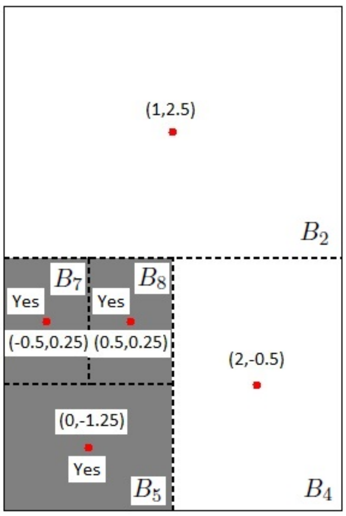

Figure 2.11: 5th iteration of BSA in Example 2.35.

In the sixth iteration, We consider $\mathcal{B}_4$ by choosing the largest length $d_2 = 3$ and split it into 2 boxes $\mathcal{B}_9 = [1, 3] \times [-2, -0.5]$ and $\mathcal{B}_{10} = [1, 3] \times [-0.5, 1]$.

$\mathcal{B}_9 : z = (2, -1.25)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -18.44 > B$. Thus, $\mathcal{B}_9$ is excluded.
$\mathcal{B}_{10} : z = (2, 0.25)$ and $d_1 = 1, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -19.94 > B$. Thus, $\mathcal{B}_{10}$ is excluded, see Figure 2.12.

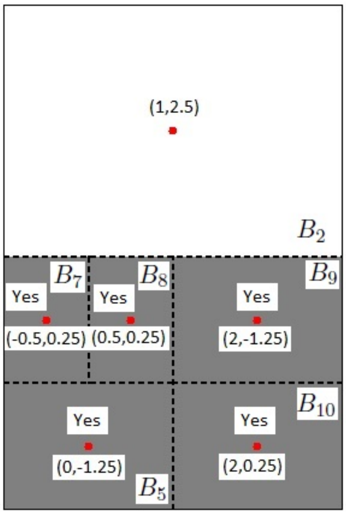

Figure 2.12: 6th iteration of BSA in Example 2.35.

In the seventh iteration, We consider $\mathcal{B}_2$ by choosing the largest length $d_1 = 4$ and split it into 2 boxes $\mathcal{B}_{11} = [-1, 1] \times [1, 4]$ and $\mathcal{B}_{12} = [1, 3] \times [1, 4]$.
$\mathcal{B}_{11} : z = (0, 2.5)$ and $d_1 = 2, d_2 = 3$. We have $f(z) - C_1 d_1 - C_2 d_2 = -29.75 < B$.
$\mathcal{B}_{12} : z = (2, 2.5)$ and $d_1 = 2, d_2 = 3$. We have $f(z) - C_1 d_1 - C_2 d_2 = -25.75 < B$., see Figure 2.13.

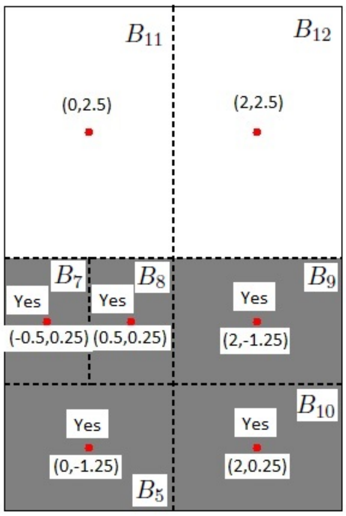

Figure 2.13: 7th iteration of BSA in Example 2.35.

In the eighth iteration, We consider $\mathcal{B}_{11}$ by choosing the largest length $d_2 = 3$ and split it into 2 boxes $\mathcal{B}_{13} = [-1, 1] \times [1, 2.5$ and $\mathcal{B}_{14} = [-1, 1] \times [2.5, 4]$.
$\mathcal{B}_{13} : z = (0, 1.75)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -20.94 > B$. Thus, $\mathcal{B}_{13}$ is excluded.
$\mathcal{B}_{14} : z = (0, 3.125)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -14.23 > B$. Thus, $\mathcal{B}_{14}$ is excluded, see Figure 2.14.

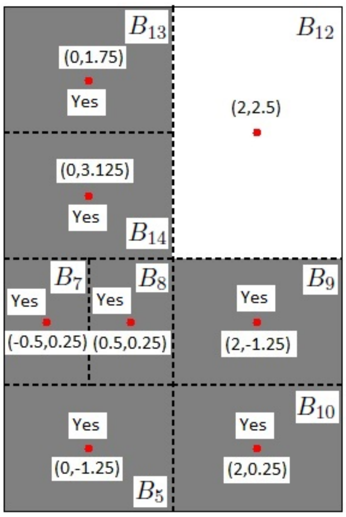

Figure 2.14: 8th iteration of BSA in Example 2.35.

In the ninth iteration, We consider $\mathcal{B}_{12}$ by choosing the largest length $d_2 = 3$ and split it into 2 boxes $\mathcal{B}_{15} = [1, 3] \times [1, 2.5]$ and $\mathcal{B}_{16} = [1, 3] \times [2.5, 4]$.

$\mathcal{B}_{15} : z = (2, 1.75)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -16.94 > B$. Thus, $\mathcal{B}_{15}$ is excluded.
$\mathcal{B}_{16} : z = (2, 3.125)$ and $d_1 = 2, d_2 = 1.5$. We have $f(z) - C_1 d_1 - C_2 d_2 = -10.23 > B$. Thus, $\mathcal{B}_{16}$ is excluded, see Figure 2.15.

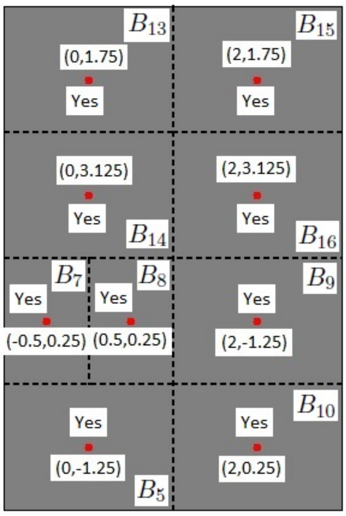

Figure 2.15: 9th iteration of BSA in Example 2.35.

Hence, $f(x_1, y_1) \geq B$ for all $x_1, y_1 \in [-1, 3] \times [-2, 4]$ and the $min f(x_1, y_1) = f(0.5, 0.25) = f(-0.5, 0.25) = 0.0625$. By Theorem 2.34, $f(x) \geq -23$ for all $x, y \in [-1, 3] \times [-2, 4]$.

Example 2.36. Recall Example 2.33. Let us consider the case when $n = 3$ for $f(x, y, z) = x^2 + y^2 + z^2$. Let z be the midpoint of box $\mathcal{B} = [-1, 3] \times [-2, 4] \times [-3, 3]$. Thus, $z = (1, 1, 0)$ and $d_1 = 4, d_2, d_3 = 6$. We have $f(z) - C_1 d_1 - C_2 d_2 - C_3 d_3 = -106 < B$. Next, we select the largest box length $d_2 = 6$ and split it into 2 boxes $\mathcal{B}_1 = [-1, 3] \times [-2, 1] \times [-3, 3]$,$\mathcal{B}_2 = [-1, 3] \times [1, 4] \times [-3, 3]$, Case S_1: $z = (1, -0.5, 0)$ and $d_1 = 4, d_2 = 3, d_3 = 6$. We have $f(z) - C_1 d_1 - C_2 d_2 - C_3 d_3 = -82.75 < B$. Therefore, we subdivide it into 2 boxes. Termination occurred at iteration $i = 115990$ with minimal value being 0.0000534 when $x_1 = -0.0039, y_1 = -0.002, z_1 = -0.0059$. See code in Supplementary Material. By Theorem 2.34, $f(x) \geq -0.1$ for all $x, y, z \in [-1, 3] \times [-2, 4] \times [-3, 3]$.

Next, we will increase the bound from -0.01 to -0.001 and use tic-toc in Matlab to compute the time, see the time taken in the following table.

bound	number of iterations	minimum value	time (sec)
-0.01	115990	5.34×10^{-5}	0.8
-0.009	4358436	7.89×10^{-7}	38.2
-0.008	5149048	3.87×10^{-7}	40.9
-0.007	6305794	3.87×10^{-7}	46.7
-0.006	7978762	2.09×10^{-7}	59.5
-0.005	10463222	1.97×10^{-7}	76.2
-0.004	14565562	9.69×10^{-8}	110.3
-0.003	22569422	5.22×10^{-8}	163.9
-0.002	41198330	2.42×10^{-8}	304.1
-0.001	116527354	6.05×10^{-9}	825.9

Table 2.3: The computation time of Box-search method when the bound changes.

In $\mathbb{R}^n$, let $a_k \leq x_k \leq b_k$, $z_k = \dfrac{a_k + b_k}{2}$ and $d_k = \dfrac{b_k - a_k}{2}$, $k \in \{1, 2, 3, \ldots, n\}$. The inequality $f(z_1, z_2, \ldots, z_n) - \sum_{i=1}^{n} C_i d_i \geq B$ must be checked. If it is true, B will be the lower bound by Theorem 2.23. If not, then the largest box is split into 2 boxes. The inequality will be calculated using the midpoint in a similar manner. If it holds, this box is excluded. Otherwise, we select the largest box and subdivide it into 2 boxes again, and so on, see Algorithm 1.

In Table 2.3, we can see that when the bound increase to -0.001 , the time also increase. In Example 2.33, it is easy to see that the minimum value is 0. If we set the expected bound which is close to 0, the program will run for long time, see Figure 2.16. Hence, it is very difficult to find the best expect bound. To solve this problem, we will construct MHS which does not depend on the expected bound.

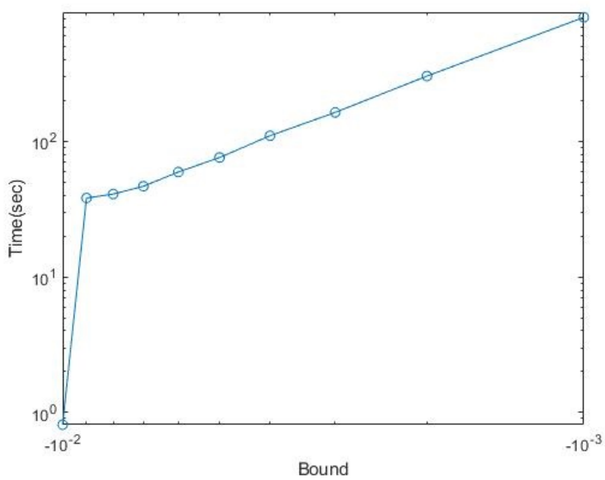

Figure 2.16: Graph of relationship between time and bound (log-log scale) in Table 2.3.

Algorithm 1 Box search algorithm (BSA)

Input: A function f, the boundaries of each parameters $a_1, b_1, a_2, b_2, a_3, b_3, \ldots, a_n, b_n$, area of rectangle $r = 0$, the number of iteration which satisfy the inequality n=0, initial minimum area a, and the bound B.

Output: Minimum area with parameters $a, x_1, \ldots, x_n$, area of rectangle r , and the number of iteration which satisfy the inequality n .

Procedure:

Function `checkmin`$(f, a_1, b_1, a_2, b_2, a_3, b_3, \ldots, a_n, b_n, a, r, n, B)$

set $x_1 := \dfrac{a_1 + b_1}{2}$, $x_2 := \dfrac{a_2 + b_2}{2}$, …,$x_n := \dfrac{a_n + b_n}{2}$

set $d_1 := \dfrac{b_1 - a_1}{2}$, $d_2 := \dfrac{b_2 - a_2}{2}$, …, $d_n := \dfrac{b_n - a_n}{2}$

set $d := max\{d_1, d_2, \ldots, d_n\}$

if $f(x_1, x_2, \ldots, x_n) - C_1 \cdot d_1 - C_2 \cdot d_2 - \ldots - C_n \cdot d_n > B$

  set $r := r + 2^n \cdot d_1 \cdot d_2 \cdot \ldots \cdot d_n$;

  $n = n + 1$;

  return;

else

  if $f(x_1, x_2, \ldots, x_n) < a$ (To find minimum)

    $f(x_1, x_2, \ldots, x_n) = a$

  end if

  switch d_k **do**

  case 1

  for i=0 **to** 1 **do**

  `checkmin`$(f, a_1 + d_1 \cdot i_1, a_1 + d_1 \cdot (i_1 + 1), a_2, b_2, \ldots, a_n, b_n, a, r, n, B)$

  end do

  case 2

  for i=0 **to** 1 **do**

  `checkmin`$f, (a_1, b_1, a_2 + d_2 \cdot i_2, a_2 + d_2 \cdot (i_2 + 1), \ldots, a_n, b_n, a, r, n, B)$

  end do

  ⋮

  case n

  for i=0 **to** 1 **do**

  `checkmin`$(f, a_1, b_1, a_2, b_2, \ldots, a_n + d_n \cdot i_n, a_n + d_n \cdot (i_n + 1), a, r, n, B)$

  end do

  end switch

end if;

end Procedure

2.2.5 Heap sort algorithm

In BSA, it is quite difficult to find the best expected bound. For example, the minimum value is 0 in Example 2.33. If we set the bound which is close to 0, the program will be very slow, see Figure 2.16. The aim of heap sort method is not to prove any given bound, but instead gradually improve the bound on successive iterations. We see that we do not need to give the expected bound in this method.

Heap sort algorithm [3] is a method to sort an array of size n by using a binary tree. Each parent node i has at most 2 children which are $2i$ and $2i+1$. In min heap, each node is numerically smaller than its children. We will apply this method in data structure to sort the bound.

Heap sort in data structure

We begin with the binary tree which is stored in an array. If all levels of the tree, except the last, are completely filled and all nodes are as far left as possible, the binary tree is called 'complete'. Using heap method, the tree must be complete. Let $\texttt{A}$ be the array of length l. Let $\texttt{A}[1]$ be the first node with $\texttt{A}[2]$ as the left child and $\texttt{A}[3]$ as the right child. Continuing this process we have $\texttt{A}[4]$ as the left child of $\texttt{A}[2]$ while $\texttt{A}[5]$ being the right child and so on. In other words, the process starts from top to bottom and left to right as shown below in Figure 2.17.

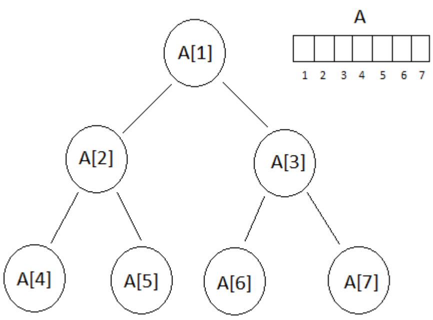

Figure 2.17: The complete binary tree with $n = 7$.

We see each parent with index i has children with indices $2i$ and $2i + 1$ and each child with index i has the parent with index $\lfloor i/2 \rfloor$. Note that odd nodes will be rounded down to the nearest integer. There are two steps to sort. First, we start from the parent with node i and compare with the children with node $2i$ and $2i+1$. If $\texttt{A}[\texttt{i}] > \texttt{A}[\texttt{2i}]$ or $\texttt{A}[\texttt{i}] > \texttt{A}[\texttt{2i} + \texttt{1}]$, we will swap $\texttt{A}[\texttt{i}]$ and $min\{\texttt{A}[\texttt{2i}], \texttt{A}[\texttt{2i} + \texttt{1}]\}$. Thereafter we use the recursive function for the node $min\{\texttt{A}[\texttt{2i}], \texttt{A}[\texttt{2i} + \texttt{1}]\}$ and so on until l. This process is called 'heapify down' [3], see Algorithm 2.

Next, we will use an insertion step which adds an element to the previous array and sort

Algorithm 2 Heapify down algorithm

Input: Node of parent i and the length l.
Output: Array which is updated.
Procedure:
Function `heapifydown`(i, l)

```
If 2i < l
    If A[2i] < A[i] || A[2i + 1] < A[i]
        If A[2i] < A[2i + 1]
         swap(A[i], A[2i])
          heapifydown(2i, l)
        else swap(A[i], A[2i + 1])
            heapifydown(2i + 1, l)
        end if;
    end if;
end if;
```

Algorithm 3 Heapify up algorithm

Input: Node i.
Output: Array which is updated.
Procedure:
Function `heapifyup`(i)

```
If i = 1
    return;
end if;
If A[i] < A[⌊i − 1⌋/2)
 swap(A[i], A[⌊i − 1⌋/2)
  heapifyup(A[⌊i − 1⌋/2])
end if;
```

it. Sorting array would compare parent and its children which is inserted. We will swap element if the value of the parent is more than the value of its children and use recursive function to move this data up until the parent is less than it. This step is called 'heapify up', see Algorithm 3.

These algorithms are modified to find the lower bound for this work. Next, we will describe the new method which relates to heap sort. We call it 'modified heap sort (MHS)'.

2.2.6 Modified heap sort

In general, there are two steps for heap sort which are heapify up and heapify down. We call it 'sorting step'. In this method, we apply heap sort algorithm and BSA to sort the bound. First, we generate two elements by the first element of the array in term of box (select the maximum length of box and generate it into two boxes). We call it 'generating step'. Next, we will use sorting step and then go to generating step and so on. That is, we set the bound to $f(z_1, z_2, \ldots, z_n) - C_1 d_1 - C_2 d_2 - \ldots C_n d_n$, where z_i are the midpoint of a box and d_i is box length, $\forall i \in \{1, 2, \ldots, n\}$ on the box domain $[a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_n, b_n]$.

Next, we generate 2 boxes by choosing the largest length and then we compare the values of the two boxes and select the minimum box to the first element and the maximum box to the new node. Next, we sort the bound from smallest to largest by sorting step and so on. The first element of the array will be the bound which we want to prove.

Example 2.37. Recall Example 2.31 with $f(x,y) = x^2 + y^2$ Let $a_1 \leq x_1 \leq b_1$ and $a_2 \leq x_2 \leq b_2$. We have $x_1 = \dfrac{a_1+b_1}{2}$, $y_1 = \dfrac{a_2+b_2}{2}$, $d_1 = \dfrac{b_1-a_1}{2}$, and $d_2 = \dfrac{b_2-a_2}{2}$. Let $\mathcal{B}_1$ be a box $[a_1,b_1] \times [a_2,b_2]$. Let $\mathtt{B}[1]$ be an array of the bound $f(z_1,z_2) - C_1d_1 - C_2d_2$, where $z_1 = \frac{a_1+b_1}{2}, z_2 = \frac{a_2+b_2}{2}$ with respect to the box $\mathcal{B}_1$. The generating bound of $\mathtt{B}[1]$ is the bounds of 2 boxes which are splitted by choosing the largest length with respect to $\mathcal{B}_1$. Let $\mathtt{A}$ be an array which collects the generating bounds and sorts them form smallest to largest. There are 2 main steps. First, the box $\mathcal{B}_1$ of $\mathtt{A}[1] = \mathtt{B}[1]$ will be divided or split into 2 boxes $\mathcal{B}_2, \mathcal{B}_3$ by subdividing the largest length and get the bounds $\mathtt{B}[2]$ and $\mathtt{B}[3]$ respectively. The minimum bound is set to $\mathtt{A}[1]$ and the maximum bound is set to be next to the last element of $\mathtt{A}$. The next step is to sort the elements of $\mathtt{A}$ by heap sort method. The minimum bound is sorted by heapify down and the maximum bound is sorted by heapify up.

Let z be the midpoint of box $\mathcal{B}_1 = [-1,3] \times [-2,4]$. We have $z = (1,1)$ and $d_1 = 4, d_2 = 6$. Let us set $\mathtt{A}[1] = \mathtt{B}[1] = f(z) - C_1d_1 - C_2d_2 = -70$.

Next, we select the largest length $d_2 = 6$ and generate two boxes $\mathcal{B}_2 = [-1,3] \times [-2,1]$ and $\mathcal{B}_3 = [-1,3] \times [1,4]$. In $\mathcal{B}_2$, $z = (1,-0.5)$ and $d_1 = 4, d_2 = 3$. We have $\mathtt{B}[2] = f(z) - C_1d_1 - C_2d_2 = -46.75$. In $\mathcal{B}_3$, $z = (1,2.5)$ and $d_1 = 4, d_2 = 3$. We have $\mathtt{B}[3] = f(z) - C_1d_1 - C_2d_2 = -40.75$. Since $\mathtt{B}[2] < \mathtt{B}[3]$, $\mathtt{A}[1] = \mathtt{B}[2]$ and $\mathtt{A}[2] = \mathtt{B}[3]$, see Figure 2.18. Hence, the bound is -46.75.

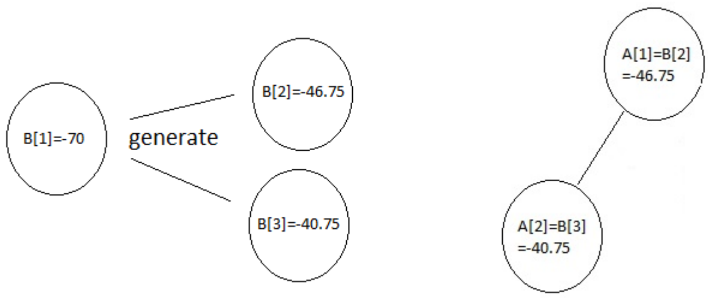

Figure 2.18: Step 1: $\mathtt{B}[1]$ is generated into $\mathtt{B}[2]$ and $\mathtt{B}[3]$ with `B[2] < B[3]`.

Step 2: $\mathtt{B}[2]$ is generated into 2 boxes $\mathcal{B}_4 = [-1,1] \times [-2,1]$ and $\mathcal{B}_5 = [1,3] \times [-2,1]$. In $\mathcal{B}_4$, $z = (0,-0.5)$ and $d_1 = 2, d_2 = 3$. We have $\mathtt{B}[4] = f(z) - C_1d_1 - C_2d_2 = -35.75$. In $\mathcal{B}_5$, $z = (2,-0.5)$ and $d_1 = 2, d_2 = 3$. We have $\mathtt{B}[5] = f(z) - C_1d_1 - C_2d_2 = -31.75$. Since $\mathtt{B}[4] < \mathtt{B}[5]$, $\mathtt{A}[1] = \mathtt{B}[4]$ and $\mathtt{A}[3] = \mathtt{B}[5]$. In the sorting step, we use heapify down for $\mathtt{B}[4]$ and heapify up for $\mathtt{B}[5]$. Since $\mathtt{B}[3] < \mathtt{B}[4]$, $\mathtt{A}[1] = \mathtt{B}[3]$ and $\mathtt{A}[2] = \mathtt{B}[4]$, see Figure 2.19. Hence, the bound is -40.75.

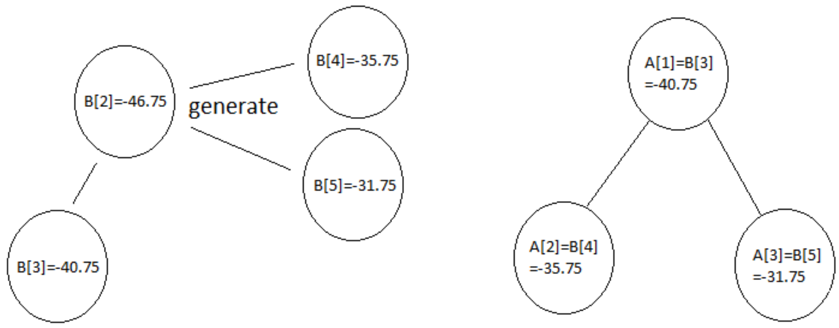

Figure 2.19: Step 2: B[2] is generated into B[4] and B[5] with B[4] < B[5].

Step 3: B[3] is divided into 2 boxes $\mathcal{B}_6 = [-1, 1] \times [1, 4]$ and $\mathcal{B}_7 = [1, 3] \times [1, 4]$.
$\mathcal{B}_6 : z = (0, 2.5)$ and $d_1 = 2, d_2 = 3$. We have B[6] $= f(z) - C_1d_1 - C_2d_2 = -29.75$.
$\mathcal{B}_7 : z = (2, 2.5)$ and $d_1 = 2, d_2 = 3$. We have B[7] $= f(z) - C_1d_1 - C_2d_2 = -25.75$. Since B[6] $<$ B[7], A[1] $=$ B[6] and A[4] $=$ B[7], see Figure 2.20. In the sorting step, we use heapify down for B[6] and heapify up for B[7]. Since B[4] $<$ B[6], A[1] $=$ B[4] and A[2] $=$ B[6], see Figure 2.21. Hence, the bound is -35.75.

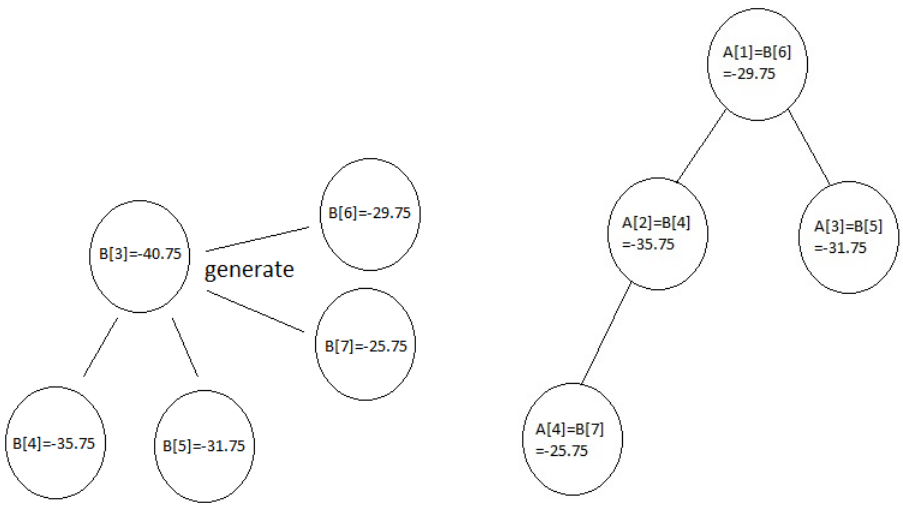

Figure 2.20: Step 3: B[3] is generated into B[6] and B[7] with B[6] $<$ B[7].

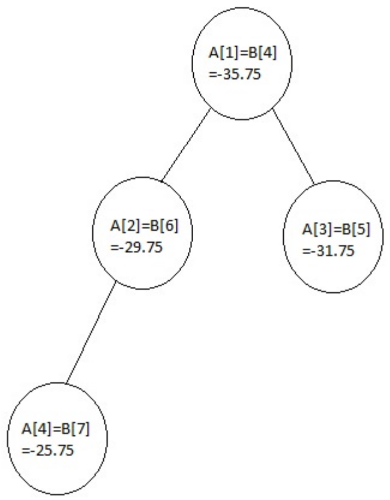

Figure 2.21: sorting step in Step 3

Step 4: B[4] is generated into 2 boxes $\mathcal{B}_8 = [-1,1] \times [-2,-0.5]$ and $\mathcal{B}_9 = [-1,1] \times [-0.5,1]$.
$\mathcal{B}_8 : z = (0,-1.25)$ and $d_1 = 2, d_2 = 1.5$. We have $\texttt{B[8]} = f(z) - C_1 d_1 - C_2 d_2 = -22.44$.
$\mathcal{B}_9 : z = (0,0.25)$ and $d_1 = 2, d_2 = 1.5$. We have $\texttt{B[9]} = f(z) - C_1 d_1 - C_2 d_2 = -23.94$. Since $\texttt{B[9]} < \texttt{B[8]}$, $\texttt{A[1]} = \texttt{B[9]}$ and $\texttt{A[5]} = \texttt{B[8]}$, see Figure 2.22. In the sorting step, we use heapify down for B[9] and heapify up for B[8]. Since $\texttt{B[5]} < \texttt{B[9]}$, $\texttt{A[1]} = \texttt{B[5]}$ and $\texttt{A[3]} = \texttt{B[9]}$, see Figure 2.23. Hence, the bound is -31.75.

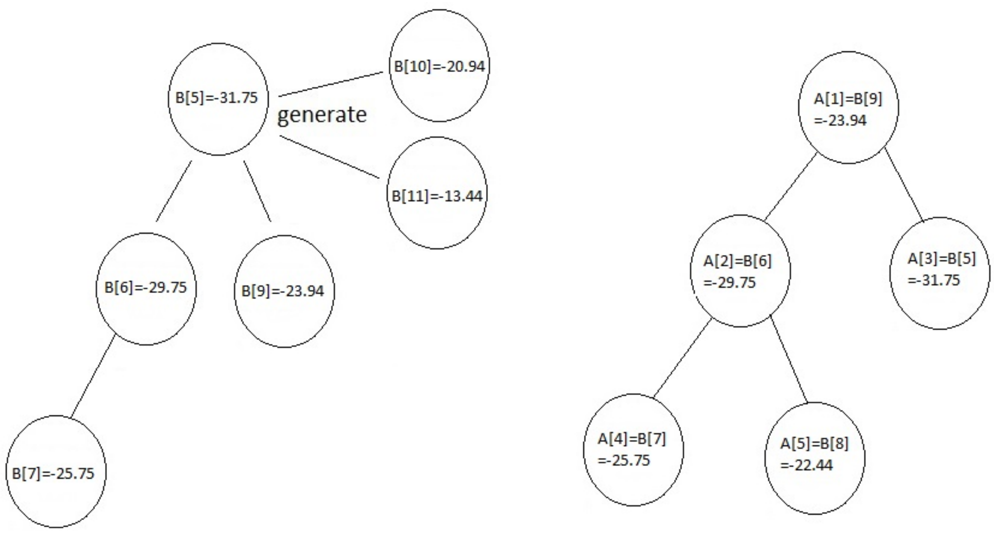

Figure 2.22: Step 4: B[4] is generated into B[8] and B[9] with $\texttt{B[9]} < \texttt{B[8]}$.

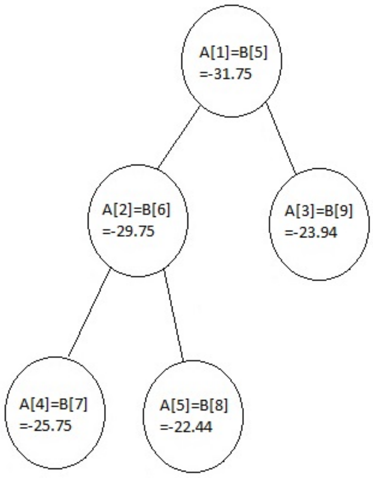

Figure 2.23: sorting step in Step 4

Step 5: B[5] is generated into 2 boxes $\mathcal{B}_{10} = [-1,1] \times [1,2.5]$ and $\mathcal{B}_{11} = [-1,1] \times [2.5,4]$. $\mathcal{B}_{10} : z = (0,1.75)$ and $d_1 = 2, d_2 = 1.5$. We have B[10] $= f(z) - C_1 d_1 - C_2 d_2 = -20.94$. $\mathcal{B}_{11} : z = (0,3.25)$ and $d_1 = 2, d_2 = 1.5$. We have B[11] $= f(z) - C_1 d_1 - C_2 d_2 = -13.44$. Since B[10] $<$ B[11], A[1] $=$ B[10] and A[6] $=$ B[11], see Figure 2.24. In the sorting step, we use heapify down for B[10] and heapify up for B[11]. Since B[6] $<$ B[10] and B[7] $<$ B[10], A[1] $=$ B[6] and A[2] $=$ B[7], see Figure 2.25. Hence, the bound is -29.75.

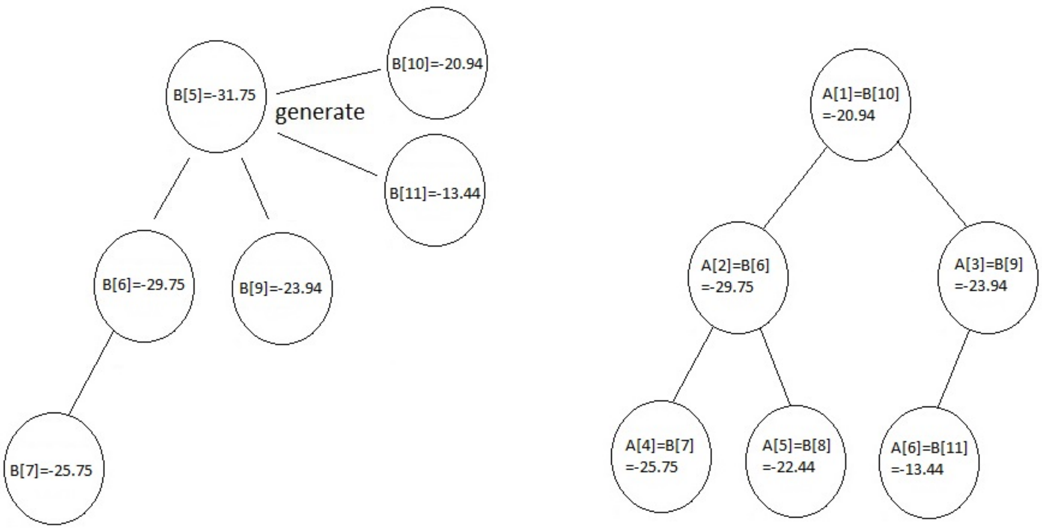

Figure 2.24: Step 5: B[5] is generated into B[10] and B[11] with B[10] $<$ B[11].

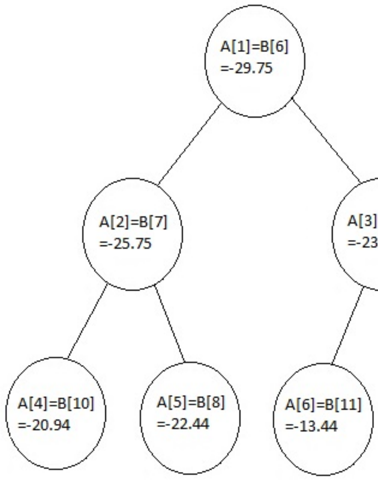

Figure 2.25: sorting step in Step 5.

Step 6: B[6] is generated into 2 boxes $\mathcal{B}_{12} = [1,3]\times[-2,-0.5]$ and $\mathcal{B}_{13} = [1,3]\times[-0.5,1]$. $\mathcal{B}_{12} : z = (2,-1.25)$ and $d_1 = 2, d_2 = 1.5$. We have $\texttt{B[12]} = f(z) - C_1d_1 - C_2d_2 = -18.44$ $\mathcal{B}_{13} : z = (2, 0.25)$ and $d_1 = 2, d_2 = 1.5$. We have $\texttt{B[13]} = f(z) - C_1d_1 - C_2d_2 = -19.94$. Since $\texttt{B[13]} < \texttt{B[12]}$, $\texttt{A[1]} = \texttt{B[13]}$ and $\texttt{A[7]} = \texttt{B[12]}$ see Figure 2.26. In the sorting step, we use heapify down for B[13] and heapify up for B[12]. Since $\texttt{B[7]} < \texttt{B[13]}$ and $\texttt{B[8]} < \texttt{B[13]}$, $\texttt{A[1]} = \texttt{B[7]}$ and $\texttt{A[2]} = \texttt{B[8]}$ see Figure 2.27. Hence, the bound is -19.94.

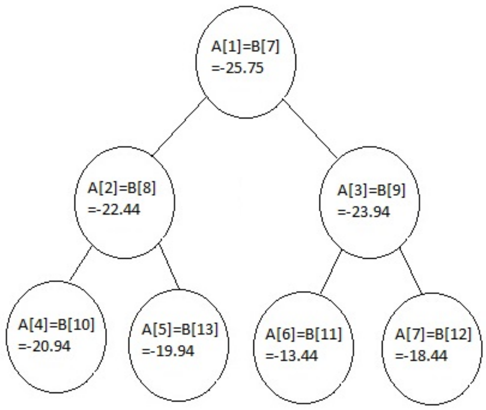

Figure 2.27: sorting step in Step 6

By Theorem 2.34, $f(x_1, y_1) \geq -19.94$ for all $x_1, y_1 \in [-1,3]\times[-2,4]$ and the $min(f(x_1, y_1)) = f(0, 0.25) = 0.0625$.

Example 2.38. Recall Example 2.33 with $f(x, y, z) = x^2 + y^2 + z^2$.

$z = (1,1,0)$ and $d_1 = 4, d_2 = 6, d_3 = 6$. We have $\texttt{A[1]} = \texttt{B[1]} = f(1,1,0) - C_1d_1 - C_2d_2 - C_3d_3 = -108$. We select $\max\{d_1, d_2, d_3\} = 6$. Since $d_2 = d_3 = 6$, we choose d_2. Thus, $d_2 = 3$. B[1] will be generated to $\texttt{B[2]} = f(1,-0.5,0) - C_1d_1 - C_2d_2 - C_3d_3 = -82.75$

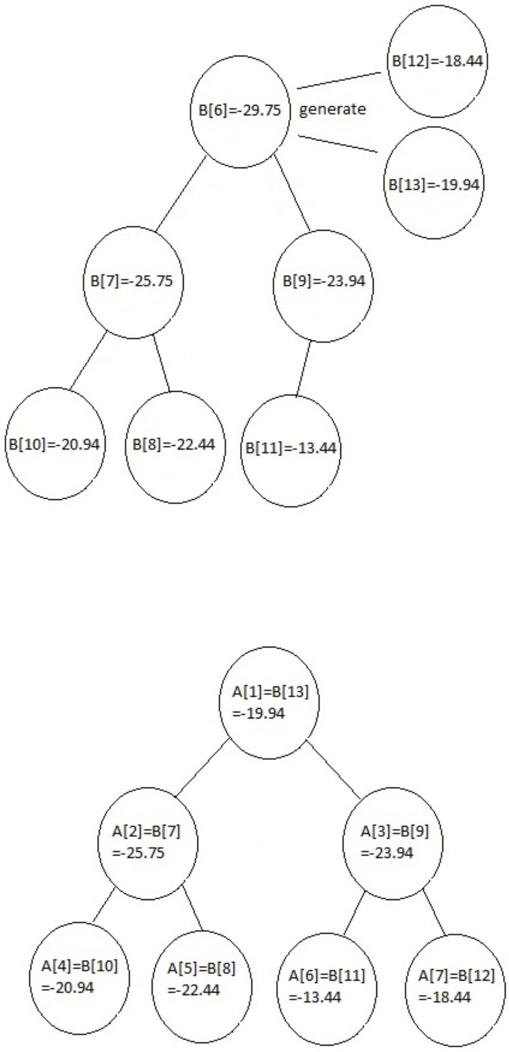

Figure 2.26: Step 6: `B[6]` is generated into `B[12]` and `B[13]` with `B[13] < B[12]`.

and `B[3]` $= f(1, 2.5, 0) - C_1 d_1 - C_2 d_2 - C_3 d_3 = -76.75$. We have `A[1]` $=$ `B[2]` and `A[2]` $=$ `B[3]`. Next, we choose $\max\{d_1, d_2, d_3\} = d_3 = 6$, So, $d_3 = 3$ and `B[2]` will be generated to `B[4]` $= f(1, -0.5, -1.5) - C_1 d_1 - C_2 d_2 - C_3 d_3 = -62.5$ and `B[5]` $= f(1, -0.5, 1.5) - C_1 d_1 - C_2 d_2 - C_3 d_3 = -62.5$. We have `A[1]` $=$ `B[4]` and `A[3]` $=$ `B[5]`. After that, `B[4]` will be sorted by heapify down and `B[5]` will be sorted by heapify up. Since `B[3]` $<$ `B[4]`, `A[1]` $=$ `B[3]` and `A[2]` $=$ `B[4]` and so on. We can see the result in Table 2.4.

Next, we will increase the number of iteration and see the results in the Table 2.4 and Matlab code in Supplementary Material.

bound	number of iterations	time (sec)
-0.10679	100000	23.3
-0.067696	200000	37.5
-0.051491	300000	50.1
-0.042799	400000	63.5
-0.036903	500000	76.8
-0.032499	600000	91.1
-0.029363	700000	108.4
-0.026847	800000	124.4
-0.024868	900000	139.0
-0.023273	1000000	153.8
-0.021844	1100000	168.9
-0.020527	1200000	183.6

Table 2.4: The bound by MHS

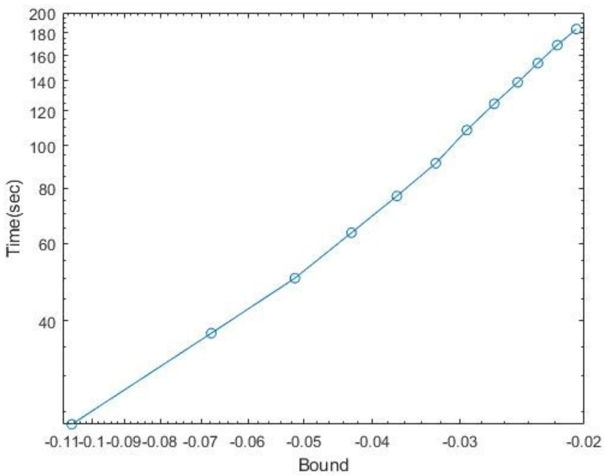

Figure 2.28: Graph of relationship between time and bound (log-log scale) in Table 2.3.

2.2.7 Discussion

We will compare MHS to the box search method for the program to run identical time, and whichever gives a better bound is the preferred method. We see that we do not need to give the expected bound in MHS. The bound is given by the program. However, this method takes a lot of time and there is a possibility of going out of memory if the number of iterations is large (see Section 3.3). Now, BSA is much quicker than MHS and we never run out of memory using this method. The drawback being we must give the expected bound, see Table 2.3 and 2.4.

In general, let $\mathcal{B}_1$ be a box $[a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_n, b_n]$. Let `B[1]` be an array which collect the bound $f(z_1, z_2, \ldots, z_n) - C_1 d_1 - C_2 d_2 - \ldots C_n d_n$ where $z_1 = \frac{a_1+b_1}{2}, z_2 = \frac{a_2+b_2}{2}, \ldots, z_n = \frac{a_n+b_n}{2}$ with respect to the box $\mathcal{B}_1$. The generating bound of `B[i]` is the bounds of 2 boxes which are generated by choosing the largest length of $\mathtt{B_i}$. Let $\mathtt{A}$ be an array which collects the generating bound and sorts them form smallest to largest. At the beginning of the process, we set $\mathtt{A}[1] = \mathtt{B}[1]$. Next, we choose $d = max\{d_1, d_2, \ldots, d_n\}$ and divide $\mathcal{B}_1$ into two boxes, $\mathcal{B}_2$ and $\mathcal{B}_3$. We can find the generating bound $\mathtt{B}[2]$ and $\mathtt{B}[3]$. The minimum of $\{\mathtt{B}[2], \mathtt{B}[3]\}$ will be in $\mathtt{A}[1]$ and the other one is next to the last element of $\mathtt{A}$, $\mathtt{A}[2]$. The minimum one will be sorted by heapify down and the other one will be sorted by heapify up, and so on.

In Table 2.4, we can see that if the number of iterations is increased, then the first element $\mathtt{A}[1]$ is close to a lower bound. However, the disadvantage of using this method is that it runs out of memory. Since the program collects all data when it is sorted, if it exceeds memory of computer, for example, 8GB in laptop, it will not run.

In 2007, Khandhawit and Sriswasdi [31] used the grid-search algorithm to improve the lower bound for unit arcs by applying Lipschitz bound to prove that

$$g(x) - \sum_{i=1}^{6} C_i d_i \geq 0.227498, \tag{2.12}$$

where $g(x)$ is an area of convex hull function of configuration of a line of length 1/2 with center $(0,0)$, a square with center at (x_1, y_1) and the angle of rotation α, and an equilateral triangle with center at (x_2, y_2) and the angle of rotation β such that $x = (x_1, y_1, \theta, x_2, y_2, \beta)$, C_i is a constant, d_i is a grid size for parameter $x_i, i = \{1, 2, 3, 4, 5, 6\}$, and $C_1 + C_2 + C_3 + C_4 = 2.44916, C_5 + C_6 = 0.49993, d_1 = d_2 = d_3 = d_4, d_5 = d_6$.

In our problem, we consider a function of an area of convex hull for some closed unit arcs. In Chapter 3, we will use BSA and MHS to solve the lower bound problem and compare the results each both.

Chapter 3

The lower bound is 0.0975

In this chapter, we combine geometric methods with numerical BSA to show that the area of convex cover for closed unit curves has area at least 0.0975 and at most 0.09763 by considering a convex hull of a circle, an equilateral triangle, and a rectangle with sides u and v and perimeter $2(u+v)=1$. We divide the problem into two parts as follow;

(i) To find the best rectangle with sides u and v and perimeter $2(u+v)=1$ by quick non-rigorous method. There is no proof in this section.

(ii) To prove that the area of convex hull of a circle, an equilateral triangle, and a rectangle from (i) is greater or equal to 0.0975.

This improves the previous optimal lower bound of 0.096694 [13] which used a circle and a curvilinear rectangle.

3.1 Numerical method to find the best rectangle

Let C be a circle with radius $r=\dfrac{1}{2\pi}$, R is a rectangle with sides $u\times v$ such that $u+v=\dfrac{1}{2}$, and T is an equilateral triangle of side $\dfrac{1}{3}$. For any fixed rectangle R, let $S(R)$ be the smallest area of convex hull of C, T, and R, where the "smallest" means with respect to all possible positions of C, T, and R on the plane. Obviously, $S(R)$ is a lower bound for area of convex cover for closed unit arcs. We then looking for rectangle R with perimeter 1 for which $S(R)$ is as large as possible to obtain as good lower bound as possible. So, we call the rectangle R which gives the best (that is, the largest) lower bound $S(R)$ to be "the best rectangle". In this section, we use Matlab to find the best rectangle. Therefore, we obtain the best lower bound for any rectangle.

Let C_1 be a regular 500-gon inscribed in C, such that the sides of R are parallel to some longest diagonals of C_1. Let us fix the center of the C_1 in origin. Let $H(C_1,R,T)$ be an area of convex hull for C_1, R and T. Let $\mathcal{T}$ be the set of all orientation-preserving motion T_1 which is a translation, and T_2 which is a composition of translation and rotation of the plane. Thus, we will solve the optimization problem:

$$\min_{T_1,T_2\in\mathcal{T}} H(C_1, T_1(R), T_2(T)) \tag{3.1}$$

There are two methods in Matlab to find a global solution for smooth function which are `GlobalSearch` and `MultiStat`. In our problem, we found that `MultiStat` is faster than `GlobalSearch`. Hence, we use `MultiStat` function to solve (3.1). Next, we use `fminbnd` function in Matlab to search R to make (3.1) is as large as possible, see in Figure 3.1.

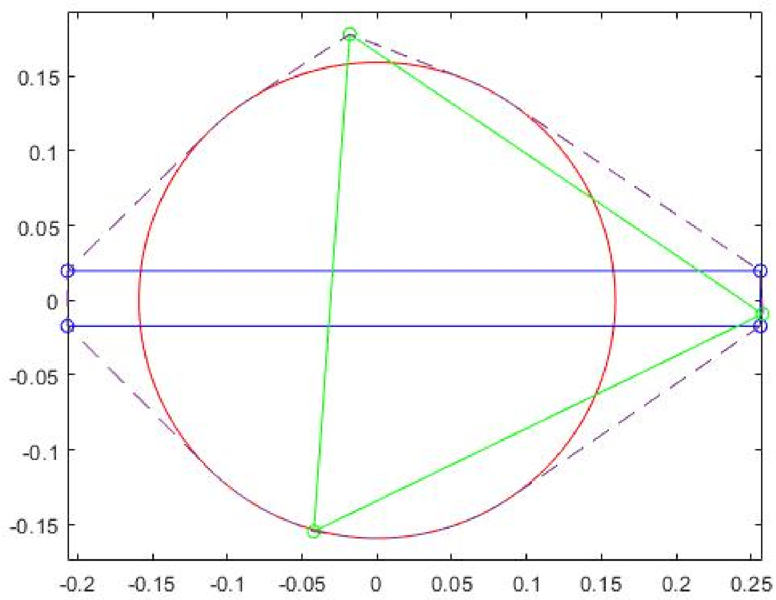

Figure 3.1: Optimal configuration for C_1, R, and T

By numerical results, we have $u = 0.0375$, which gives us the maximal value, see Matlab code in Supplementary Material. Now, we use this rectangle to prove the lower bound as follow

Theorem 3.1. *Any convex set S on the plane which can cover circle of perimeter 1, equilateral triangle of perimeter 1, and rectangle of size 0.0375×0.4625 (and perimeter 1) has area at least 0.0975.*

Let α_3 be the minimal area of convex cover for closed unit arcs.

Theorem 3.1 immediately implies that

$$0.0975 \leq \alpha_3,$$

which is an improvement comparing the best published lower bound 0.096694, as well as comparing an unpublished lower bound 0.096905.

If α' is the minimal area of a set which can cover circle, equilateral triangle, and *any* rectangle of perimeter 1, then Theorem 3.1 states that

$$0.0975 \leq \alpha' \leq \alpha_3.$$

Our computation shows that the *actual* value of α' is about

$$\alpha' \approx 0.09762742.$$

The bound in Theorem 3.1 is slightly weaker, because we need some margin to allow rigorous analysis of our numerical algorithms. We also show rigorously that $\alpha' \leq 0.09763$. This implies that, to improve lower bound for α_3 to 0.09763 and beyond, a different approach is required.

In Section 3.2, we shall prove Lipschitz conditions using geometric method.

3.2 Geometric analysis

Assume that C is a circle with radius $r = \dfrac{1}{2\pi}$, R is a rectangle with sides $u \times v$ such that $u + v = \dfrac{1}{2}$ and $u = 0.0375$, and T is an equilateral triangle with side $\dfrac{1}{3}$. Remark that C, R, and T are convex polygons in $\mathbb{R}^2$. Our aim is to prove that, no matter how C, R, and T are placed in $\mathbb{R}^2$, the area of their convex hull is at least 0.0975.

Let F be a regular 500-gon inscribed in the circle, such that the sides of R are parallel to some longest diagonals of F. We will call the union $X = F \cup R \cup T$ a *configuration*. For any configuration X, let $\mathcal{H}(X)$ denote the convex hull of X, and $\mathcal{A}(X)$ the area of $\mathcal{H}(X)$.

Let us put a coordinate center $(0, 0)$ at the center of F, and let X axis and Y axis be parallel to the longer and shorter sides of the rectangle, respectively. Let $R_0(x_1, y_1)$ be the center of R. We can orient the axes in such a way that $x_1 \geq 0$ and $y_1 \geq 0$. The vertices of R are defined by $R_1\left(x_1 - \left(\frac{\frac{1}{2}-u}{2}\right), y_1 + \frac{u}{2}\right)$, $R_2\left(x_1 + \left(\frac{\frac{1}{2}-u}{2}\right), y_1 + \frac{u}{2}\right)$, $R_3\left(x_1 + \left(\frac{\frac{1}{2}-u}{2}\right), y_1 - \frac{u}{2}\right)$, and $R_4\left(x_1 - \left(\frac{\frac{1}{2}-u}{2}\right), y_1 - \frac{u}{2}\right)$. Let $T_0(x_2, y_2)$ be the center of T and let T_1 be the vertex of T so that θ, the angle between X-axis and T_0T_1, has the smallest non-negative value.

Then $T_1(x_2 + \frac{\sqrt{3}}{9}\cos\theta, y_2 + \frac{\sqrt{3}}{9}\sin\theta)$, $T_2(x_2 + \frac{\sqrt{3}}{9}\cos(\theta + \frac{2\pi}{3}), y_2 + \frac{\sqrt{3}}{9}\sin(\theta + \frac{2\pi}{3}))$ and $T_3(x_2 + \frac{\sqrt{3}}{9}\cos(\theta + \frac{4\pi}{3}), y_2 + \frac{\sqrt{3}}{9}\sin(\theta + \frac{4\pi}{3}))$ are the vertices of triangle T.

In summary, the location of F, R, and T is fully described by 5 parameters: x_1, y_1, x_2, y_2, and θ.

Let $f : \mathbb{R}^5 \to \mathbb{R}$ be a function which maps vector $(x_1, y_1, x_2, y_2, \theta)$ to the area $\mathcal{A}(X)$ of the convex hull of the corresponding configuration X. Clearly, f is a continuous function.

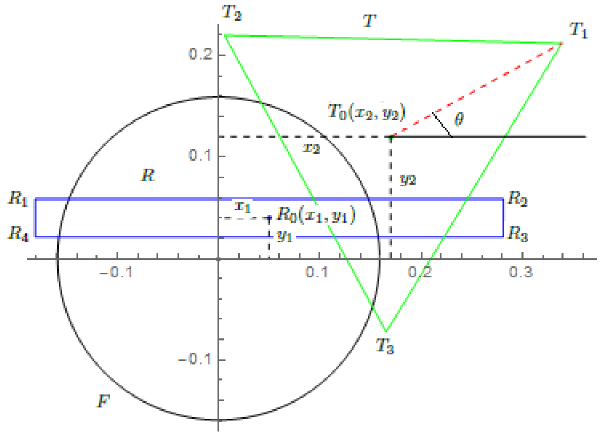

Figure 3.2: The configuration X which depends on $x_1, y_1, x_2, y_2, \theta$

Because F is a subset of C, Theorem 3.1 would follow from the inequality

$$f(x_1, y_1, x_2, y_2, \theta) > 0.0975, \quad \forall\, x_1, y_1, x_2, y_2, \theta.$$

The following result of Fary and Redei [11] plays an important role in our analysis

Lemma 3.2. *[11] Let S_1 and S_2 be two bounded convex sets in $\mathbb{R}^2$. If S_1 is translated along a line with constant velocity, then the volume of the convex hull of S_1 and S_2 is a convex function of time.*

Corollary 3.3. *Function f is a convex function in each of the coordinates x_1, y_1, x_2, y_2.*

Proof. Convexity of f with respect to x_1 follows from Lemma 3.2 with S_2 being the convex hull of F and T, while $S_1 = R$ moving along the X axis. Convexity of f with respect to y_1, x_2, and y_2 follows from Lemma 3.2 in a similar way. $\square$

Lemma 3.4. *Let Z be a region of points $z = (x_1, y_1, x_2, y_2, \theta)$ in $\mathbb{R}^5$ satisfying the inequalities*

$$0 \le x_1 \le 0.05,\ 0 \le y_1 \le 0.04,\ -0.17 \le x_2 \le 0.17,\ -0.13 \le y_2 \le 0.13,\ 0 \le \theta \le \frac{2\pi}{3}.$$

If $f(z) > 0.0975$ for all $z \in Z$, then in fact $f(z) > 0.0975$ for all $z \in \mathbb{R}^5$.

Proof. Let $\psi(x_1, y_1)$ be the area of the convex hull of F and R only. Lemma 3.2 implies that $\psi(x_1, y_1)$ is a convex function in both coordinates. Assume that $x_1 \ge 0.05$. By symmetry, $\psi(x_1, y_1) = \psi(x_1, -y_1)$, hence

$$\psi(x_1, y_1) \ge \psi(x_1, 0), \quad \forall x_1, y_1.$$

Also, by symmetry, $\psi(x_1, 0) = \psi(-x_1, 0)$, hence the convexity of $\psi(x_1, 0)$ implies that $\psi(x_1, 0) \geq \psi(0, 0)$, and that $\psi(x_1, 0)$ is non-decreasing in x_1 for $x_1 \geq 0$. Hence, $x_1 \geq 0.05$ implies that

$$\psi(x_1, y_1) \geq \psi(x_1, 0) \geq \psi(0.05, 0) > 0.0975,$$

where the last equality is verified directly. For similar reasons,

$$\psi(x_1, y_1) \geq \psi(0, y_1) \geq \psi(0, 0.04) > 0.0975,$$

whenever $y_1 \geq 0.04$.

From symmetry, we may assume that $x_1 \geq 0$ and $y_1 \geq 0$. Hence, either $f(z) > 0.0975$, or we may assume that $0 \leq x_1 \leq 0.05$, and $0 \leq y_1 \leq 0.04$.

Next, assume that $|x_2| \geq 0.17$. Then $\sqrt{x_2^2 + y_2^2} \geq |x_2| \geq 0.17$. Let C_1 be the incircle of T with radius $\frac{\sqrt{3}}{18}$ and center (x_2, y_2), see Figure 3.3.

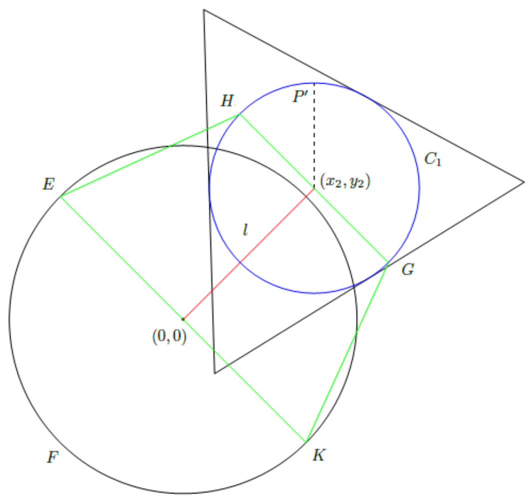

Figure 3.3: $\mathcal{A}(X)$ is bounded by the area of $EKGH, S(F)/2$ and the semicircle C_1

Let l be the line segment between $(0, 0)$ and (x_2, y_2). Next, let points $H, G \in C_1$ and $E, K \in F$ be such that line segments HG and EK are perpendicular to l, and pass through (x_2, y_2) and $(0, 0)$, respectively, see Figure 3.3. Then $EKGH$ is trapezoid with base lengths $|HG| = \frac{\sqrt{3}}{9}$ and $|EK| \geq 2r\cos\left(\frac{\pi}{500}\right)$, where $r = \frac{1}{2\pi}$. The area of F is $S(F) = 500\frac{r^2}{2}\sin\left(\frac{2\pi}{500}\right)$. Thus,

$$\mathcal{A}(X) > \frac{1}{2}\left(2r\cos\left(\frac{\pi}{500}\right) + \frac{\sqrt{3}}{9}\right)\left(\sqrt{x_2^2 + y_2^2}\right) + \frac{S(F)}{2} + \frac{\pi\left(\frac{\sqrt{3}}{18}\right)^2}{2} > 0.0975$$

To prove the bound for y_2, we need the following claim.

Claim 1. If there is a point $P' \in T$ with y-coordinate $y^* \geq 0.13 + \frac{\sqrt{3}}{18}$, then $f(z) >$

0.0975.

Indeed, let $R' = (0, -r)$, and R_1, R_2, R_3, R_4 be the vertices of the rectangle, see Figure 3.4.

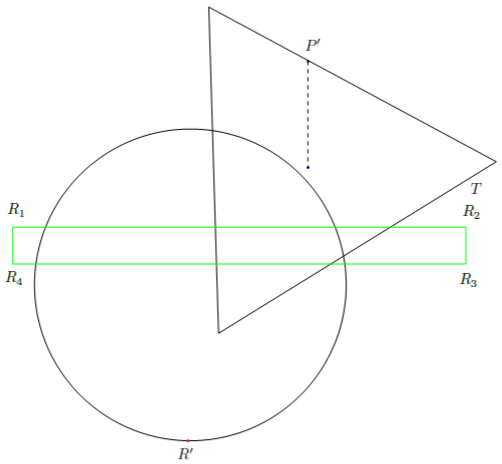

Figure 3.4: $\mathcal{A}(X)$ is bounded by the area of convex hull of R', R_1, R_2, R_3, R_4 and P'

Because $P', R_1, R_2, R_3, R_4, R' \in \mathcal{H}(X)$, we have $\mathcal{A}(X) > \mathcal{A}(\{P', R_1, R_2, R_3, R_4, R'\})$ $= u(\frac{1}{2} - u) + \frac{1}{2}(\frac{1}{2} - u)((y_2 + \frac{\sqrt{3}}{18}) + r - u) > 0.0975$, and the claim follows.

Now, assume that $y_2 \geq 0.13$. Let C_1 be the same circle as above, see Figure 3.3, and let P' be a point on C_1 with coordinates $(x_2, y_2 + \frac{\sqrt{3}}{18})$. Then $P' \in T$, and $f(z) > 0.0975$ by the claim.

The cases $x_2 \leq -0.17$ and $y_2 \leq -0.13$ are considered similarly. $\square$

Corollary 3.5. *Either $f(z) > 0.0975$, or $F \cup T \cup R$ is a subset of a rectangle with side lengths 0.386×0.644.*

Proof. By Lemma 3.4, we can assume that $z = (x_1, y_1, x_2, y_2, \theta) \in Z$. Let Y_1 and Y_2 be the points of configuration $X = F \cup T \cup R$ with the lowest and highest y-coordinates y_1^* and y_2^*, respectively. Because $0 \leq y_1 \leq 0.04$, Y_1 and Y_2 cannot belong to the rectangle R. If they both belong to F, then $y_2^* - y_1^* = \frac{1}{\pi} < 0.386$. If they both belong to the triangle, then $y_2^* - y_1^* \leq \frac{1}{3} < 0.386$. If $Y_1 \in F$ and $Y_2 \in T$, then $y_1^* = -\frac{1}{2\pi}$, and the inequality $y_2^* \leq 0.13 + \frac{\sqrt{3}}{18}$ follows from the Claim 1 in the proof of Lemma 3.4. Then $y_2^* - y_1^* \leq 0.13 + \frac{\sqrt{3}}{18} + \frac{1}{2\pi} < 0.386$.

Similarly, Let X_1 and X_2 be the points of configuration $X = F \cup T \cup R$ with the lowest and highest x-coordinates x_1^* and x_2^*, respectively. $z \in Z$ implies that neither X_1 nor X_2 belongs to F. If $X_1 \in T$ and $X_2 \in R$, then, by Lemma 3.4 $x_1^* \geq -0.17 - \frac{\sqrt{3}}{9}$, and $x_2^* \leq 0.05 + \frac{0.4625}{2}$, hence $x_2^* - x_1^* \leq 0.05 + \frac{0.4625}{2} + 0.17 + \frac{\sqrt{3}}{9} < 0.644$. $\square$

The following lemma established Lipschitz continuity of f in Z.

Lemma 3.6. *For every* $(x_1, y_1, x_2, y_2, \theta) \in Z$*, and any* $\epsilon_i \geq 0$*,* $i = 1, \ldots, 5$*,*

$$|f(x_1 + \epsilon_1, y_1 + \epsilon_2, x_2 + \epsilon_3, y_2 + \epsilon_4, \theta + \epsilon_5) - f(x_1, y_1, x_2, y_2, \theta)| \leq \sum_{i=1}^{5} \epsilon_i C_i,$$

with constants $C_1 = 0.212$*,* $C_2 = 0.322$ *,* $C_3 = 0.326$ *,* $C_4 = 0.398$*, and* $C_5 = 0.134$*.*

Proof. If function $g : \mathbb{R} \to \mathbb{R}$ is convex on $\mathbb{R}$ and

$$C = \max\left[\lim_{t\to-\infty} \frac{g(t)}{t}, \lim_{t\to+\infty} \frac{g(t)}{t}\right] < \infty,$$

then

$$|g(t + \epsilon) - g(t)| \leq C\epsilon, \quad \forall t, \forall \epsilon > 0.$$

Indeed, the inequality $C' := (g(t_0+\epsilon)-g(t_0))/\epsilon > C$ for some t_0 and ϵ, would, by convexity of g, imply that $g(t_0 + 2\epsilon) > g(t_0) + 2C'\epsilon$, and, by induction, $g(t_0 + 2^n\epsilon) > g(t_0) + 2^n C'\epsilon$, a contradiction with $\lim\limits_{t\to+\infty} \frac{g(t)}{t} \leq C < C'$. The inequality $(g(t_0 + \epsilon) - g(t_0))/\epsilon < -C$ leads to a contradiction for similar reasons.

Let us apply this result to convex function $g(x_1) = f(x_1, y_1, x_2, y_2, \theta)$, where y_1, x_2, y_2, θ are fixed. In this case,

$$\lim_{x_1\to-\infty} \frac{g(x_1)}{x_1} = \lim_{x_1\to+\infty} \frac{g(x_1)}{x_1} \leq \frac{0.386 + 0.0375}{2} < C_1,$$

where 0.386 comes from Corollary 3.5, while 0.0375 is the height of R, see Figure 3.5.

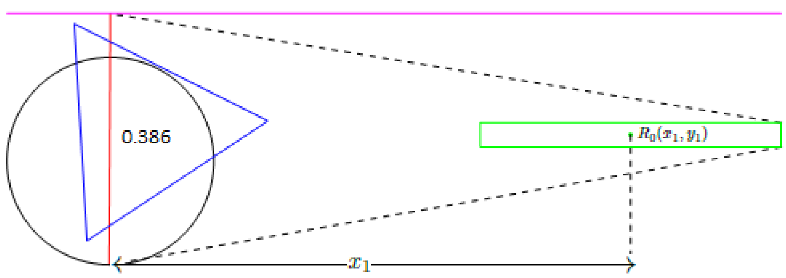

Figure 3.5: The ratio between $g(x_1)$ and x_1 when $x_1 \to +\infty$

Hence,

$$|f(x_1 + \epsilon_1, y_1, x_2, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_1\epsilon_1.$$

Similarly, with $g(x_2) = f(x_1, y_1, x_2, y_2, \theta)$ for fixed x_1, y_1, y_2, θ,

$$\lim_{x_2\to-\infty} \frac{g(x_2)}{x_2} = \lim_{x_2\to+\infty} \frac{g(x_2)}{x_2} \leq \frac{2r + 1/3}{2} < C_3,$$

see Figure 3.6, while with $g(y_2) = f(x_1, y_1, x_2, y_2, \theta)$,

$$\lim_{y_2 \to -\infty} \frac{g(y_2)}{y_2} = \lim_{y_2 \to +\infty} \frac{g(y_2)}{y_2} \leq \frac{1/3 + 0.4625}{2} < C_4,$$

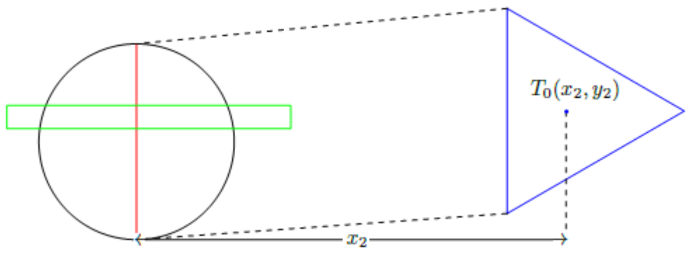

Figure 3.6: The ratio between $g(x_2)$ and x_2 when $x_2 \to +\infty$

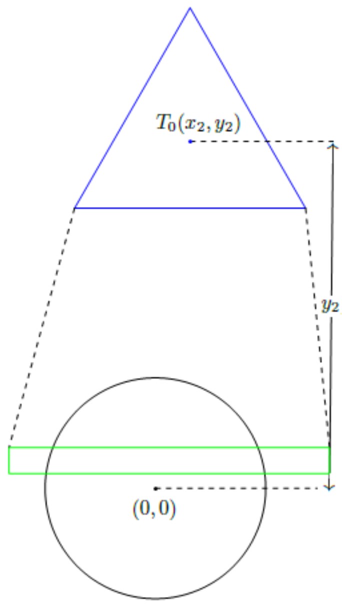

Figure 3.7: The ratio between $g(y_2)$ and y_2 when $y_2 \to +\infty$

see Figure 3.7. This implies that

$$|f(x_1, y_1, x_2 + \epsilon_3, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_3\epsilon_3,$$

and

$$|f(x_1, y_1, x_2, y_2 + \epsilon_4, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_4\epsilon_4.$$

The proof of similar bounds for the second and the fifth coordinates requires a different approach. For the second coordinate, we need to prove that

$$|f(x_1, y_1 + \epsilon_2, x_2, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_2\epsilon_2. \tag{3.2}$$

We claim that it is sufficient to prove 3.2 for $\epsilon_2 \in (0, \epsilon)$ for some $\epsilon > 0$, which can depend on x_1, y_1, x_2, y_2, and θ. Indeed, assume, by contradiction, that, for some y_1, 3.2 holds for $\epsilon_2 \in (0, \epsilon)$ but not for all $\epsilon_2 > 0$. Let ϵ^* be the supremum of all ϵ such that 3.2 holds for $\epsilon_2 \in (0, \epsilon)$. Then, by continuity of f, 3.2 also holds for $\epsilon_2 = \epsilon^*$, that is,

$$|f(x_1, y_1 + \epsilon^*, x_2, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_2\epsilon^*.$$

Applying 3.2 to $y_1' = y_1 + \epsilon^*$, we find that

$$|f(x_1, y_1 + \epsilon^* + \delta_2, x_2, y_2, \theta) - f(x_1, y_1 + \epsilon^*, x_2, y_2, \theta)| \leq C_2\delta_2$$

holds for all $\delta_2 \in (0, \delta)$ for some $\delta > 0$. But the last two inequalities imply that 3.2 holds for all $\epsilon_2 \in (0, \epsilon^* + \delta)$, a contradiction with the definition of ϵ^*.

We next prove 3.2 for $\epsilon_2 \in (0, \epsilon)$. Let R' with vertices $R_1'R_2'R_3'R_4'$ be the rectangle R which moved up by ϵ_2 in Y-axis's direction. Convex hulls $\mathcal{H}(R, F, T)$ and $\mathcal{H}(R', F, T)$ are polygons, and, by selecting ϵ sufficiently small, we can assume that all vertices of these polygons, which are not vertices of R and R', coincide. Then $\mathcal{A}(R', F, T) - \mathcal{A}(R, F, T)$ is bounded by the total area of three triangles, say $R_1'R_1D_1$, $R_2'R_2D_2$, and $R_2'R_2T^*$ which $D_1, D_2 \in F$ and $T^* \in T$, see Figure 3.8. Let h_1, h_2, h_3 be the height of $R_1'R_1D_1$, $R_2'R_2D_2$, $R_2'R_2T^*$, respectively. By Corollary 3.5, $|f(x_1, y_1 + \epsilon_2, x_2, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq \frac{1}{2}\epsilon_2(h_1 + h_2 + h_3) \leq \frac{1}{2}\epsilon_2(0.644) = 0.322\epsilon_2 = C_2\epsilon_2$.

Finally, we need to prove that

$$|f(x_1, y_1, x_2, y_2, \theta + \epsilon_5) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_5\epsilon_5. \tag{3.3}$$

To prove the bound for C_5, we need the following claim.

Claim 2. The diameter $d(\mathcal{F} \cup \mathcal{R})$ of $\mathcal{F} \cup \mathcal{R}$ is less than 0.46402.

Indeed, let $R_0 = (0.05, 0.04)$. We get $R_2 = (0.28125, 0.05875)$. Let $F_1 \in F$ be a point where $d(x, R_2)$ is maximum for all $x \in F$, see Figure 3.9. By direct calculation, $R_2F_1 = 0.4465 < |R_2R_4|$. Hence, the diameter of $\mathcal{F} \cup \mathcal{R}$ is $|R_2R_4| < 0.46402$.

Next, we will prove (3.3).

Let T' with vertices T_1', T_2', T_3' be the triangle T rotated around T_0 by angle ϵ_5. Then $|T_1T_1'| = 2|T_0T_1|\sin(\epsilon_5/2) < 2|T_0T_1|(\epsilon_5/2) = |T_0T_1|\epsilon_5 = \frac{\sqrt{3}}{9}\epsilon_5$. Similarly, $|T_2T_2'| = |T_3T_3'| < \frac{\sqrt{3}}{9}\epsilon_5$.

By selecting ϵ_5 sufficiently small, we can ensure that all vertices of polygons $\mathcal{H}(R, F, T)$ and $\mathcal{H}(R, F, T')$ coincide, except possibly the vertices of T and T'.

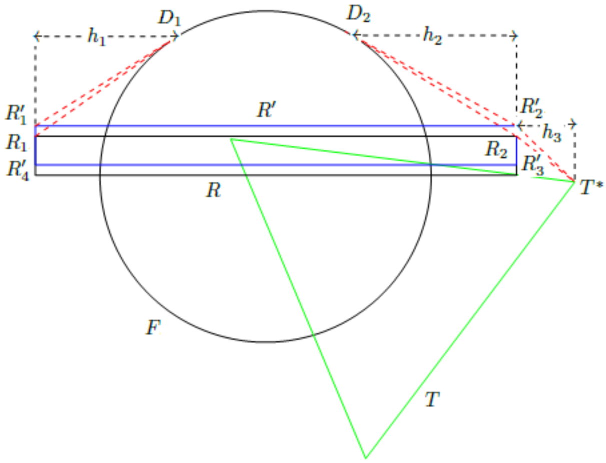

Figure 3.8: The three triangles which increase the area of the convex hull with ϵ_2

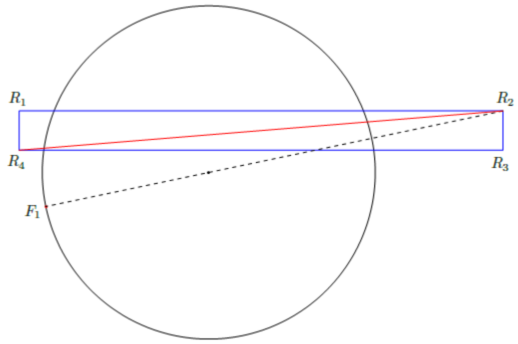

Figure 3.9: The longest diameter between R_2 and F

We assume that no vertices of the triangle T are adjacent in the convex hull $\mathcal{H}(R, F, T)$. Let X_1 and X_2 be the vertices of $\mathcal{H}(R, F, T)$ adjacent to T_1, X_3 and X_4 vertices of $\mathcal{H}(R, F, T)$ adjacent to T_2, and X_5 and X_6 vertices of $\mathcal{H}(R, F, T)$ adjacent to T_3, see Figure 3.10. Let us denote $S(ABC)$ the area of any triangle ABC.

Then area difference $|\mathcal{A}(R, F, T') - \mathcal{A}(R, F, T)|$ is equal to

$$|(S(T_1X_1X_2) + S(T_2X_3X_4) + S(T_3X_5X_6)) - (S(T_1'X_1X_2) + S(T_2'X_3X_4) + S(T_3'X_5X_6))|.$$

But $|(S(T_1X_1X_2) - S(T_1'X_1X_2)| = |\frac{1}{2}h_1|X_1X_2| - \frac{1}{2}h_2|X_1X_2|| = \frac{1}{2}|X_1X_2| \cdot |h_1 - h_2|$, where h_1 and h_2 are heights of triangles $T_1X_1X_2$ and $T_1'X_1X_2$, respectively, see Figure 3.11. But $|X_1X_2| < 0.46402$ by Claim 2, and $|h_1 - h_2| \leq |T_1T_1'| < \frac{\sqrt{3}}{9}\epsilon_5$, hence $|(S(T_1X_1X_2) - S(T_1'X_1X_2)| < 0.46402 \cdot \frac{\sqrt{3}}{18}\epsilon_5$. The same bound holds for $|(S(T_2X_3X_4) - S(T_2'X_3X_4)|$ and

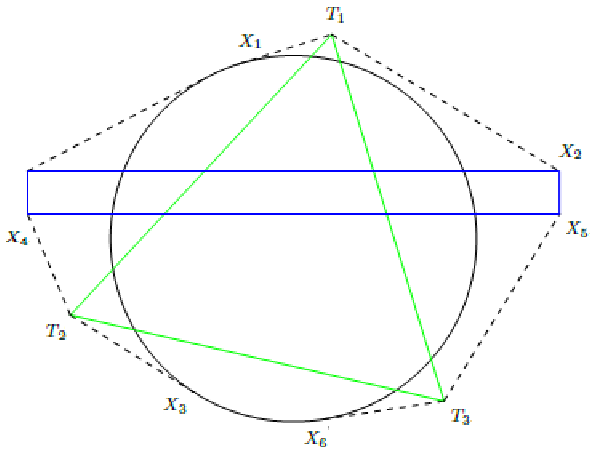

Figure 3.10: Polygon $\mathcal{H}(R,F,T)$ adjacent T_1, T_2, T_3

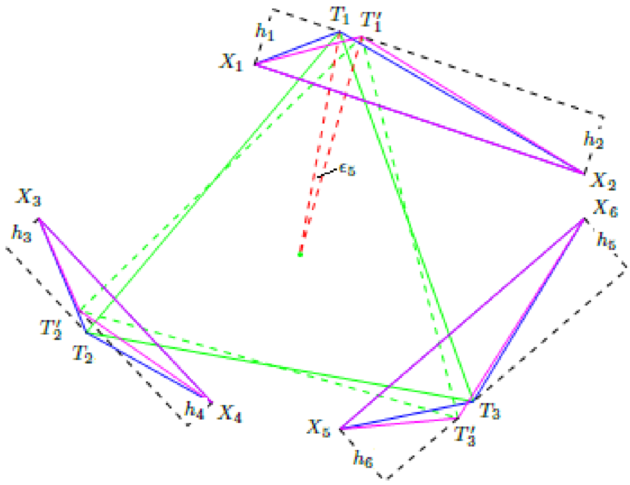

Figure 3.11: Six triangles which T rotated by angle ϵ_5

$|(S(T_3X_5X_6) - S(T_3'X_5X_6)|$. Hence,

$$|\mathcal{A}(R,F,T') - \mathcal{A}(R,F,T)| < 3 \cdot 0.46402 \cdot \frac{\sqrt{3}}{18}\epsilon_5 < 0.134\epsilon_5 = C_5\epsilon_5.$$

□

On the other hand, we can find the C_1, C_2, C_3, C_4 and C_5 by numerical method to find the maximum slope of graph of convex hull in each parameters. The results are $C_1 = 0.1624$, $C_2 = 0.20418$, $C_3 = 0.32483$, $C_4 = 0.38256$ and $C_5 = 0.03517$, see Figures 3.17-3.21.

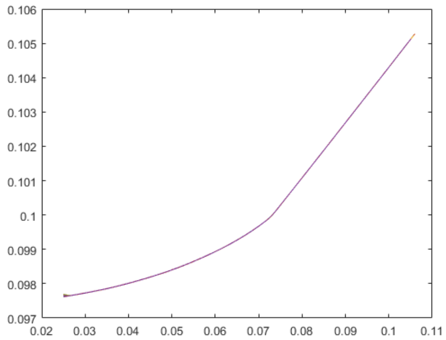

Figure 3.12: The maximum slope for C_1 is 0.1624

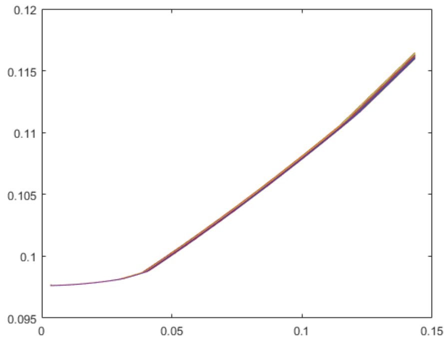

Figure 3.13: The maximum slope for C_2 is 0.20418

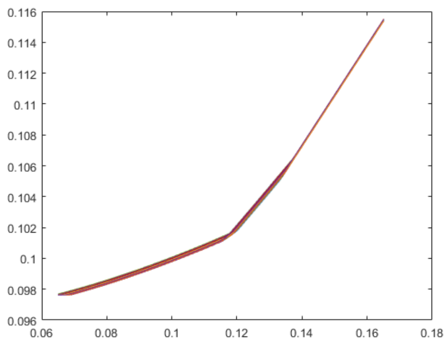

Figure 3.14: The maximum slope for C_3 is 0.32483

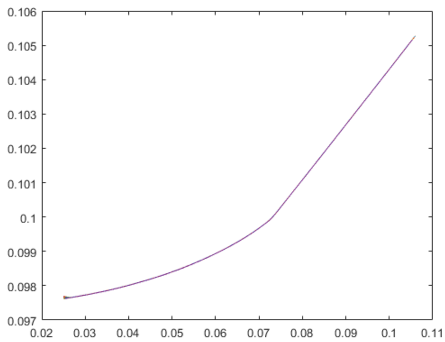

Figure 3.15: The maximum slope for C_4 is 0.38256

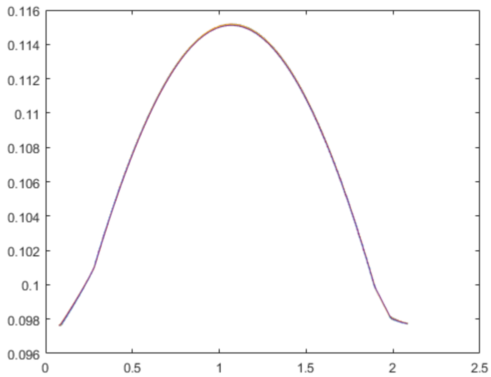

Figure 3.16: The maximum slope for C_5 is 0.03517

We can see that the constants C_1, C_2, C_3, C_4, C_5 by geometric proof are close to numerical results, meaning all constants shown in geometric proof are reliable. Next, we will use BSA and MHS to prove Theorem 3.1.

3.3 Computational results

First, we use BSA to prove that the minimal value of function $f(z) = f(x_1, y_1, x_2, y_2, \theta)$ in region Z defined in Lemma 3.4 is grater than 0.0975.

In general, let B be a box $[a_1, b_1] \times [a_2, b_2] \times [a_3, b_3] \times [a_4, b_4] \times [a_5, b_5]$ (that is, set of points $z = (x_1, y_1, x_2, y_2, \theta)$ such that $a_1 \leq x_1 \leq b_1$, $a_2 \leq y_1 \leq b_2$, $a_3 \leq x_2 \leq b_3$, $a_4 \leq y_2 \leq b_4$, $a_5 \leq \theta \leq b_5$. Let $z^* = (\frac{a_1+b_1}{2}, \frac{a_2+b_2}{2}, \frac{a_3+b_3}{2}, \frac{a_4+b_4}{2}, \frac{a_5+b_5}{2})$ be the center of the box). Then, if

$$f(z^*) - d_1C_1 - d_2C_2 - d_3C_3 - d_4C_4 - d_5C_5 \geq 0.0975, \tag{3.4}$$

where $d_i = \frac{b_i - a_i}{2}$, $i = 1, 2, 3, 4, 5$, then, by Lemma 3.4, $f(z) \geq 0.0975$ for all $z \in B$.

If the condition (3.4) does not hold for B, we will divide B into two sub-boxes B_1 and B_2, by dividing its maximal edge by half. For example, if $b_2 - a_2 \geq b_i - a_i$, $i = 1, 3, 4, 5$, then edge with length $b_2 - a_2$ is the maximal one, and we divide B into $B_1 = [a_1, b_1] \times [a_2, (a_2 + b_2)/2] \times [a_3, b_3] \times [a_4, b_4] \times [a_5, b_5]$ and $B_2 = [a_1, b_1] \times [(a_2 + b_2)/2, b_2] \times [a_3, b_3] \times [a_4, b_4] \times [a_5, b_5]$.

We then check (3.4) for B_1 and B_2. If it holds in both cases, then $f(z) \geq 0.0975$ for all $z \in B_1$ and for all $z \in B_2$, hence $f(z) \geq 0.0975$ for all $z \in B$. If (3.4) does not hold for B_1 (or for B_2, of for both), we divide the corresponding box by two sub-boxes, and proceed iteratively. $a_1 = 0, b_1 = 0.05, a_2 = 0, b_2 = 0.04, a_3 = -0.17, b_3 = 0.17, a_4 = -0.13, b_4 = 0.13, a_5 = 0, b_5 = \frac{2\pi}{3}$. Then we evaluate f in the box center $z^* = (0.025, 0.02, 0, 0, \pi/3)$ to check whether (3.4) holds. In this case, (3.4) reduces to

$$f(z^*) - \frac{0.05}{2}C_1 - \frac{0.04}{2}C_2 - \frac{0.17 - (-0.17)}{2}C_3 - \frac{0.13 - (-0.13)}{2}C_4 - \frac{2\pi/3}{2}C_5 > 0.0975,$$

or equivalently, to $f(z^*) > 0.3567$. However, the computation show that $f(z^*) \approx 0.10605 < 0.3567$, hence (3.4) does not hold. Hence, we need to subdivide B into B_1 and B_2. In this case, $b_1 - a_1 = 0.05$, $b_2 - a_2 = 0.04$, $b_3 - a_3 = 0.34$, $b_4 - a_4 = 0.26$, and $b_5 - a_5 = 2\pi/3 \approx 1.047$. Hence, $b_5 - a_5 > b_i - a_i$, $i = 1, 2, 3, 4$, and we divide B into $B_1 = [a_1, b_1] \times [a_1, b_2] \times [a_3, b_3] \times [a_4, b_4] \times [a_5, (a_5 + b_5)/2]$ and $B_2 = [a_1, b_1] \times [a_1, b_2] \times [a_3, b_3] \times [a_4, b_4] \times [(a_5 + b_5)/2, b_5]$. Then we repeat the above procedure for B_1 and for B_2, and proceed iteratively.

We use Matlab® R2016a to implement this algorithm, see Algorithm 1. The actual Matlab code is presented in Supplementary Material.

The program successfully verified the inequality $f(z) > 0.0975$ for all $z \in Z$ after $n = 7,180,439,126$ iterations. The program actually returned the minimal area 0.09762 for the optimal configuration with $x_1 = 0.0251$, $y_1 = 0.00258$, $x_2 = 0.0653$, $y_2 = 0.00542$, $\theta = 0.07989$ see Figure 6.1 and Appendix A.1.

Repeating the calculation for this particular configuration in Mathematica with actual circle instead of 500-gon shows that the optimal convex hull area is about

$$S_{min} \approx 0.09762742.$$

Next, we use powerful High Performance Computing (HPC) to run MHS, see Matlab code in Supplementary Material.

Bound	Number of Array	Time (hours)
0.097155	1×10^8	10.1
0.097246	2×10^8	20.5
0.097292	3×10^8	30.6
0.097322	4×10^8	41.0
0.097345	5×10^8	51.0
0.097362	6×10^8	60.4
0.097376	7×10^8	70.0
0.097386	8×10^8	80.6
0.097394	9×10^8	91.1
0.097401	10×10^8	101.7
0.097407	11×10^8	112.2
0.097412	12×10^8	122.8

Table 3.1: Table shows the results for MHS.

The program terminated when $n = 1{,}200{,}000{,}000$ because it was of out of memory. From Table 3.1, it shows that the bound is 0.097412 which is smaller than 0.0975. We can see that MHS collect many arrays to sort and stop when the memory is out. On the other hand, BSA does not collect all data. Thus, there is no out of memory for this method. In conclusion, BSA would be more flexible than MHS. Thus, we will use BSA to prove the bound. However, we should find a better expected bound such that the program does not run for long time.

3.4 Main Theorem

Theorem (Theorem 3.1)**.** *Any convex set S on the plane which can cover circle of perimeter 1, equilateral triangle of perimeter 1, and rectangle of size 0.0375×0.4625 (and perimeter 1) has area at least 0.0975.*

Proof. By numerical results, $f(z) > 0.0975$ for all $z \in Z$, where Z is defined in Lemma 3.4. By Lemma 3.4, this implies that $f(z) > 0.0975$ for all $z \in \mathbb{R}^5$, hence $\mathcal{A}(F, R, T) > 0.0975$. Because 500-gon F is the subset of circle C, $\mathcal{A}(C, R, T) \geq \mathcal{A}(F, R, T) > 0.0975$. □

Corollary 3.7. *Any convex cover for closed unit curves has area of at least 0.0975.*

Proof. Because every convex cover for closed unit curves cover the circle C, equilateral triangle T, and rectangle R of size 0.0375×0.4625, the claim follows from Theorem 3.1. □

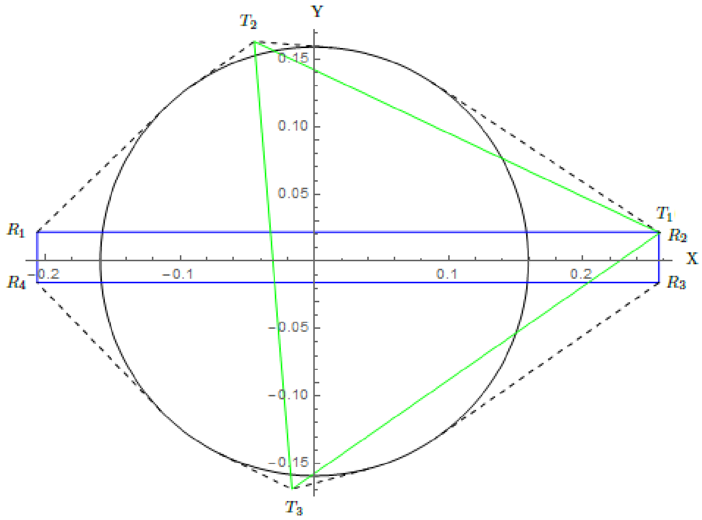

Figure 3.17: The convex hull of the configuration of the minimum area with 0.097627 acquired from the BSA

The following Theorem implies that this method (with circle, equilateral triangle, and rectangle of perimeter 1) cannot be used to improve the lower bound in Corollary 3.7 beyond 0.09763 see Matlab code in Supplementary Material.

Theorem 3.8. *For any rectangle R' with perimeter* 1*, there is a convex cover of R', C, and T with area at most* 0.09763*.*

Proof. Let l, w be the length and width of rectangle R' such that $l+w = \frac{1}{2}$ and $w \in [0, 0.25]$. Let F' be the regular 500-gon inscribed into the circle with $r' = \frac{sec(\frac{\pi}{500})}{2\pi}$. Then $C \subset F'$, and $\mathcal{H}(X) = \mathcal{H}(R, C, T) \subset \mathcal{H}(R, F', T)$. Thus, $\mathcal{A}(X) \le \mathcal{A}(R, F', T)$.

Let $f(w)$ denotes the minimal area of convex cover R, F', T.

Claim For any $\epsilon > 0$, $|f(w+\epsilon) - f(w)| \le 0.318\epsilon$. It suffices to prove the claim only for small ϵ. We will prove that $f(w) - f(w+\epsilon) \le 0.318\epsilon$, the proof for inequality $f(w+\epsilon) - f(w) \le 0.318\epsilon$ is similar. Let R'' be the rectangle with width $w+\epsilon$ and perimeter 1. Consider optimal configuration of R'', F', T, so that $f(\omega + \epsilon) = \mathcal{A}(R'', F', T)$. Let us put R' parallel to R'' as shown on Figure 3.18. This configuration is not necessary optimal, and, because f denotes the area of the *optimal* configuration, $f(w) \le \mathcal{A}(R', F', T)$. Hence, $f(w) - f(w+\epsilon) \le \mathcal{A}(R', F', T) - \mathcal{A}(R'', F', T)$.

Convex hulls $\mathcal{H}(R', F', T)$ and $\mathcal{H}(R'', F', T)$ are polygons, and, by selecting ϵ sufficiently small, we can assume that all vertices of these polygons, which are not vertices of R' and R'', coincides. Then $\mathcal{A}(R', F', T) - \mathcal{A}(R'', F', T)$ is bounded by the total area of triangles XQ_1R_2, YQ_2R_3, and rectangle $Q_1R_2R_3Q_2$, which is

$$\frac{1}{2}h_1\epsilon + \frac{1}{2}h_2\epsilon + Q_1Q_2\epsilon = \frac{\epsilon}{2}(h_1 + h_2 + 2Q_1Q_2)$$

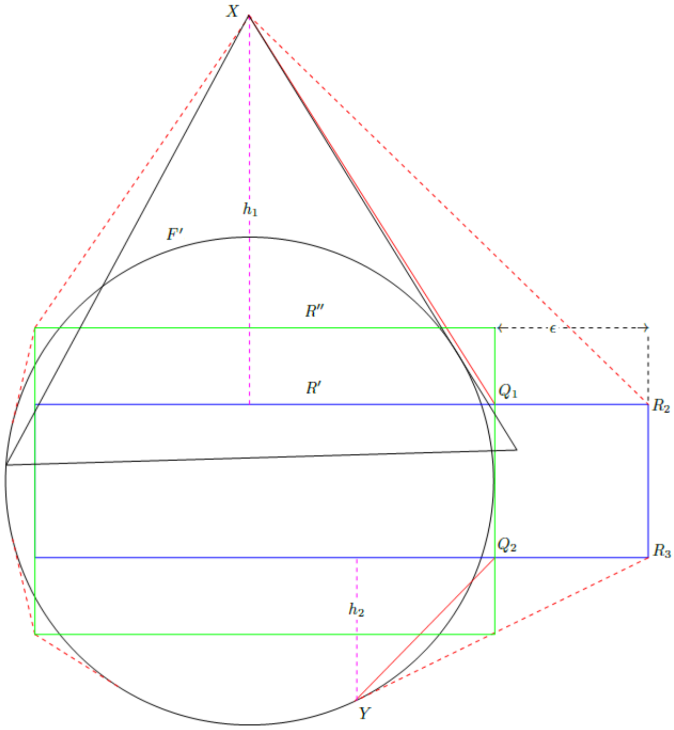

Figure 3.18: The configuration of R and R'

We have $Q_1Q_2 = w \le 0.25$, and, by Corollary 3.5, $h_1 + h_2 + Q_1Q_2 \le 0.386$. Hence,

$$f(w) - f(w+\epsilon) \le \mathcal{A}(R', F', T) - \mathcal{A}(R'', F', T) \le \frac{\epsilon}{2}(0.386 + 0.25) = 0.318\epsilon,$$

which proves the claim.

To verify inequality $f(w) < 0.09763$ at some *specific* point w, it is not necessary to find the *optimal* configuration of R', F', and T. In suffices just to find *some* configuration with $\mathcal{A}(R', F', T) < 0.09763$, and then conclude that $f(w) \le \mathcal{A}(R', F', T) < 0.09763$. This makes the numerical verification simple.

We will verify inequality $f(w) < 0.09763$ for w belonging to some finite set $W = \{w_1, w_2, \dots, w_N\}$, where $0 \le w_1 \le w_2 \le \dots \le w_N \le 0.25$ are points to be specified below. By the claim, inequality $f(w_i) < 0.09763$ implies that $f(w) \le 0.09763$ in the whole interval $w \in [w_i - d_i, w_i + d_i]$, where $d_i = (0.09763 - f(w_i))/0.318$.

We will select set W in such a way that intervals $[w_i - d_i, w_i + d_i]$, $i = 1, 2, \dots, N$ cover the whole interval $[0, 0.25]$. In other words, $w_1 - d_1 < 0$, $w_N + d_N > 0.25$, and

$$w_i + d_i < w_{i+1} - d_{i+1}, \quad i = 1, 2, \dots, N-1.$$

Set W with $N = 772$ points with this property is presented in the Appendix A.2. For example, $w_1 = 0.00020$, $w_2 = 0.0034$, $w_3 = 0.0086$, and so on, $w_{772} = 0.2415$.

$\square$

To conclude, we used the geometric method and numerical BSA to show that the optimal area of convex cover for a circle of perimeter 1, an equilateral triangle of side 1/3, and rectangle of perimeter 1 is between 0.0975 and 0.09763. Next, we will find an area of convex hull of centrally symmetric objects.

Chapter 4

Systematic search

We want to improve the bound found in Chapter 3, which is 0.0975. We divided this problem into two steps as follow

(i) Find the set of objects S for which the area of convex hull of S is as large as possible;

(ii) To prove that the lower bound for the area of convex hull of $S \geq B$.

In this chapter, we use numerical search to find the set of objects S. It contains neither proof nor any form of rigorous analysis. However, this chapter is important to understand how we selected the objects in S. In Chapter 5, we will prove our results found in Chapter 4.

The organization of this chapter is as follows. First, we consider the convex hull of n centrally symmetric objects. Next, we use a systematic numerical search to find the optimal area for 2 and 3 objects.

4.1 Centrally symmetric objects

In this section, we want to find optimal area by considering n centrally symmetric objects. We shall find that the best lower bound for 2 centrally symmetric objects which have the smallest area of convex hull are curvilinear rectangle and a circle of perimeter 1, previously having a lower bound of 0.096694 [13].

Definition 4.1. ([19]. p.1) Let an arc $\tau \in \mathbb{R}^2$ and c is a center of τ. τ is called "centrally symmetric object" if the point $2c - x$ is on τ, for all points x on τ.

Example 4.2. A circle, a line and a rectangle are centrally symmetric objects, but an equilateral triangle and an isosceles trapezoid are not centrally symmetric objects.

By Lemma 3.2, the area of convex hull of both objects which coincide is the smallest. Thus, we conjecture that the smallest area for n centrally symmetric should coincide. Next, we will increase n starting from $3, 4, 5$ and so on. If this conjecture is true, we will fix the centre of n centrally symmetric at $(0, 0)$ and increase n to improve the lower bound.

However, Lemma 3.2 cannot be applied for n centrally symmetric objects when $n \geq 3$. We will give counterexample in next section.

4.1.1 Numerical method to find the bound for two centrally symmetric objects

Let A_1 be a regular 1000-gon inscribed in the circle of perimeter one. Let A_2 be any object with vertices $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$. Let $A_2^{'}$ be a convex hull of $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ and $(-x_1, -y_1), (-x_2, -y_2), \ldots, (-x_n, -y_n)$ scale to have perimeter one. By symmetry with origin, A_2' is a centrally symmetric object. We use Algorithm 4 to find the area of convex hull of A_1,$A_2^{'}$ as follow

---

Algorithm 4 Algorithm to find the area of convex hull of $A_1, A_2^{'}$

Input: The coordinates of A_2 which are $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
Output: Area of the convex hull of A_1 and $A_2^{'}$**.**
Procedure:
Function $a =$per$(x_1, y_1, x_2, y_2, x_3, y_3, \ldots, x_n, y_n)$
1: Construct centrally symmetric object
$K :=$ `convhull`$(x_1, y_1, x_2, y_2, \ldots, x_n, y_n, -x_1, -y_1, -x_2, -y_2, \ldots, -x_n, -y_n)$;
2: $P :=$ perimeter of K
3: Scale : $x_1^{'} := \dfrac{x_1}{P}, x_2^{'} := \dfrac{x_2}{P}, \ldots, y_n^{'} := \dfrac{y_n}{P}$
4: Set $A_2^{'} :=$ `convhull`$(x_1^{'}, y_1^{'}, x_2^{'}, y_2^{'}, \ldots,\ x_n^{'} y_n^{'}, -x_1^{'}, -y_1^{'}, -x_2^{'}, -y_2^{'}, \ldots, -x_n^{'}, -y_n^{'})$
5. Find the area of convex hull of A_1 and $A_2^{'}$: $a =$ `convhull`(A_1, A_2').

end Procedure

Example 4.3. Let $(1, 3), (-2, 1), (-3, -2)$ be the vertices of A_2. Thus, the convex hull of $(1, 3), (-2, 1), (-3, -2)$ and $(-1, -3), (2, -1), (3, 2)$ is the hexagon with vertices $(1, 3), (-2, 1), (-3, -2)$ and $(-1, -3), (2, -1), (3, 2)$ and the perimeter is approximately 18. The vertices of $A_2^{'}$ are $(1, 3)/18, (-2, 1)/18, (-3, -2)/18, (-1, -3)/18, (2, -1)/18$, and $(3, 2)/18$.

Next, we will find the maximum of per$(x_1, y_1, x_2, y_2, x_3, y_3, \ldots, x_n, y_n)$ to improve the lower bound for 2 objects. If we assume that this problem is smooth and try to use `Globalsearch` or `MultiStat`, we found that the maximum value does not converge to solution. Thus, we will use the methods to find a global solution for non-smooth problem. There are many methods in Matlab to find the solution. We try to use every method and found that `patternsearch` is the fastest function to find a solution in this problem. To find the maximum, we will use `patternsearch` function to find the maximum automatically. Let n be 500, hence A_2 has 500 points, and let us randomize this 1000 times, giving us 1000 difference combinations of A_2. The area is about 0.096693 which is close to the optimal of 0.096694 [13] see Figure 4.1, now we see in the experiment the lower bound is close to the optimal bound however we cannot improve the lower bound for two centrally symmetric objects, see Matlab code in Supplementary Material.

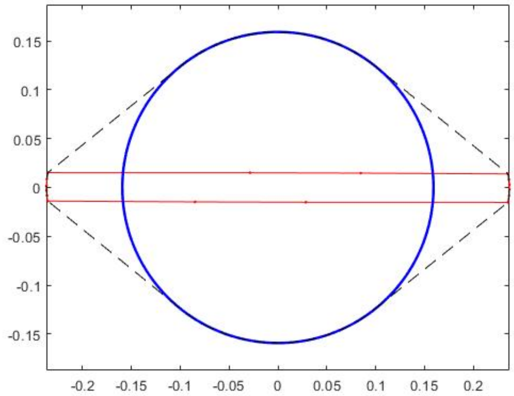

Figure 4.1: The maximum area of convex hull of A_1 and A_2' with the convex hull area of 0.0966693

4.1.2 Numerical method to find the bound for three centrally symmetric objects

Since Lemma 3.2 does not guarantee the optimal solution for $n > 2$ centrally symmetric objects, we observe three centrally symmetric objects such that the centers of these objects coincide. Let B_1, B_2, B_3 be centrally symmetric objects with perimeter one. Let $G(B_1, B_2, B_3)$ be an area of convex hull for B_1, B_2 and B_3. Initially, let us set B_1, B_2, B_3 be a regular 50-gon inscribed in the circle of perimeter one. Let $\mathcal{T}$ be the set of all orientation-preserving motion T which is a rigid rotation of the plane. Thus, we will solve the optimization problem:

$$\min_{T \in \mathcal{T}} G(B_1, T_1(B_2), T_2(B_3)) \tag{4.1}$$

Next, we use `patternseach` function in Matlab to find B_1, B_2, B_3 which make (4.1) the convex hull as large as possible.

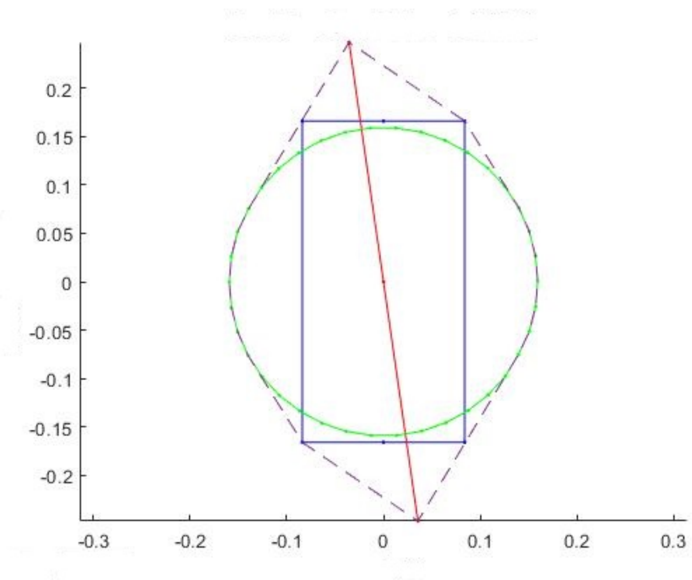

Figure 4.2: The maximum area of convex hull of B_1, B_2 and B_3 which has the area of 0.100685

From `patternsearch`, the maximum area is 0.0100685 where B_1 is a circle, B_2 is 0.1727×0.3273 rectangle and B_3 is a line of length $\frac{1}{2}$. Next, we will show that Lemma 3.2 can not be applied for the centrally symmetric n objects when $n \geq 3$ by giving an counterexample, see Matlab code in Supplementary Material.

4.1.3 Lemma 3.2 cannot be extended to $n \geq 3$

Let B_1 be a circle of perimeter one, B_2 be a 0.1727×0.3273 rectangle, B_3 be a line of length $\frac{1}{2}$ (see Figure 4.2). Let $C_0(0,0)$ be the center of B_1, (x_1, y_1) be a center point of B_2, (x_2, y_2) be a center point of B_3, and θ be a angle of rotation around a center of B_3. Clearly, $x_1, y_1 \geq 0$ and $0 \leq \theta \leq \pi$. Let us fix B_1 at $C_0(0,0)$. We find that if $x_1 = 0.00946, y_1 = 0.00934, x_2 = 0.08254, y_2 = -0.02247$ and $\theta = 0.18859$, then $\mathcal{A}(B_1, B_2, B_3) = 0.100465 < 0.100685$ see Figure 4.3. Thus, we cannot use coincide property to improve a lower bound.

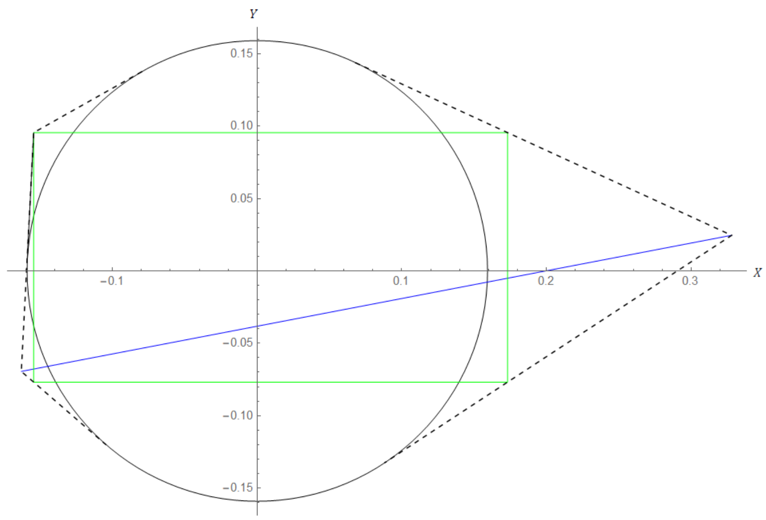

Figure 4.3: the optimal configuration for 3 centrally symmetric objects with the convex hull area of 0.100465

Although we cannot apply Lemma 3.2 to find the optimal area of convex hull of 3 centrally symmetric objects, the counterexample has an area of 0.100465 which is better than the previous bound in Chapter 3. Next, we will find the optimal area of 3 non centrally symmetric objects. We shall see this is the optimal bound for any 3 objects and use BSA to prove the bound.

4.2 A systematic search for shape

In this section, we will use numerical method to find the optimal area of convex hull of two and three arbitrary polygons to improve the previous bound in Chapter 3. For each specific set of polygons, we find a minimum area of convex hull with respect to all translations and rotations, and then we modify the shapes of the polygons to make this minimal area as large as possible. This results in a minimax problem with highly non-smooth and non-convex objective function. We used Matlab `patternsearch` function with random initial data to try to find the global optimal value in various cases.

Let α be the area of the smallest cover for closed unit curves. Let us start with 2 polygons. Let $F(X_1, X_2)$ be an area of convex hull for polygons X_1 and X_2 . Let Γ be the set of all orientation-preserving motion T which is a composition of translation and rotation of the plane. Let X_1, X_2 be any two polygons with unit length. Thus, the solution of the optimization problem:

$$\min_{T\in\mathcal{T}} F(X_1, T(X_2)) \tag{4.2}$$

is the lower bound for α.

Next, we will find polygons X_1 and X_2 for which this lower bound is as large as possible.

Let N_1 and N_2 be a number of vertices in polygons X_1 and X_2 respectively. We assume that N_1 and N_2 are fixed but X_1 and X_2 can vary. Let $\chi(N)$ be the set of all convex polygons with N vertices and unit perimeter. We consider optimization problem

$$b(N_1, N_2) = \max_{X_1 \in \mathcal{X}(N_1), X_2 \in \mathcal{X}(N_2)} \min_{T \in \mathcal{T}} F(X_1, T(X_2)). \tag{4.3}$$

It is clear that for any N_1, N_2,

$$b(N_1, N_2) \leq \alpha,$$

In other word, $b(N_1, N_2)$ is a lower bound for α.

4.2.1 Numerical results for 2 objects

First, we construct a Matlab function `NN2per` to solve the minimization problem (4.2) see Algorithm 5. The input of the function is $2N_1 + 2N_2$ coordinates of vertices of X_1 and X_2. The function first calculates the perimeters of the polygons, and scale them to make the perimeters to be equal to 1. Then it applies the motion T to X_2, which is described by three parameters: vector of translation (x_1, y_1) and angle of rotation θ_1, and use Matlab function `convhull` to estimate $F(X_1, T(X_2))$. In Chapter 3, we found that `Multistat` is the best function to find a global minimum for our minimization problem. Hence, we will use function `MultiStart` in Matlab to find the minimum of $F(X_1, T(X_2))$ by fixing polygon X_1 and translate and rotate polygon X_2 over parameters x_1, y_1, θ_1.

Next, we apply a `patternsearch` function in Matlab and Algorithm 6 to solve the maximization problem (4.3) by searching maximal possible output of function `NN2per` for fixed N_1 and N_2.

We repeat this procedure for various small values of N_1 and N_2. Specifically, we consider the cases of a line and a triangle (2+3 vertices), two triangles (3+3 vertices), a triangle and a quadrilateral (3+4 vertices), two quadrilaterals (4+4 vertices), and so on. The results are shown in Table 4.1 and Matlab code is provided in Supplementary Material.

Algorithm 5 **Algorithm to find minimum of** $F(X_1, X_2)$

Input:v **-** the point which are from `N2object` function.
Output: `ar` - the minimum area of convex hull of X_1, X_2.
Procedure:
Function `ar=NN2per(v)`
1: To find convex hull of both objects by using function `convhull`
2: To find perimeter of both objects : a, b- the perimeter of first and second object respectively.
3: To scale both objects to perimeter one by a and b, say X_1^* and X_2^*
4: To calculate the minimum area of convex hull of X_1^* and X_2^* of the configuration described by parameters x_1, y_1, θ by using function `MultiStart` with initial condition $(0, 0, 0)$ and random $n = 80$ points

end Procedure

Algorithm 6 **Algorithm to find minimax of** $F(X_1, X_2)$

Input:N_1, N_2 **-** the number of point of first and second object respectively.
Output: $fval$ - the minimum area of convex hull of X_1, X_2, x - the points which get $fval$ (final point).
Procedure:
Function $[x, fval] = N2object(N_1,N_2)$
1: To set the initial points of X_1, X_2 which are regular N_1-gon and N_2-gon, respectively.
2: To use `patternsearch` function to search maximin of $F(X_1, X_2)$
end Procedure

Note that we use the random $n = 80$ points in `MultiStat` because if n is a large number, the program will run slow and we found that if $n < 80$, the result is not a global minimum.

Type ($n + m$ points)	Optimal area	Time (sec)
$2 + 3$	0.072169	1.6
$3 + 3$	0.072375	2.2
$3 + 4$	0.085377	3.9
$4 + 4$	0.085377	4.8
$3 + 5$	0.087902	4.8
$5 + 5$	0.087902	6.6
$9 + 19$	0.095790	66.6
$9 + 50$	0.096605	70.0
$11 + 50$	0.096605	71.1

Table 4.1: The numerical series for 2 objects.

From the Table 4.1, it can be seen that the maximum lower bound we found is 0.0966053 when two polygons are 11-gon and 50-gon, see Figure 4.12. It is close to 0.0966675 [13] which is the best known lower bound for 2 objects. Furthermore, we found that the optimal

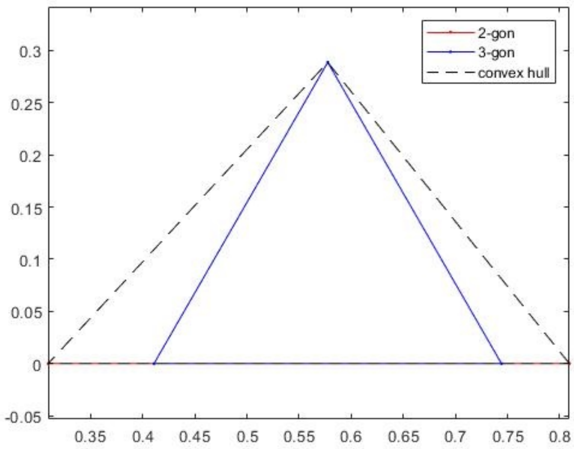

Figure 4.4: The optimal configuration of 2+3 points with the convex hull area of 0.072169.

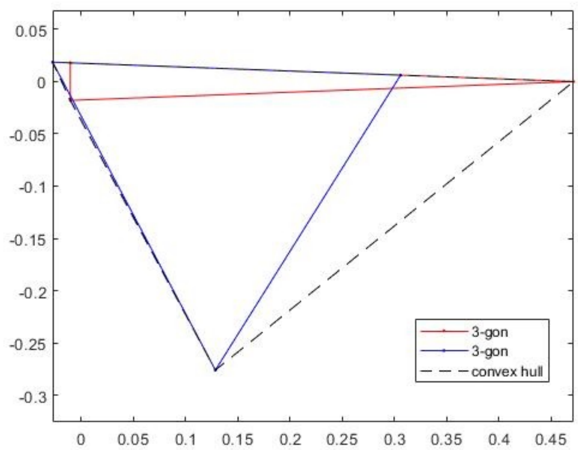

Figure 4.5: The optimal configuration of 3+3 points with the convex hull area of 0.072375.

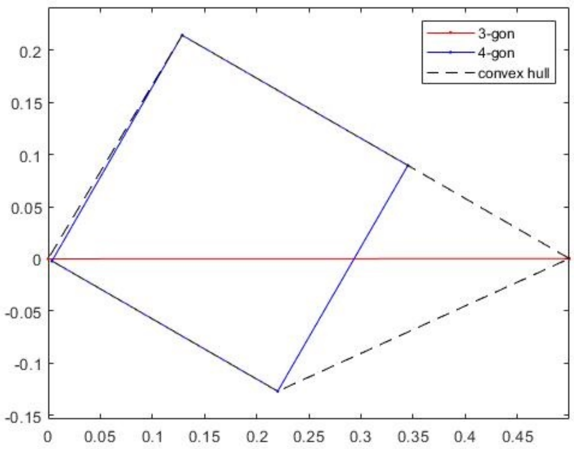

Figure 4.6: The optimal configuration of 3+4 points with the convex hull area of 0.085377.

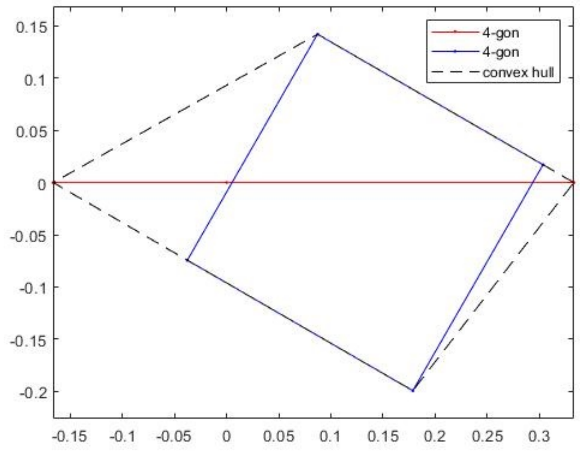

Figure 4.7: The optimal configuration of 4+4 points with the convex hull area of 0.085377.

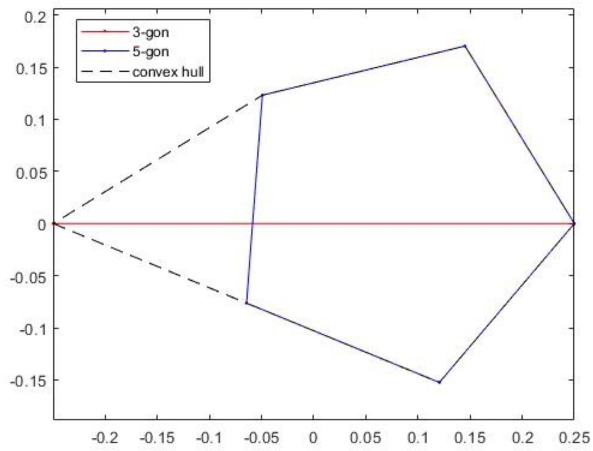

Figure 4.8: The optimal configuration of 3+5 points with the convex hull area of 0.087902.

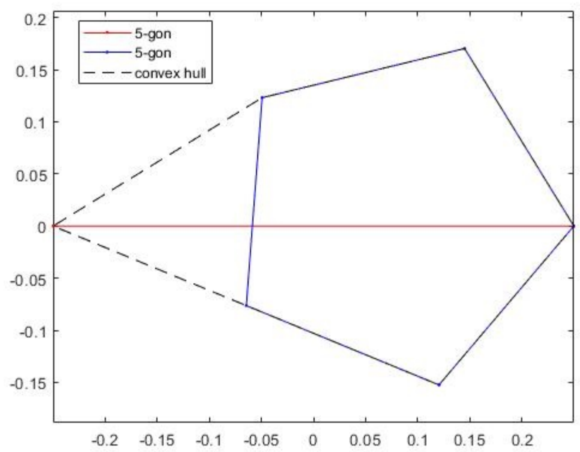

Figure 4.9: The optimal configuration of 5+5 points with the convex hull area of 0.087902.

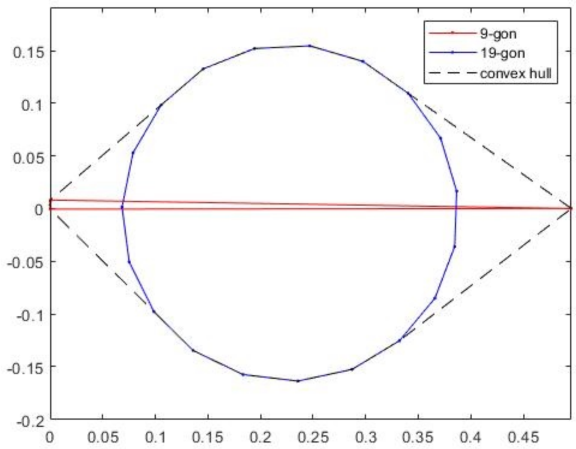

Figure 4.10: The optimal configuration of 9 + 19 points with the convex hull area of 0.095790.

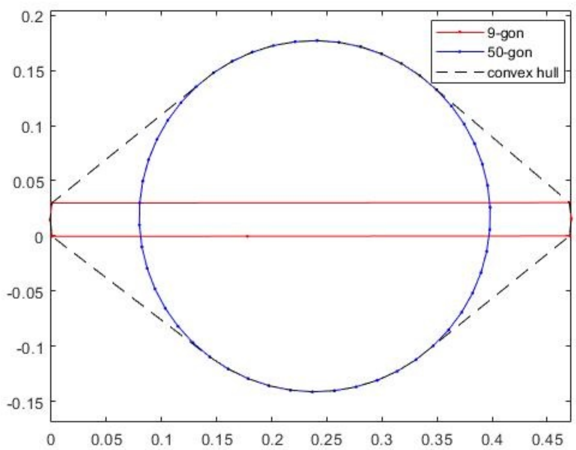

Figure 4.11: The optimal configuration of 9 + 50 points with the convex hull area of 0.0966051.

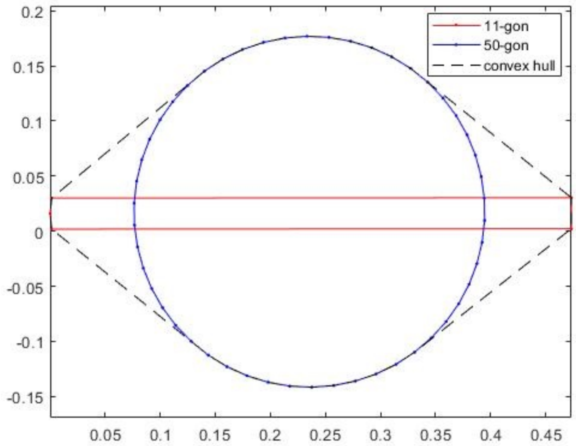

Figure 4.12: The optimal configuration of 11 + 50 points with the convex hull area of 0.0966053.

area for two arbitrary centrally symmetric objects is 0.0966693 which is close to 0.0966675 [13]. Hence, it may be concluded that we did not improve the existing best lower bound for two objects but almost recovered the best result in the literature.

4.2.2 Numerical results for 3 objects

Since we can not improve the bound by considering the smallest area of convex hull of two objects, we will use the same idea as for 2 objects to find the smallest area of convex hull of three objects. Let X_3 be a third polygon with N_3 vertices which can be translated and rotated. Let $F(X_1, T_1(X_2), T_2(X_3))$ be an area of convex hull of X_1, X_2, and X_3. Hence, we will solve the maximin optimization problem :

$$b(N_1, N_2, N_3) = \max_{X_1 \in \mathcal{X}(N_1), X_2 \in \mathcal{X}(N_2), X_3 \in \mathcal{X}(N_3),} \min_{T_1 \in \mathcal{T}, T_2 \in \mathcal{T}} F(X_1, T_1(X_2), T_2(X_3)), \qquad (4.4)$$

Again, $b(N_1, N_2, N_3)$ is the lower bound for α.

We applied the idea of searching the maximin function of two objects to solve the maximin optimization problem (4). Let us fix the center of X_1at $(0, 0)$. Let (x_1, y_1, θ_1) and (x_2, y_2, θ_2) be translation and rotation vector for X_2 and X_3, respectively. Thus, we have six parameters to find minimum of $F(X_1, T_1(X_2), T_2(X_3))$ and $2N_1 + 2N_2 + 2N_3$ coordinates to search the maximin optimization problem (4.4).

Let us start to solve (4.4) for small simple vertices (N_1, N_2, N_3 are small numbers) and increase the number of vertices until we obtain the acceptable value. We start the cases of line and two triangles $(2+3+3)$, three triangles $(3+3+3)$, two triangles and quadrilaterals $(3 + 4 + 4)$ and so on. We use the Algorithm 7 and 8 to solve (4.4) and we can see the results in Table 4.2 and Matlab code is provided in Supplementary Material.

Algorithm 7 Algorithm to find minimax of $F(X_1, X_2, X_3)$

Input: N_1, N_2, N_3 - the number of point of first and second object respectively.
Output: `fval` - the minimum area of convex hull of X_1, X_2, X_3, `x` - the points which get `fval` (final point).
Procedure:
Function `[x,fval]=N3object(`N_1,N_2, N_3`)`
1: To set the initial points of X_1, X_2, X_3 which are regular N_1-gon, N_2-gon and N_3-gon, respectively.
2: To use `patternsearch` function to search maximin of $F(X_1, X_2, X_3)$
end Procedure

Algorithm 8 Algorithm to find minimum of $F(X_1, X_2, X_3)$

Input: v - the point which are from `N2object` function.
Output: ar - the minimum area of convex hull of X_1, X_2, X_3.
Procedure:
Function `ar=NN3per(v)`
1: To find convex hull of both objects by using function `convhull`
2: To find perimeter of both objects : a, b, c- the perimeter of first and second object respectively.
3: To scale the both objects to perimeter one by a, b and c, say $X_1^*, X_2^*$and X_3^*
4: To calculate the minimum area of convex hull of X_1^*, X_2^* and X_3^* of the configuration described by parameters x_1, y_1, θ_1 and x_2, y_2, θ_2 by using function `MultiStart` with initial condition $(0, 0, 0, 0, 0, 0)$ and $n = 80$ points

end Procedure

Type ($n + m$ points)	Optimal area	Time (sec)
$2 + 3 + 3$	0.072169	7.2
$3 + 3 + 3$	0.072419	8.1
$2 + 3 + 4$	0.087867	9.9
$3 + 3 + 4$	0.087887	10.7
$3 + 3 + 5$	0.088478	14.3
$10 + 10 + 10$	0.093546	57.9
$2 + 4 + 50$	0.100403	90.9
$4 + 4 + 50$	0.100407	140.5
$7 + 7 + 50$	0.100417	284.0

Table 4.2: The numerical series for 3 objects

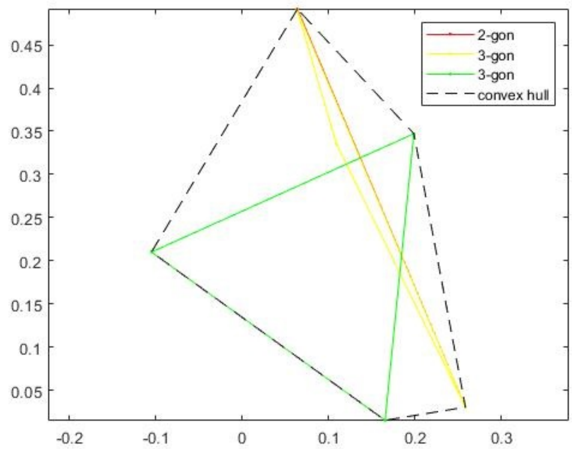

Figure 4.13: The optimal configuration of $2 + 3 + 3$ points with the convex hull area of 0.072169.

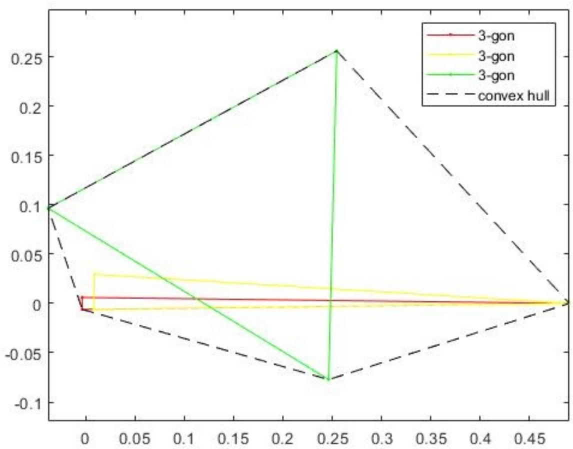

Figure 4.14: The optimal configuration of $3 + 3 + 3$ points with the convex hull area of 0.072419.

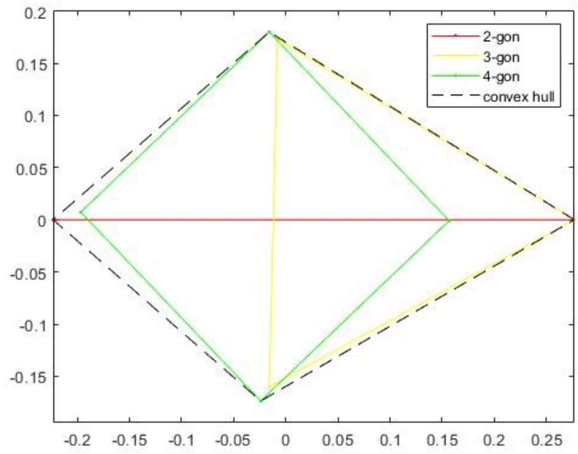

Figure 4.15: The optimal configuration of $2 + 3 + 4$ points with the convex hull area of 0.087867.

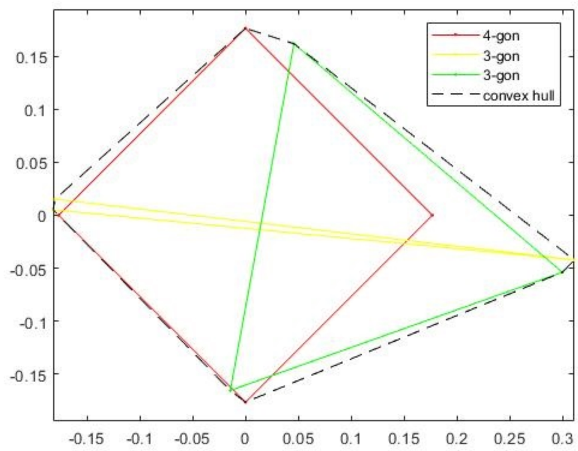

Figure 4.16: The optimal configuration of $3 + 3 + 4$ points with the convex hull area of 0.087887.

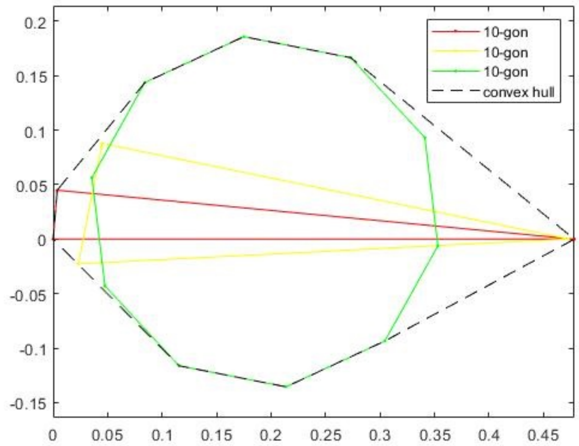

Figure 4.17: The optimal configuration of $10 + 10 + 10$ points with the convex hull area of 0.093546.

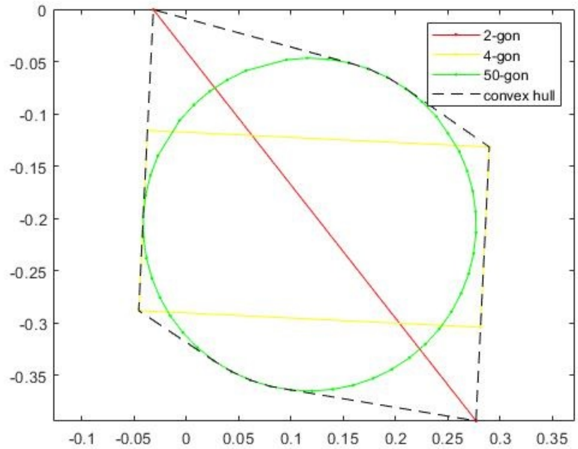

Figure 4.18: The optimal configuration of $2 + 4 + 50$ points with the convex hull area of 0.100403.

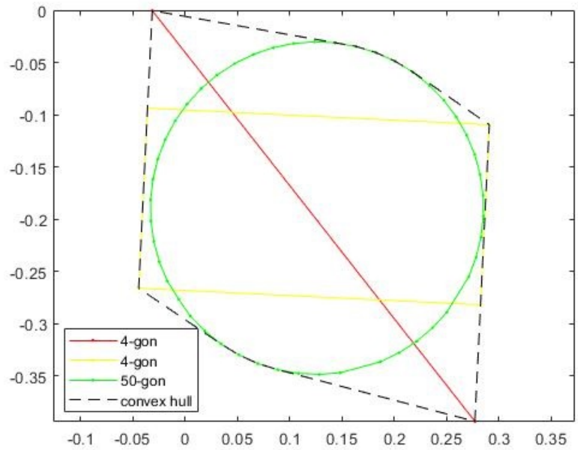

Figure 4.19: The optimal configuration of $4+4+50$ points with the convex hull area of 0.100407.

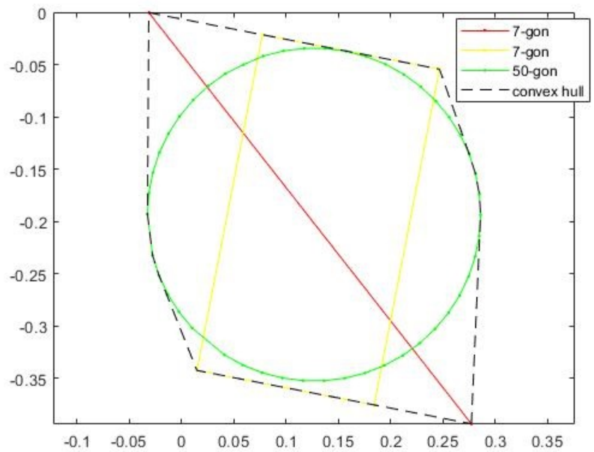

Figure 4.20: The optimal configuration of $7+7+50$ points with the convex hull area of 0.100417.

From Table 4.2, it can be seen that the maximum lower bound we found is 0.100417 when X_1 is a regular 50-gon, X_2 is a line of length 0.5, and X_3 is a rectangle of size 0.1727×0.3273 see Figure 4.20. This area is close to the area of counterexample of three centrally symmetric objects and is better than 0.0975 which is the bound in Chapter 3. It is observed that the three objects which achieve the maximin value should be a line of length 0.5, a rectangle of size 0.1727×0.3273 and a circle of perimeter one. Next, we will try to find three objects in the form: circle, line, and n-gon. We fix X_1 is a regular 500-gon and X_2 is a line as a 2-gon and then increase a number of vertices of X_3 from $N_3 = 3$ to $N_3 = 10$ to find the maximin of F. The results are presented in Table 4.3 and Matlab code is provided in Supplementary Material.

Type (n points)	Optimal area	Time (hours)
3	0.097043	3.8
4	0.1003	8.0
5	0.100304	9.3
6	0.100374	15.3
7	0.100386	22.2
8	0.100390	22.6
9	0.100418	33.1
10	0.100473	34.3

Table 4.3: The numerical series for 3 objects when 2 objects are fixed.

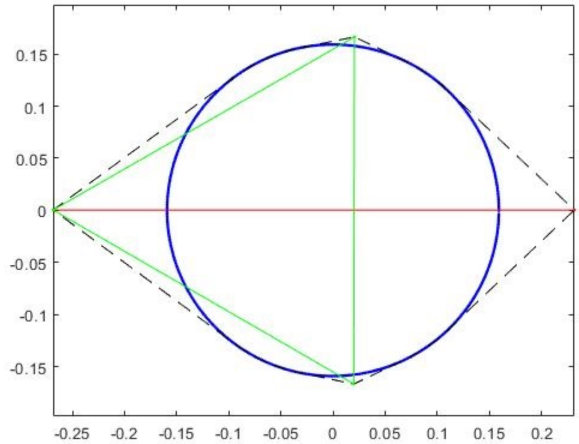

Figure 4.21: The optimal configuration of 500-gon, a line and a triangle (green) with the convex hull area of 0.097043.

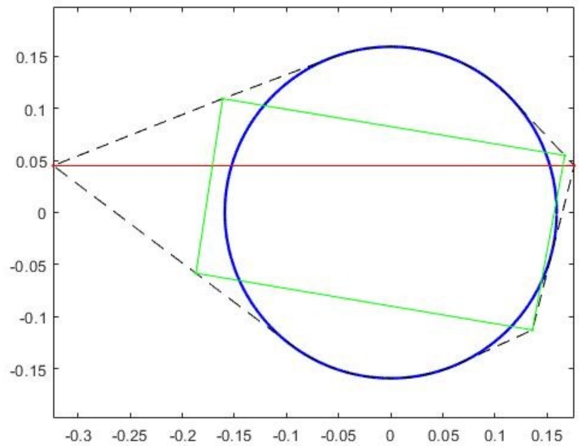

Figure 4.22: The optimal configuration of 500-gon, a line and a quadrilateral (green) with the convex hull area of 0.1003.

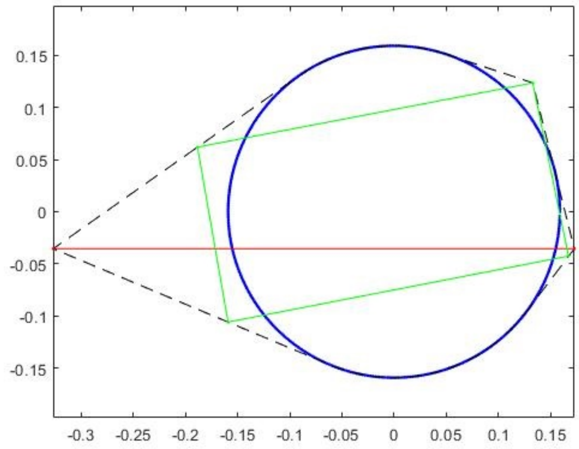

Figure 4.23: The optimal configuration of 500-gon, a line and a pentagon (green) with the convex hull area of 0.100304.

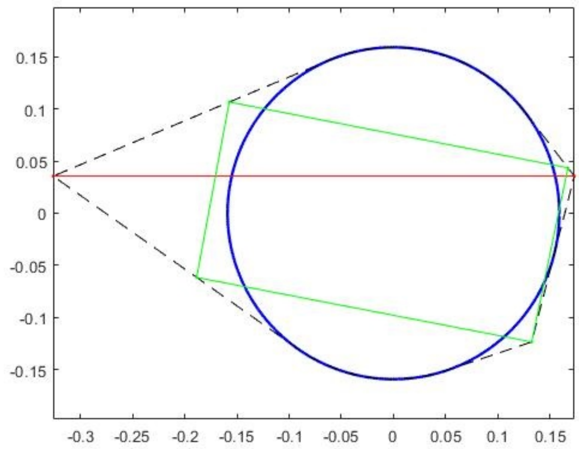

Figure 4.24: The optimal configuration of 500-gon, a line and a hexagon (green) with the convex hull area of 0.100374.

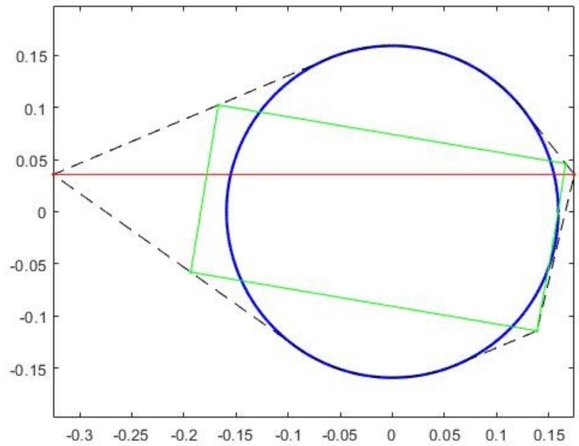

Figure 4.25: The optimal configuration of 500-gon, a line and 7-gon (green) with the convex hull area of 0.100386.

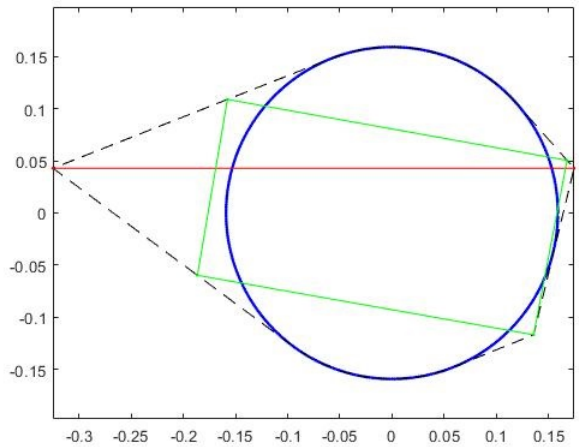

Figure 4.26: The optimal configuration of 500-gon, a line and 8-gon (green) with the convex hull area of 0.100390.

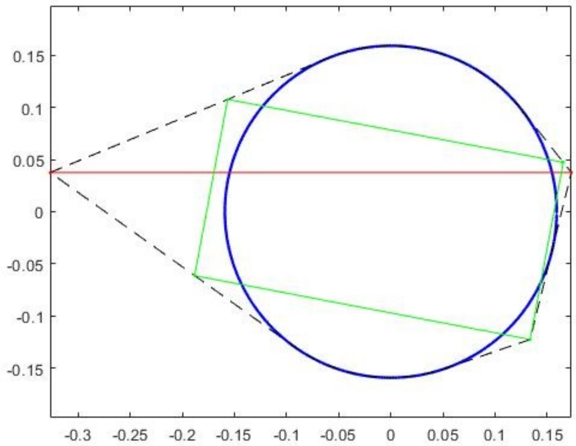

Figure 4.27: The optimal configuration of 500-gon, a line and 9-gon (green) with the convex hull area of 0.100418.

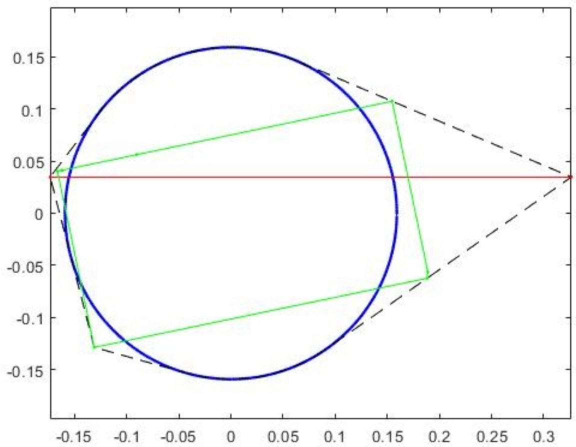

Figure 4.28: The optimal configuration of 500-gon, a line and 10-gon (green) with the convex hull area of 0.100473.

From Table 4.3, the optimal area for $n = 3$ is 0.097043 which is close to Som-am's bound [48]. Because $(n-1)$-gon is a special case of n-gon with coinciding vertices, the lower bound improves by definition, and it is best for $n = 10$ which has an area of 0.100473 and the third object is similar to a rectangle of size 0.1727×0.3273 which we obtained when $N_1 = 7, N_2 = 7, N_3 = 50$ see Figure 4.28. The resulting bound 0.100473 is also very close to the bound 0.100417 in Table 4.2. Next, we will give a rigorous proof for the new lower bound by considering the smallest area of convex hull of regular 500−gon, a line of length 0.5, and a rectangle of size 0.1727×0.3273 in Chapter 5.

Chapter 5

The lower bound is 0.1

In Chapter 4, we used a numerical method to find that the smallest area of convex hull of a circle, a line and a rectangle of size 0.1727×0.3273 is about 0.1004. In this chapter, we will combine geometric method and BSA give a rigorous proof for the following Theorem.

Theorem 5.1. *Any convex set S on the plane which can cover a circle of perimeter 1,a rectangle of size 0.1727×0.3273, and a line of length $\frac{1}{2}$ has area at least 0.1.*

Corollary 5.2. *Any convex cover for closed unit curves has area of at least 0.1.*

5.1 Geometric analysis

Let C be a circle of perimeter 1, R be a rectangle of size $u \times v$ where $u = 0.1727$ and $v = \dfrac{1}{2} - u$, and L be line of length $\dfrac{1}{2}$.

Let us fix the center of circle to be $C_0(0,0)$. Let F be a regular 500-gon inscribed in C, such that the sides of R are parallel to some longest diagonals of F. A *configuration* X is a union $F \cup R \cup L$.

Let $R_0(x_1, y_1)$ be the center of R. Thus, the vertices of R are $R_1\,(x_1 - v/2, y_1 + u/2)$, $R_2\,(x_1 + v/2, y_1 + u/2)$, $R_3\,(x_1 + v/2, y_1 - u/2)$, and $R_4\,(x_1 - v/2, y_1 - u/2)$. By the symmetry of circle, we may assume that $x_1, y_1 \geq 0$. Let $L_0(x_2, y_2)$ be the center of L and θ be the angle between X axis and L_0L_2, see Figure 5.1.

We have the vertices of L are $L_1(x_2 + \frac{1}{4}\cos(\theta + \pi), y_2 + \frac{1}{4}\sin(\theta + \pi))$ and $L_2(x_2 + \frac{1}{4}\cos(\theta), y_2 + \frac{1}{4}\sin(\theta))$. Obviously, $0 \leq \theta \leq \pi$. It is similar to Chapter 3. We define the continuous function $f : \mathbb{R}^5 \to \mathbb{R}$ which maps the vector $(x_1, y_1, x_2, y_2, \theta)$ to $\mathcal{A}(X)$. Since F is a subset of C, Theorem 5.1 would follow from the inequality

$$f(x_1, y_1, x_2, y_2, \theta) \geq 0.1 \quad \forall\, x_1, y_1, x_2, y_2, \theta.$$

Theorem 5.1 will be satisfied.

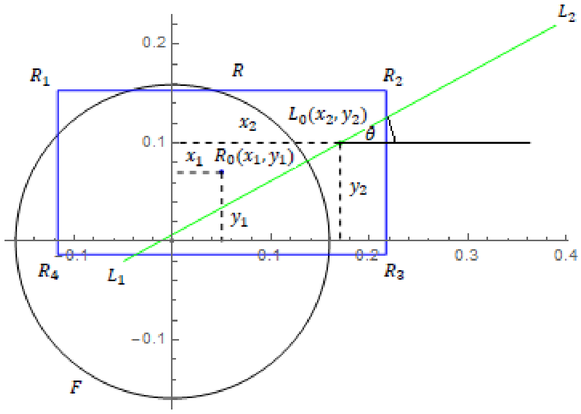

Figure 5.1: The configuration X.

The result of Fary and Redei [11] in Chapter 3, see Lemma 3.2 and Corollary 3.3, will be applied to find some lemmas which will help us to prove Theorem 5.1.

Lemma 5.3. *Let Z be a region of points $z = (x_1, y_1, x_2, y_2, \theta)$ in $\mathbb{R}^5$satisfying the inequalities*

$$0 \leq x_1 \leq 0.0741, 0 \leq y_1 \leq 0.0976, -0.148 \leq x_2 \leq 0.148, -0.148 \leq y_2 \leq 0.148, 0 \leq \theta \leq \pi.$$

If $f(z) > 0.1$ for all $z \in Z$, then in fact $f(z) > 0.1$ for all $z \in \mathbb{R}^5$.

Proof. Let $\psi(x_1, y_1)$ be the area of convex hull of F and L only. By [11] and symmetry in both coordinates, we have

$$\psi(x_1, y_1) \geq \psi(x_1, 0) \geq \psi(0.0741, 0) > 0.1,$$

whenever $x_1 \geq 0.0741$. Similarly,

$$\psi(x_1, y_1) \geq \psi(0, y_1) \geq \psi(0, 0.0976) > 0.1,$$

whenever $y_1 \geq 0.0976$.

Let $\phi(x_2, y_2)$ be the area of convex hull of F and L only. By [11], L can be moved in $\overrightarrow{L_1L_2}$ which $\phi(x_2, y_2)$ attains minimum when C_0L_0 is perpendicular to L_1L_2 at L_0.

Next, assume that $|x_2| \geq 0.148$ or $|y_2| \geq 0.148$. Then $\sqrt{x_2^2 + y_2^2} \geq |x_2| \geq 0.148$.

Let l be the line segments C_0L_0 and EK are perpendicular to l at C_0, see Figure 5.2. Then EKL_1L_2 is trapezoid with bases lengths $|EK| \geq 2r\cos(\frac{\pi}{500})$ and $|L_1L_2|$,where $r = \frac{1}{2\pi}$.The area of F is $S(F) = 500 \cdot \frac{r^2}{2} \cdot \sin(\frac{2\pi}{500})$. Thus,

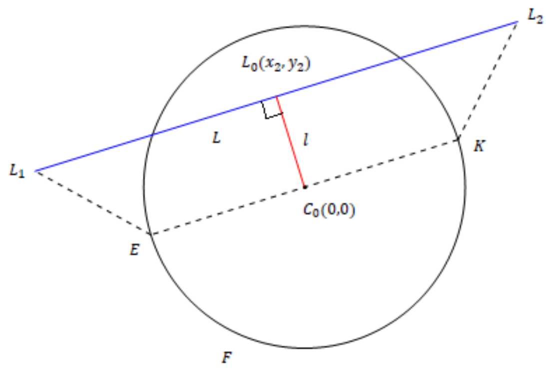

Figure 5.2: $EKL_1L_2$and l.

$$\mathcal{A}(X) > \frac{1}{2}\left(\frac{1}{2} + 2r\cos\left(\frac{\pi}{500}\right)\right)\sqrt{x_2^2 + y_2^2} + \frac{S(F)}{2} > 0.1$$

□

Lemma 5.4. *Either $f(z) > 0.1$, or $F \cup L \cup R$ is a subset of a rectangle with side lengths* 0.439×0.0636.

Proof. By Lemma 5.3, we can assume that $z = (x_1, y_1, x_2, y_2, \theta) \in Z$. Let Y_1 and Y_2 be the points of configuration $X = F \cup L \cup R$ with the lowest and highest $y-$coordinates y_1^* and y_2^* respectively. Because $0 \le y_1 \le 0.0976$, Y_1 is below R. Let h_1, h_2 be the height from Y_1, Y_2 to R, respectively. $h_2 = 0$ if Y_2 is below or on R_1R_2. Let $y_2^* - y_1^* > 0.439$. We have $\mathcal{A}(X) > u\left(\frac{1}{2} - u\right) + \frac{1}{2}\left(\frac{1}{2} - u\right)(h_1 + h_2) = u\left(\frac{1}{2} - u\right) + \frac{1}{2}\left(\frac{1}{2} - u\right)(y_2^* - y_1^* - u) > 0.1$, see Figure 5.3.

Let X_1 and X_2 be the points of configuration $X = F \cup T \cup R$ with the smallest and largest $x-$coordinates x_1^* and x_2^*, respectively. Let $z = (x, y) \in X$. If z is any point on the circle, then $|z| < 0.159$. Lemma 5.3 implies that if z is any point on the rectangle, then $|z| < 0.2378$ and if z is any point on the line, then $|z| < 0.398$. If $X_1, X_2 \in L$, then $x_2^* - x_1^* \le |x_2^*| + |x_1^*| < 0.398 + 0.2378 < 0.636$, see Figure 5.4.

□

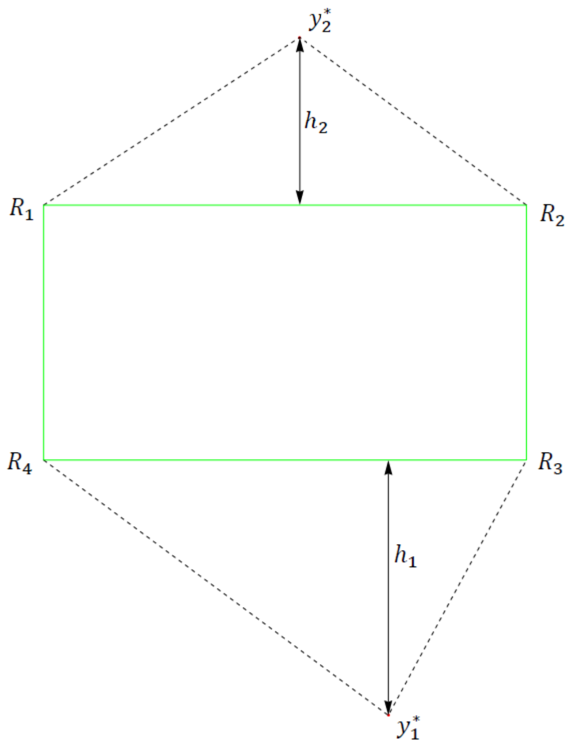

Figure 5.3: The configuration of y_1^*, y_2^* and R.

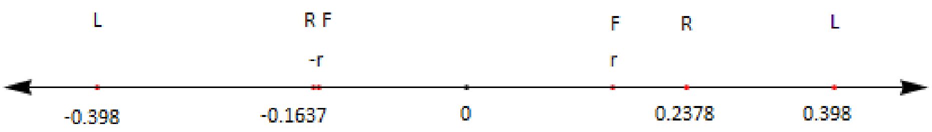

Figure 5.4: The line shows the optimum possible position of F, R, L.

Lemma 5.5. *For every* $(x_1, y_1, x_2, y_2, \theta) \in Z$ *, and any* $\epsilon_i \geq 0$, $i = 1, \dots, 5$,

$$|f(x_1 + \epsilon_1, y_1 + \epsilon_2, x_2 + \epsilon_3, y_2 + \epsilon_4, \theta + \epsilon_5) - f(x_1, y_1, x_2, y_2, \theta)| \leq \sum_{i=1}^{5} \epsilon_i C_i,$$

with constants $C_1 = 0.306$, $C_2 = 0.443$, $C_3 = 0.392$, $C_4 = 0.449$ *and* $C_5 = 0.115$.

Proof. Let $g : \mathbb{R} \to \mathbb{R}$ be a convex function on $\mathbb{R}$. We apply Lemma 3.6 in Chapter 3 to find constants $C_1, C_2, \dots, C_5$.

Let $g(x_1) = f(x_1, y_1, x_2, y_2, \theta)$, where y_1, x_2, y_2, θ are fixed and $r = \frac{1}{2\pi}$. We have

$$\lim_{x_1 \to -\infty} \frac{g(x_1)}{x_1} = \lim_{x_1 \to +\infty} \frac{g(x_1)}{x_1} \leq \frac{0.439 + u}{2} < C_1,$$

where 0.439 comes from Lemma 5.4, while u is the height of R, see Figure 5.5.

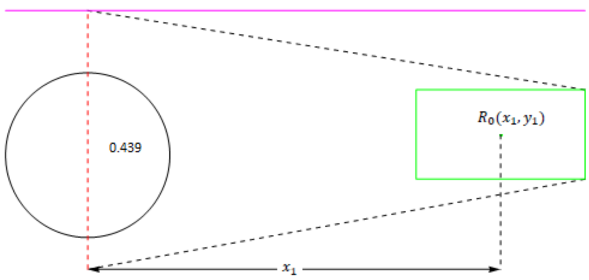

Figure 5.5: The ratio between $g(x_1)$ and $x_1$when $x_1 \to +\infty$.

Hence,

$$|f(x_1 + \epsilon_1, y_1, x_2, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_1 \epsilon_1. \tag{5.1}$$

Similarly, with $g(y_1) = f(x_1, y_1, x_2, y_2, \theta)$ for fixed x_1, x_2, y_2, θ,

$$\lim_{y_1 \to -\infty} \frac{g(y_1)}{y_1} = \lim_{y_1 \to +\infty} \frac{g(y_1)}{y_1} \leq \frac{(0.5 - u) + (x_2 + 0.25 + r)}{2} < C_2,$$

where $r = \frac{1}{2\pi}$ and $|x_2| \leq 0.148$, while u is the height of R, see Figure 5.6.

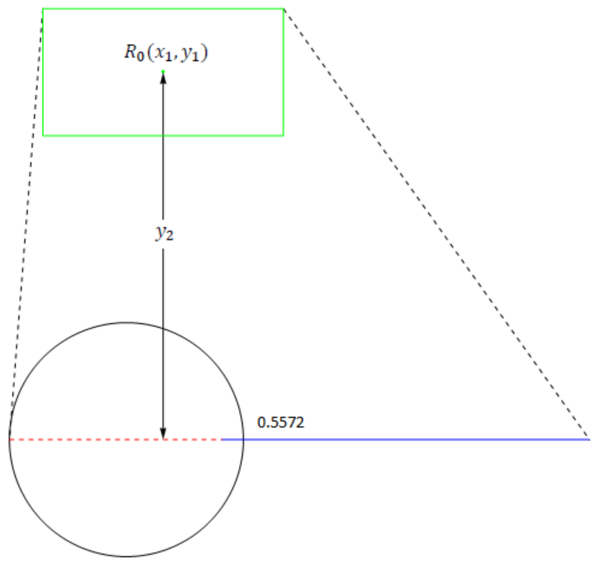

Figure 5.6: The ratio between $g(y_1)$ and y_1 when $y_1 \to +\infty$.

With $g(x_2) = f(x_1, y_1, x_2, y_2, \theta)$,

$$\lim_{x_2 \to -\infty} \frac{g(x_2)}{x_2} = \lim_{x_2 \to +\infty} \frac{g(x_2)}{x_2} \leq \frac{0.439 + (y_1 + u/2 + r)}{2} < C_3,$$

where 0.439 comes from Lemma 5.4 and $0 \leq y_1 \leq 0.0976$ see Figure 5.7.

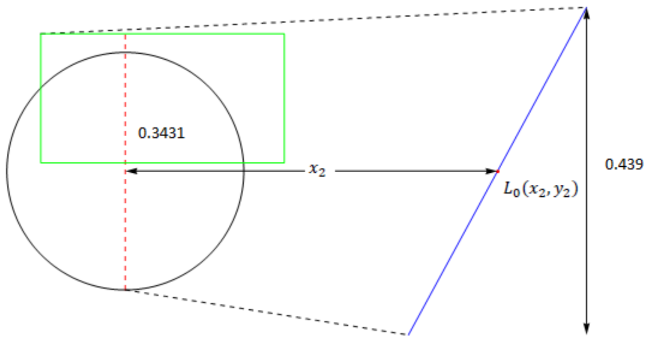

Figure 5.7: The ratio between $g(x_2)$ and x_2 when $x_2 \to +\infty$.

With $g(y_2) = f(x_1, y_1, x_2, y_2, \theta)$,

$$\lim_{y_2 \to -\infty} \frac{g(y_2)}{y_2} = \lim_{y_2 \to +\infty} \frac{g(y_2)}{y_2} \leq \frac{0.5 + (x_1 + (0.5 - u)/2 + r)}{2} < C_4,$$

where $0 \leq x_1 \leq 0.0741$ see Figure 5.8.

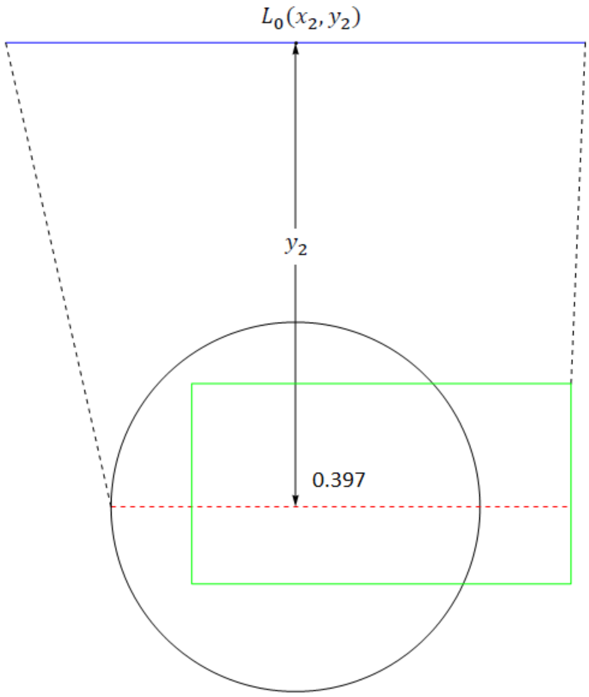

Figure 5.8: the ratio between $g(y_2)$ and y_2 when $y_2 \to +\infty$.

This implies that

$$|f(x_1, y_1 + \epsilon_2, x_2, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_2\epsilon_2, \tag{5.2}$$

$$|f(x_1, y_1, x_2 + \epsilon_3, y_2, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_3\epsilon_3, \tag{5.3}$$

$$|f(x_1, y_1, x_2, y_2 + \epsilon_4, \theta) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_4\epsilon_4. \tag{5.4}$$

Finally, we need to prove that

$$|f(x_1, y_1, x_2, y_2, \theta + \epsilon_5) - f(x_1, y_1, x_2, y_2, \theta)| \leq C_5\epsilon_5. \tag{5.5}$$

To prove the bound for C_5, we need the following claim.

Claim 1. The diameter $d(\mathcal{F} \cup \mathcal{R})$ of $\mathcal{F} \cup \mathcal{R}$ is less than 0.45976.

Indeed, let $R_0 = (0.0741, 0.0976)$. We get $R_2 = (0.23775, 0.18395)$. Let $F_1 \in F$ be a point such that $d(F_1, R_2) > d(x, R_2)$ for all $x \in F$, see Figure 5.9. By direct calculation, $|R_2R_4| < 0.37007 < |R_2F_1|$. Hence, the diameter of $F \cup R$ is $|R_2F_1|< 0.45976$.

Next, we will prove (5.5).

Let L' with endpoints L_1',L_2' be the line L rotated around L_0 by angle ϵ_5. Then $|L_1L_1'| = 2|L_0L_1|\sin(\epsilon_5/2) < 2|L_0L_1|(\epsilon_5/2) = |L_0L_1|\epsilon_5 = \frac{1}{4}\epsilon_5$.Similarly, $|L_2L_2'| < \frac{1}{4}\epsilon_5$.

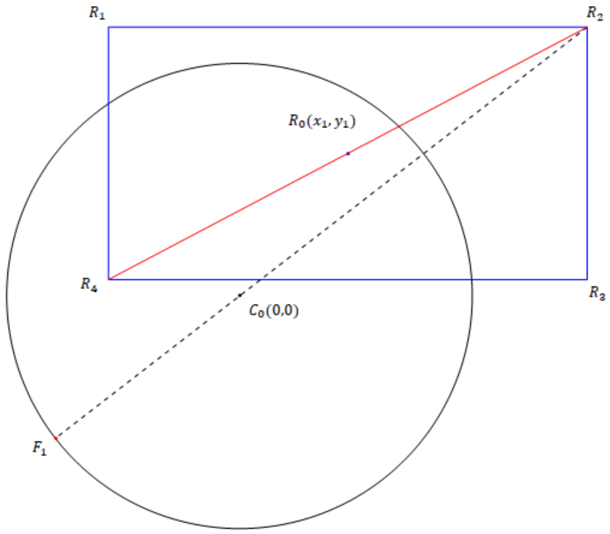

Figure 5.9: The longest distance between R_2 and F.

By selecting ϵ_5 sufficiently small, we can ensure that all vertices of polygons $\mathcal{H}(R, F, L)$ and $\mathcal{H}(R, F, L')$ coincide, except possibly the endpoints of L and L'. Then area difference $|\mathcal{A}(R, F, L') - \mathcal{A}(R, F, L)|$ is bounded by the total area of four triangles $X_1L_1X_2$, $X_1L_1^{'}X_2$, $X_3L_2X_4$, $X_3L_2^{'}X_4$, where X_i, $i = 1, 2, 3, 4$, are vertices of the polygon $\mathcal{H}(R, F, L)$ adjacent to L_1, L_2, see Figures 5.10 and 5.11.

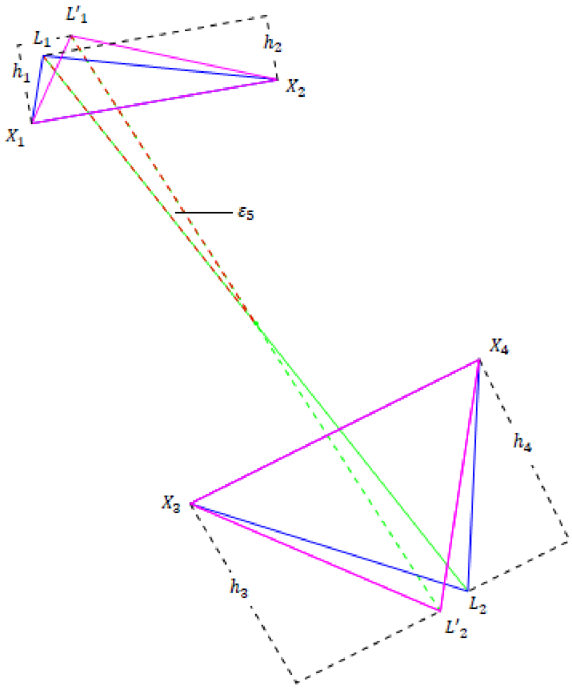

Figure 5.11: Four triangles with L rotated by angle ϵ_5.

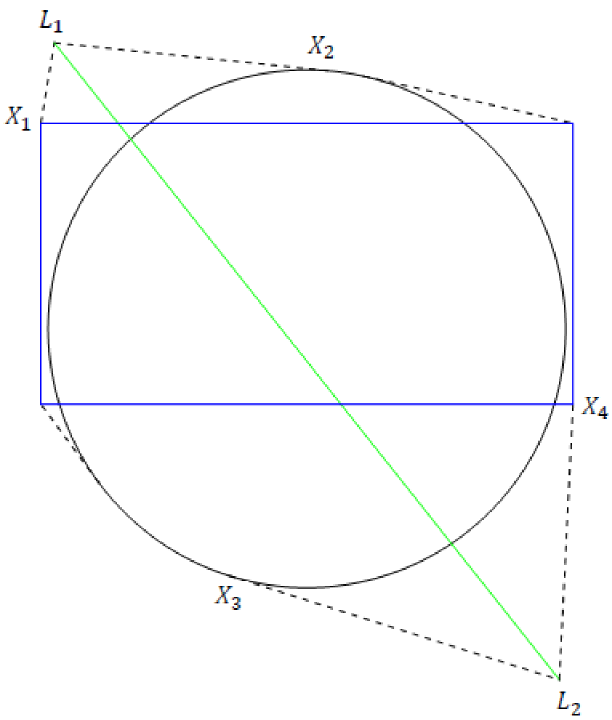

Figure 5.10: Polygon $\mathcal{H}(R, F, L)$ adjacent to L_1,L_2.

Let h_1, h_2 be the height of triangle with respect to base X_1X_2. Let h_3, h_4 be the height of triangle with respect to base X_3X_4, see Figure 5.11. By claim 1, We have

$|\mathcal{A}(R,F,L')-\mathcal{A}(R,F,L)| \leq |\frac{1}{2}h_1X_1X_2-\frac{1}{2}h_2X_1X_2|+|\frac{1}{2}h_3X_3X_4-\frac{1}{2}h_4X_3X_4|=\frac{1}{2}X_1X_2|h_1-h_2|+\frac{1}{2}X_3X_4|h_3-h_4| < \frac{1}{2}X_1X_2|L_1L_1^{'}|+\frac{1}{2}X_3X_4|L_2L_2^{'}| \leq 2\times\frac{1}{2}d(\mathcal{F}\cup\mathcal{R})\times\frac{1}{4}\epsilon_5 \leq \frac{1}{4}\times 0.45976$
$< 0.115\epsilon_5 = C_5\epsilon_5$. $\square$

On the other hand, if we apply the same method in Chapter 3 to approximate C_1, C_2, C_3, C_4 and C_5, we have $C_1 = 0.2535$, $C_2 = 0.4166$, $C_3 = 0.3482$, $C_4 = 0.4191$ and $C_5 = 0.0256$. see Figure 5.12-5.16.

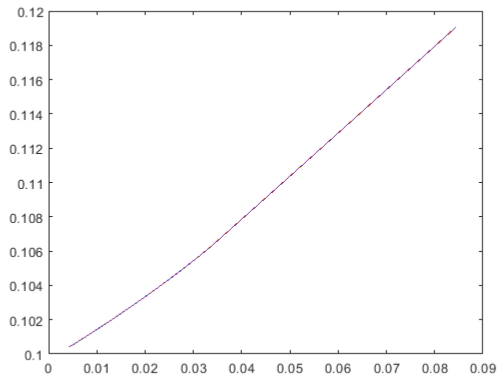

Figure 5.12: The maximum slope for C_1 is 0.2535.

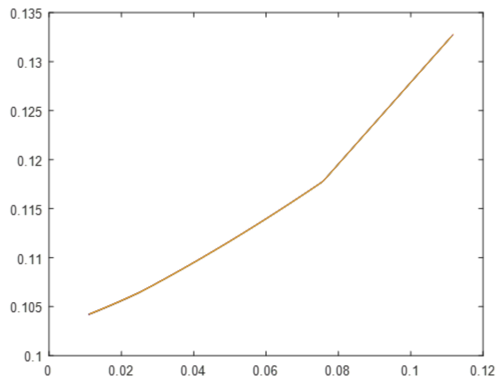

Figure 5.13: The maximum slope for C_2 is 0.4166.

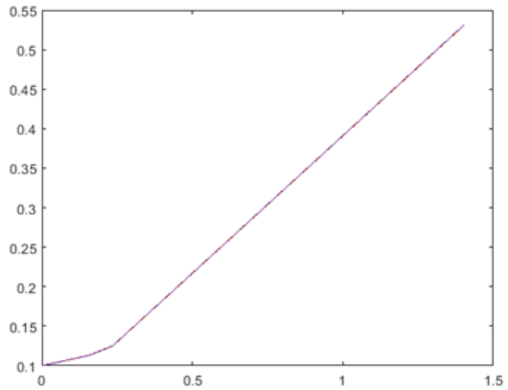

Figure 5.14: The maximum slope for C_3 is 0.3482.

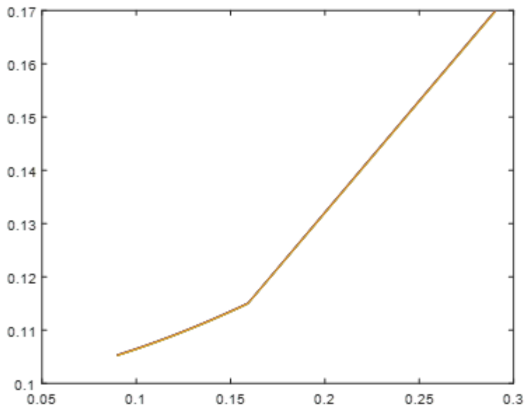

Figure 5.15: The maximum slope for C_4 is 0.4191.

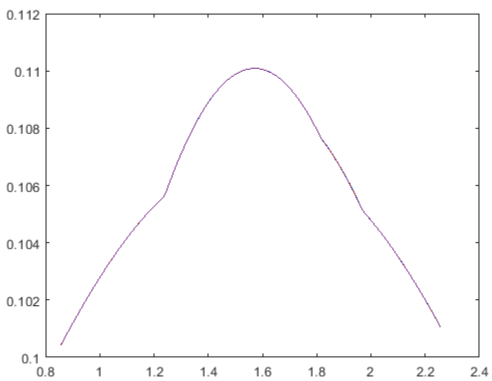

Figure 5.16: The maximum slope for C_5 is 0.0256.

we see the constants C_1, C_2, C_3, C_4, C_5 by geometric proof are close to numerical results, hence all of constants by geometric proof are reliable. Next, we will use BSA to prove

Theorem 5.1.

5.2 Computational results

Let Z be a region in Lemma 5.3. In this section, we use BSA in Chapter 2 to prove that

$$f(z) = f(x_1, y_1, x_2, y_2, \theta) > 0.1, \forall z \in Z \tag{5.6}$$

Let $z*$ be the center of a box B which has the form $B = [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3] \times [a_4, b_4] \times [a_5, b_5]$. On every step, we check the inequality

$$f(z*) - d_1C_1 - d_2C_2 - d_3C_3 - d_4C_4 - d_5C_5 \geq 0.1 \tag{5.7}$$

where $d_i = \frac{b_i - a_i}{2}$. If (6) holds, then (5) holds by Lemma 5.5.

If (6) does not hold, we will choose the largest length and split B into two boxes and then we check (6). If (6) does not hold, we subdivide the corresponding boxes again and proceed iteratively.

We start with $B = Z$, and, when the program halts, we are quaranteed that $f(z) > 0.1, \forall z \in Z$.

We run the BSA in Chapter 2 by using Matlab® R2018a. The actual Matlab code is presented in Supplementary Material.

The program halts after $n = 527, 754, 566$ iterations which show that $f(z) > 0.1, \forall z \in Z$. The program actually returned the minimal area 0.1004 for the optimal configuration with $x_1 = 0.00434$, $y_1 = 0.00648$, $x_2 = 0.00434$, $y_2 = -0.00434$, $\theta = 0.85711$ see Figure 6.2 and Appendix A.3.

5.3 Main theorem

Theorem. *(Theorem 5.1) The area of convex cover S for circle of perimeter 1, line of length 1/2, and rectangle of size 0.1727×0.3273 is at least 0.1.*

Proof. Let Z be a region in Lemma 5.3. The fact that BSA halted together with Lemma 5.5 implies that $f(z) > 0.1$ for all $z \in Z$. then, by Lemma 5.3, $f(z) > 0.1$ holds for all $z \in \mathbb{R}^5$. Thus, $\mathcal{A}(F, R, L) > 0.1$.Since $F \subset C$, $\mathcal{A}(C, R, L) \geq \mathcal{A}(F, R, L) > 0.1$. □

Corollary 5.6. *(Corallary 5.2) Any convex cover for closed unit curves has area of at least 0.1.*

Proof. Let S be a convex cover for closed unit curves. Then S can accommodate C, R, and L, hence $\mathcal{H}(C, R, L) \subset S$. Thus the area of S is at least $\mathcal{A}(F, R, L) > 0.1$ by Theorem 5.1. □

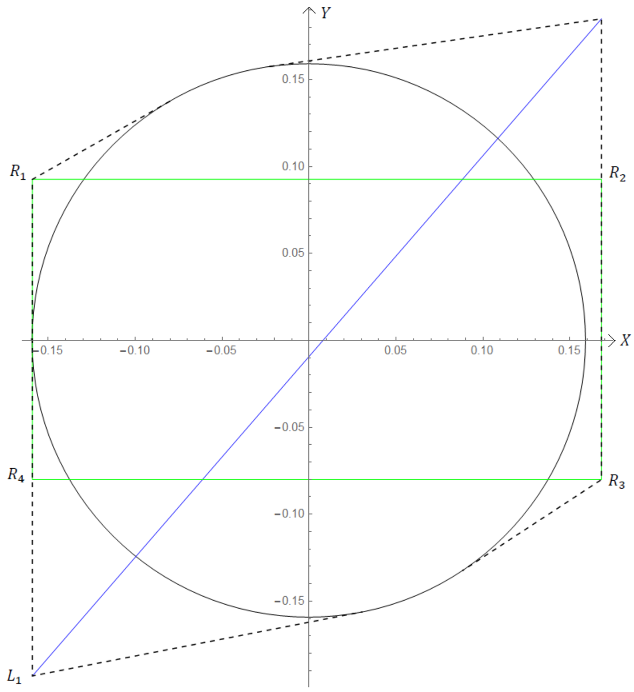

Figure 5.17: The convex hull of the configuration of the minimum area with 0.10044 acquired from the BSA.

Next, we will show that the area of convex hull of C, R, L is at most 0.1005, see Matlab code in Supplementary Material.

Theorem 5.7. *For any rectangle R' with perimeter* 1*, there is a convex cover of R', C, and L with area at most* 0.1005*.*

Proof. Let l, w be the length and width of rectangle R' such that $l+w = \frac{1}{2}$ and $w \in [0, 0.25]$. Let F' be the regular 500-gon inscribed into the circle with $r' = \frac{sec(\frac{\pi}{500})}{2\pi}$. Then $C \subset F'$, and $\mathcal{H}(X) = \mathcal{H}(R, C, T) \subset \mathcal{H}(R, F', T)$. Thus, $\mathcal{A}(X) \leq \mathcal{A}(R, F', T)$.

Let $f(w)$ denotes the minimal area of convex cover R, F', T.

Claim For any $\epsilon > 0$, $|f(w+\epsilon) - f(w)| \leq 0.345\epsilon$. It suffices to prove the claim only for small ϵ. We will prove that $f(w) - f(w+\epsilon) \leq 0.345\epsilon$, the proof for inequality $f(w+\epsilon) - f(w) \leq 0.345\epsilon$ is similar. Let R'' be the rectangle with width $w+\epsilon$ and perimeter 1. Consider optimal configuration of R'', F', L, so that $f(\omega+\epsilon) = \mathcal{A}(R'', F', L)$. Let us put R' parallel to R'' as shown on Figure 5.18. This configuration is not necessary optimal, and, because f denotes the area of the *optimal* configuration, $f(w) \leq \mathcal{A}(R', F', L)$. Hence, $f(w) - f(w+\epsilon) \leq \mathcal{A}(R', F', L) - \mathcal{A}(R'', F', L)$.

Convex hulls $\mathcal{H}(R', F', L)$ and $\mathcal{H}(R'', F', L)$ are polygons, and, by selecting ϵ sufficiently small, we can assume that all vertices of these polygons, which are not vertices of R' and R'', coincides. Then $\mathcal{A}(R', F', L) - \mathcal{A}(R'', F', L)$ is bounded by the total area of triangles

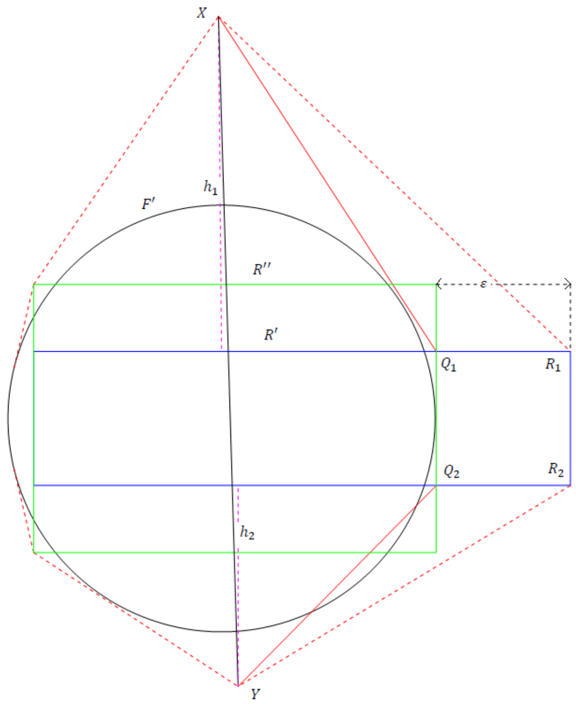

Figure 5.18: The configuration of R and R'.

XQ_1R_2, YQ_2R_3, and rectangle $Q_1R_2R_3Q_2$, which is

$$\frac{1}{2}h_1\epsilon + \frac{1}{2}h_2\epsilon + Q_1Q_2\epsilon = \frac{\epsilon}{2}(h_1 + h_2 + 2Q_1Q_2)$$

We have $Q_1Q_2 = w \le 0.25$, and, by Lemma 5.4, $h_1 + h_2 + Q_1Q_2 \le 0.439$. Hence,

$$f(w) - f(w+\epsilon) \le \mathcal{A}(R', F', L) - \mathcal{A}(R'', F', L) \le \frac{\epsilon}{2}(0.439 + 0.25) = 0.345\epsilon,$$

which proves the claim.

To verify inequality $f(w) < 0.1005$ at some *specific* point.

We will select set W in such a way that intervals $[w_i - d_i, w_i + d_i]$, $i = 1, 2, \ldots, N$ cover the whole interval $[0, 0.25]$. In other words, $w_1 - d_1 < 0$, $w_N + d_N > 0.25$, and

$$w_i + d_i < w_{i+1} - d_{i+1}, \quad i = 1, 2, \ldots, N-1.$$

Set W with $N = 100$ points with this property is presented in the Appendix A.4. For example,$w_1 = 0.00010$, $w_2 = 0.01537$, $w_3 = 0.02932$, and so on, $w_{101} = 0.24524$.

□

To conclude, we used the geometric method and numerical BSA to show that the optimal area of convex cover for a circle of perimeter 1, line of length 1/2, and rectangle of perimeter 1 is between 0.1 and 0.1005.

Chapter 6

Conclusions and Conjecture

6.1 Conclusions

Moser's worm problem is a famous unsolved problem in geometry which asks for the region of smallest area in the plane which can be rotated and translated to cover every unit arc. If the region is convex, there exists a solution which follows from Blaschke selection theorem [33]. Thus, we will restrict the region to be convex. This problem still open, but we know the bound of the solution which is between 0.2322 [30] and 0.2618[42]. In our problem, we restrict the arc to be closed unit arcs. The previous bound of this problem was between 0.096694 [13] and 0.11023 [56]. We focused on improving the lower bound for this problem. For lower bound problem, there is only Fary and Redei's theorem [11] to guarantee the smallest area of convex hull of two centrally symmetric convex objects. Furedi and Wetzel [13] applied this theorem to find the current lower bound. In 2005, Brass and Sharifi [5] used numerical method to improve a lower bound for universal cover's problem by considering the smallest convex hull of three objects. Later, Khandhawit and Srisawas [31] used Brass grid method to improve the lower bound for Moser's worm problem and then they use min-max method to prove the current lower bound. In 2010, Som-am [48] used Brass grid method to improve the current lower bound for closed arcs, but it is not published. We can see that there is only Brass grid method to improve a lower bound by considering three or more objects. In Chapter 2, we construct BSA which is the method to check inequality (2.4) for Lipschitz continuous function. We start from the first box which is the mid point of box. If (2.4) holds, then we are done. If not, we will choose the largest length of parameters and then divided in to two boxes and check the inequality (2.4) again and so on. Furthermore, we create heap sort algorithm which is a method to sort the array of size n in term of a binary tree to prove a lower bound. We start with box B and apply the BSA to split it into 2 boxes. Next, we compare the two boxes generated from the first array and check which box has the smaller numerical value. The smaller one will become the first element of the array and the other one will be the last element. After that, this array will be sorted by heap method and generate the n two new boxes from first array, and so on. We improved the lower bound to 0.0975 by combining a geometric method, which proves the Lipschitz bound for the corresponding function, and numerical BSA. Our numerical results actually imply lower bound 0.097627 corresponding to the optimal configuration with parameter $x_1 = 0.0251$, $y_1 = 0.00258$, $x_2 = 0.0653$, $y_2 = 0.00542$, and $\theta = 0.07989$, see Figure 6.1. Moreover, we used MHS to prove the lower bound, but it did not finish because the computer run out of memory. We can only prove the bound 0.097412. We see that BSA is more efficient than MHS. Thus, we have used BSA to prove the bound. Although, the bound 0.0975 is a weaker bound, we have proved that the smallest area of convex hull of a circle, a rectangle and an equilateral triangle is at most 0.09763.

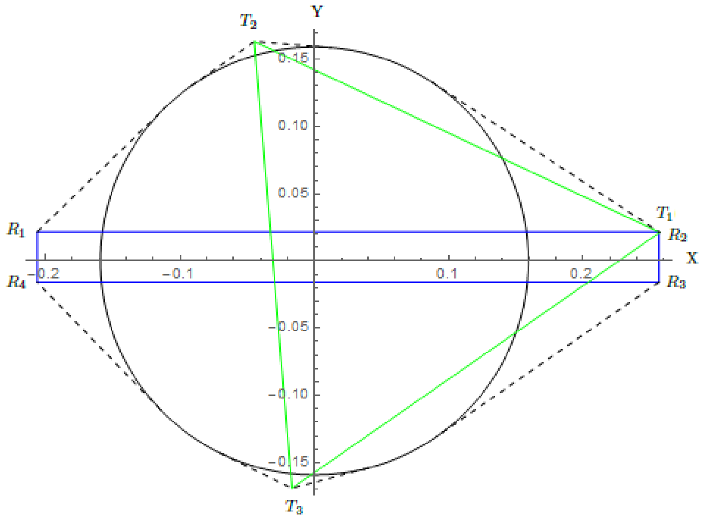

Figure 6.1: The convex hull of the configuration with the minimum area of 0.097627 acquired from the BSA.

Next, we wanted to extend Fary and Redei's theorem for three or more centrally symmetric objects, but it is not true, see counterexample in Section 4.1.3. However, its area is 0.100465 which is better than the bound, 0.0975 found in Chapter 3. We used numerical method to find the optimal area of convex hull of two and three arbitrary polygons. For each specific set of objects, we find minimum area convex hull with respect to all translations and rotations, and then we modify the shape of the objects to make this minimal area as large as possible. We start from polygons with small numbers of vertices and then increase the number of vertices. For example, we consider the case of two triangles (3+3 vertices), a triangle and quadrilateral (3+4 vertices), two quadrilaterals (4+4 vertices) and so on. The best results for two objects is regular 50-gon and 11-gon, whose minimal convex hull area is 0.0966. For 3 objects, we try to find the optimal shape of 2 triangles and line (3+3+2 vertices), 3 triangles (3+3+3 vertices), 2 triangles and a quadrilateral (3+3+4 vertices) and so on. The best three objects we found are circle, line and rectan-

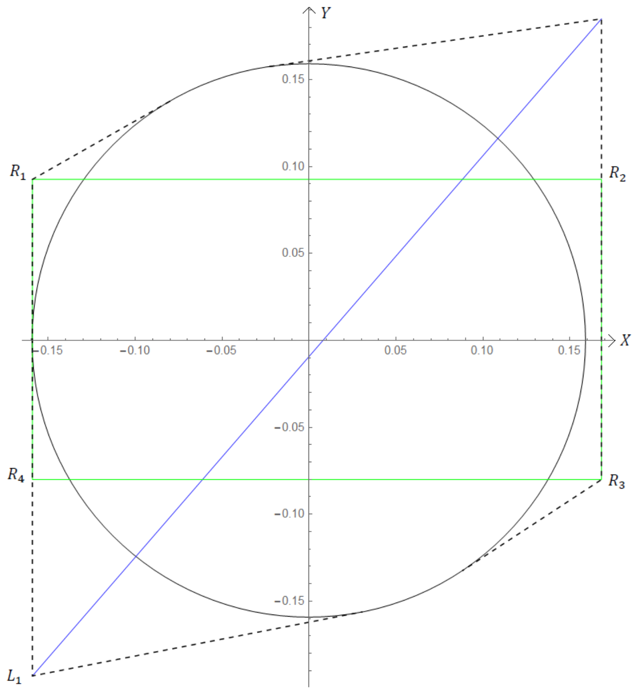

Figure 6.2: The convex hull of the configuration with the minimum area of 0.10044 acquired from the BSA.

gle with sides 0.1727×0.3273 (and perimeter one). The minimal area of convex hull for this object is, numerically, about 0.1004. We combine the geometric method and BSA to prove that this area is greater than 0.1. **Hence, we have proved a new lower bound** 0.1. By BSA, we can get the smallest area is 0.10044 for the optimal configuration with $x_1 = 0.00434$, $y_1 = 0.00648$, $x_2 = 0.00434$, $y_2 = -0.00434$, $\theta = 0.85711$ see Figure 6.2.

However, we cannot prove this bound. To improve beyond this, different configurations of objects should be considered. The numerical results in Chapter 4 suggest that no configuration of three objects can give a bound much better than this. Because considering four and more objects significantly increases the number of parameters hence is computationally difficult. It looks like the bound 0.1 (or slightly better) may be the limit using current techniques and new ideas are required to improve it significantly. Next, we will state a conjecture for upper bound of this problem.

6.2 Conjecture

In this section, we will give a conjecture for the upper bound. First, we will show that the configuration in Figure 6.2 is not a cover for closed unit arcs by finding some arcs which cannot be fitted in it. Next, we use numerical method to find the region which can cover all closed unit arcs. We cannot prove it, but we will show the result by numerical method.

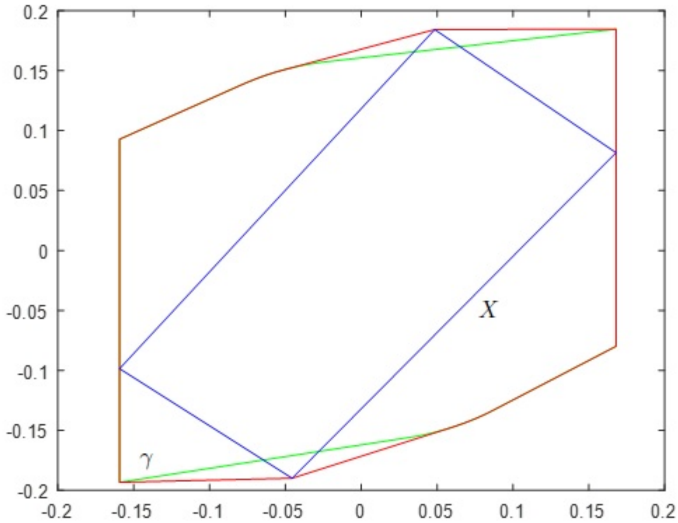

Figure 6.3: The optimal configuration of γ (green) and X (blue).

Let $F(X, \gamma)$ be an area of convex hull for polygon X and a fixed region γ. Let $\mathcal{T}$ be the set of all orientation-preserving motions T which is a composite of translations and rotations of the plane. Let X be any polygon with unit perimeter. Thus, we will find the minimum area of convex hull for X and γ by considering the optimization problem:

$$\min_{T \in \mathcal{T}} F(T(X), \gamma) \tag{6.1}$$

Next, we will find polygon X which makes (6.1) as large as possible. Let N be a number of vertices in polygon X. We assume that N is fixed but X can vary. Let $\mathcal{X}(N)$ be the set of all convex polygons with N vertices and unit perimeter. We consider maximin problem

$$\max_{X \in \mathcal{X}(N)} \min_{T \in \mathcal{T}} F(T(X), \gamma) \tag{6.2}$$

Let γ be the convex hull of the configuration of the minimum area with 0.10044 acquired from the BSA see Figure 6.2. Let $N = 4$. We use Algorithm 9 and 10 to find that the maximin value is about 0.104129 with quadrilateral X with sides $0.1459, 0.3509, 0.1576,$ and 0.3455. Since X does not fit in γ, γ is not a cover for closed unit arcs, see Figure 6.3.

Algorithm 9 **Algorithm to find maximin of** $F(X,\gamma)$

Input:N**-** the number of vertices of object X.
Output: $fval$ - the minimum area of convex hull of X and cover γ, x - the points which get $fval$ (final point).
Procedure:
Function `[x,fval]=CLRobject(N)`
1: To set the initial points of X which is a regular N-gon .
2: To use the `patternsearch` function to search maximin of $F(X,\gamma)$.
end Procedure

Algorithm 10 **Algorithm to find minimum of** $F(X,\gamma)$

Input:v **-** the point which are from CLRobject function.
Output: ar - the minimum area of convex hull of X and cover γ
Procedure:
Function `ar=CLRperN3(v)`
1: To set the coordinate of cover γ.
2: To find perimeter of X : a.
3: To scale X to perimeter one by a , say X_1^*.
4: To calculate the minimum area of convex hull of X_1^* and cover γ of the configuration described by parameters x_1, y_1, θ_1 by using function MultiStart with initial condition $(0,0,0)$ and $n = 80$ points.
end Procedure

Conjecture 6.1. *Let $\mathcal{X}$ be the optimal configuration of convex hull of a circle of perimeter* 1*, a line segment of length $\frac{1}{2}$, an equilateral triangle of size $\frac{1}{3}$, a square of size $\frac{1}{4}$, a rectangle of perimeter* 1 *and a hexagon of perimeter* 1*. Then $\mathcal{X}$ is a cover for all closed unit arcs and its area is about* 0.1046 ± 0.0001.

Let C be a circle of perimeter 1, L is a line segment of length $\frac{1}{2}$, T is an equilateral triangle of size $\frac{1}{3}$, S is a square of size $\frac{1}{4}$, R is a rectangle of sides 0.1727×0.3273 and H is a hexagon with sides $0.3467, 0.0829, 0.0730, 0.3420, 0.0316$, and 0.1238. Let $\mathcal{X}$ be the smallest area of convex hull of C, L, T, S, R, H. We fix the center of a circle at $(0,0)$. Let (x_i, y_i) and θ_i, $i = 1,2,3,4,5$ be the center and angle of rotation of L, T, S, R, H, respectively. Based on numerical method, we conjecture that X is a cover for closed unit arcs when $x_1 = 0.0687, y_1 = -0.044, x_2 = 0.0256, y_2 = 0.0001, x_3 = 0.0099, y_3 = 0.0135, x_4 = -0.0165, y_4 = 0.0039, x_5 = -0.1373, y_5 = -0.034, \theta_1 = -0.2065, \theta_2 = 0.3286, \theta_3 = 290.4667, \theta_4 = 2.0814, \theta_5 = 0.0172$ and its area is 0.1046 ± 0.0001, see Figure 6.4.

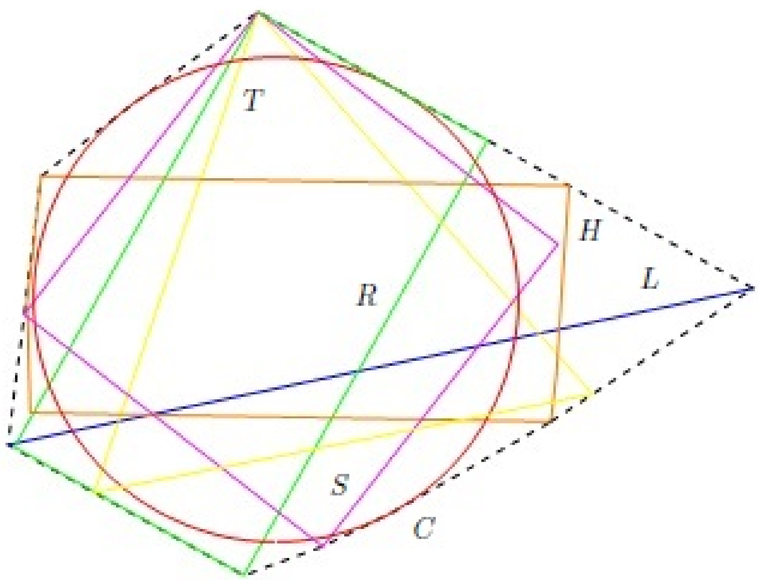

Figure 6.4: The conjecture' s cover for closed unit arcs and its area is about 0.104597

Let γ be the optimal configuration of convex hull of a circle of perimeter 1, a line segment of length $\frac{1}{2}$, an equilateral triangle of size $\frac{1}{3}$, a square of size $\frac{1}{4}$, rectangle of size 0.1727×0.3273 and a hexagon with side $0.3467, 0.0829, 0.0730, 0.3420, 0.0316$, and 0.1238. We will find polygon X which makes (6.1) as large as possible by increasing $N = 3, 4, \ldots, 10$ and 20. We find that the area of convex hull for γ and X does not change. Thus, γ should be a cover for closed unit arcs see Figure 6.5 - 6.13 and Matlab code is provided in Supplementary Material.

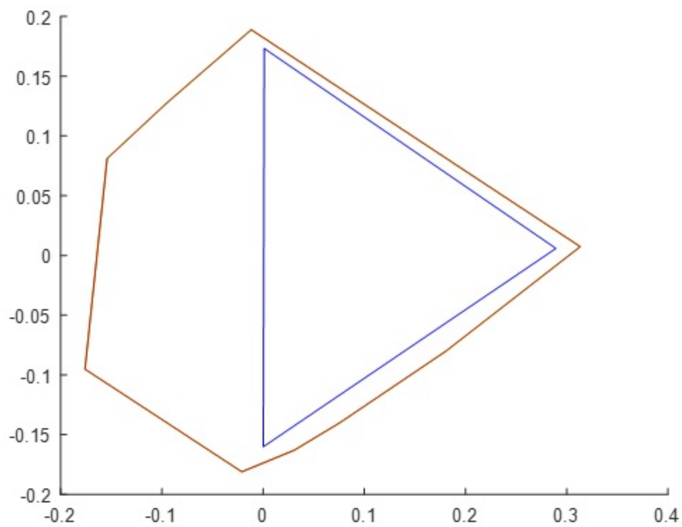

Figure 6.5: The configuration of cover γ and triangle X

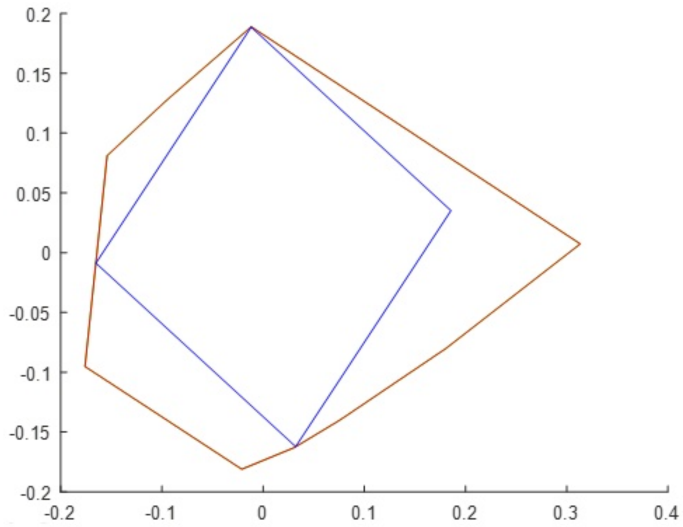

Figure 6.6: The configuration of cover γ and quadrilateral X

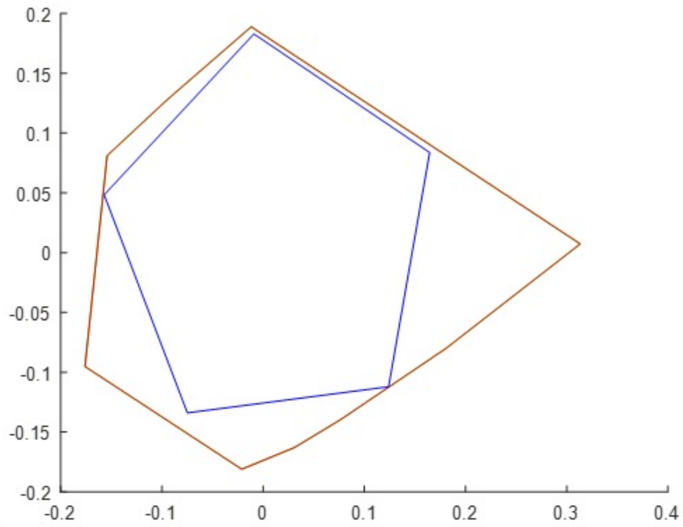

Figure 6.7: The configuration of cover γ and pentagon X

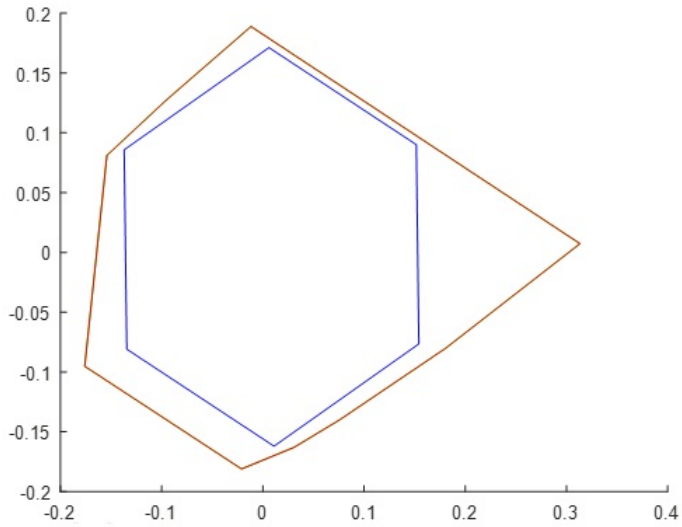

Figure 6.8: The configuration of cover γ and hexagon X

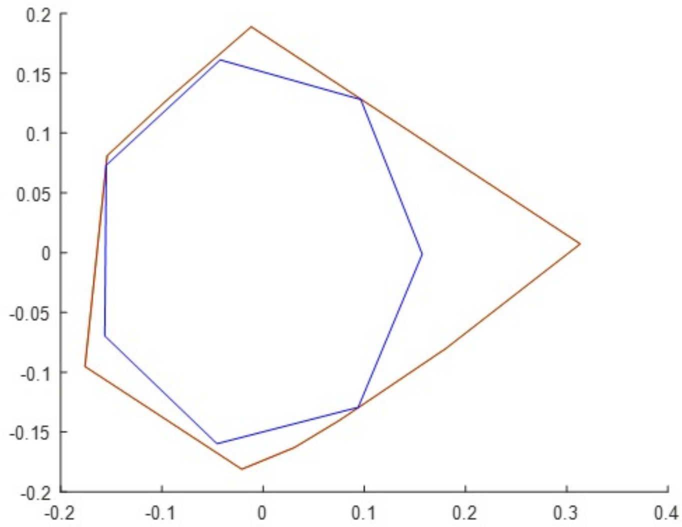

Figure 6.9: The configuration of cover γ and 7-gon X

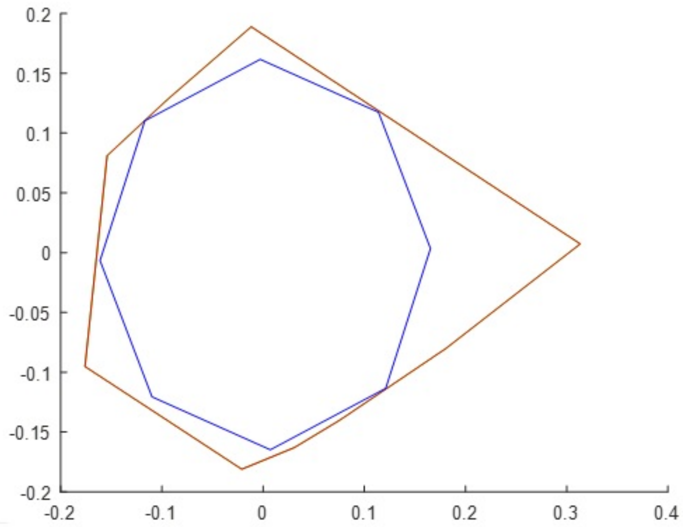

Figure 6.10: The configuration of cover γ and 8-gon X

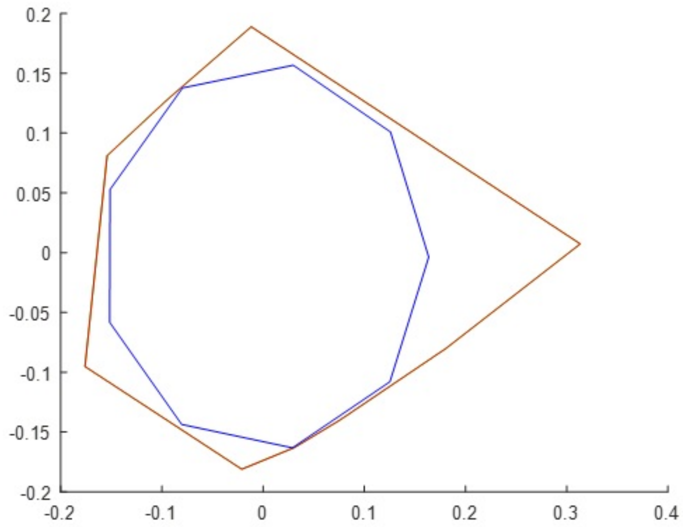

Figure 6.11: The configuration of cover γ and 9-gon X

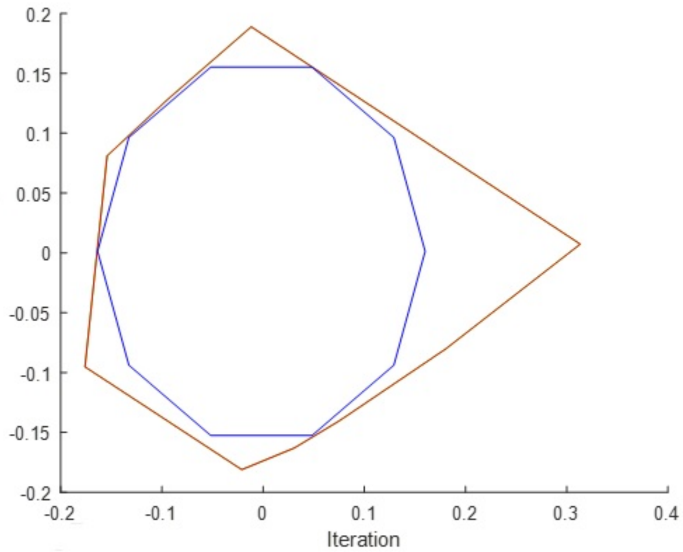

Figure 6.12: The configuration of cover γ and 10-gon X

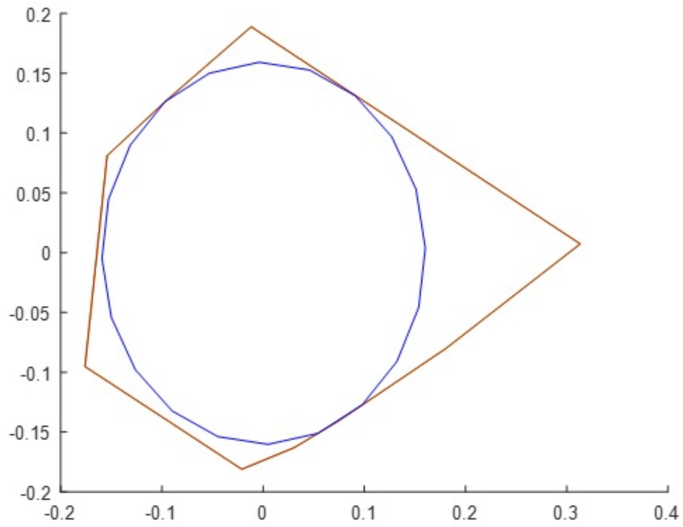

Figure 6.13: The configuration of cover γ and 20-gon X

Appendix

A.1 The BSA results in Chapter 3

The BSA displayed a message every $1,000,000$ steps. Figure presents the output of these messages for (approximately) every $1,000,000,000$ steps. Here, the first column represents progress, in terms of the percentage of the area of the initial box for which the inequality (3.4) is verified. The second column is the iteration number. Figure presents the graphical illustration how progress depends on the number of iterations.

Persentage of r	n
7.0083%	1000000000
7.93%	2000000000
7.9671%	3000000000
8.3073%	4000000000
97.2442%	5000000000
97.641%	6000000000
98.9946%	7000000000

Table 6.1: The table of percentage of r and n

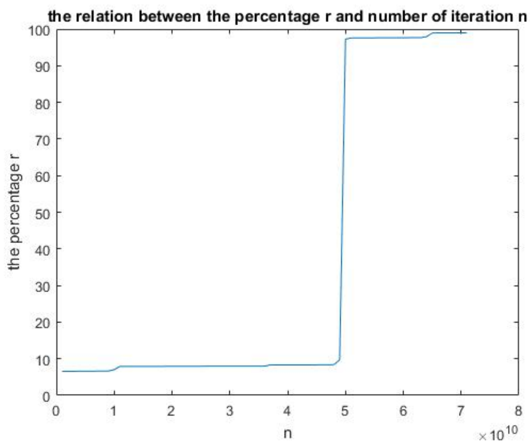

Figure 6.14: The graph of percentage of r and n

A.2 Results for Theorem 3.4 in Chapter 3

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
1	0.00020	0.09709	1.70E-03	-0.00150	0.00190
2	0.00335	0.09714	1.53E-03	0.00183	0.00488
3	0.00618	0.09721	1.33E-03	0.00485	0.00751
4	0.00864	0.09725	1.19E-03	0.00745	0.00983
5	0.01083	0.09730	1.05E-03	0.00978	0.01189
6	0.01278	0.09734	9.25E-04	0.01185	0.01370
7	0.01449	0.09737	8.18E-04	0.01367	0.01531
8	0.01600	0.09739	7.42E-04	0.01526	0.01675
9	0.01738	0.09742	6.60E-04	0.01672	0.01804
10	0.01860	0.09744	5.93E-04	0.01800	0.01919
11	0.01969	0.09746	5.31E-04	0.01916	0.02022
12	0.02067	0.09748	4.80E-04	0.02019	0.02115
13	0.02156	0.09749	4.31E-04	0.02113	0.02199
14	0.02236	0.09751	3.89E-04	0.02197	0.02275
15	0.02308	0.09752	3.52E-04	0.02273	0.02343
16	0.02373	0.09753	3.20E-04	0.02341	0.02405
17	0.02432	0.09754	2.92E-04	0.02403	0.02462
18	0.02486	0.09754	2.68E-04	0.02460	0.02513
19	0.02536	0.09755	2.46E-04	0.02511	0.02560
20	0.02581	0.09756	2.26E-04	0.02559	0.02604
21	0.02623	0.09756	2.10E-04	0.02602	0.02644
22	0.02662	0.09757	1.94E-04	0.02643	0.02682
23	0.02698	0.09757	1.80E-04	0.02680	0.02716
24	0.02731	0.09758	1.68E-04	0.02715	0.02748
25	0.02762	0.09758	1.57E-04	0.02747	0.02778
26	0.02792	0.09758	1.47E-04	0.02777	0.02806
27	0.02819	0.09759	1.37E-04	0.02805	0.02832
28	0.02844	0.09759	1.30E-04	0.02831	0.02857
29	0.02868	0.09759	1.23E-04	0.02856	0.02880
30	0.02891	0.09759	1.16E-04	0.02879	0.02903
31	0.02912	0.09760	1.10E-04	0.02901	0.02923
32	0.02933	0.09760	1.04E-04	0.02922	0.02943
33	0.02952	0.09760	9.83E-05	0.02942	0.02962
34	0.02970	0.09760	9.35E-05	0.02961	0.02980
35	0.02988	0.09760	9.02E-05	0.02978	0.02997
36	0.03004	0.09760	8.59E-05	0.02996	0.03013
37	0.03020	0.09760	8.20E-05	0.03012	0.03028
38	0.03035	0.09761	7.80E-05	0.03027	0.03043
39	0.03050	0.09761	7.43E-05	0.03042	0.03057
40	0.03063	0.09761	7.07E-05	0.03056	0.03070
41	0.03077	0.09761	6.74E-05	0.03070	0.03083
42	0.03089	0.09761	6.68E-05	0.03082	0.03096
43	0.03101	0.09761	6.42E-05	0.03095	0.03108
44	0.03113	0.09761	6.19E-05	0.03107	0.03119
45	0.03125	0.09761	5.92E-05	0.03119	0.03131
46	0.03136	0.09761	5.68E-05	0.03130	0.03141
47	0.03146	0.09761	5.44E-05	0.03141	0.03152
48	0.03156	0.09761	5.21E-05	0.03151	0.03161

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
49	0.03166	0.09761	5.00E-05	0.03161	0.03171
50	0.03175	0.09761	4.79E-05	0.03170	0.03180
51	0.03184	0.09762	4.67E-05	0.03179	0.03189
52	0.03193	0.09762	4.50E-05	0.03188	0.03197
53	0.03201	0.09762	4.33E-05	0.03197	0.03205
54	0.03209	0.09762	4.17E-05	0.03205	0.03213
55	0.03217	0.09762	4.02E-05	0.03213	0.03221
56	0.03224	0.09762	3.87E-05	0.03220	0.03228
57	0.03231	0.09762	3.77E-05	0.03227	0.03235
58	0.03238	0.09762	3.64E-05	0.03235	0.03242
59	0.03245	0.09762	3.51E-05	0.03241	0.03248
60	0.03251	0.09762	3.39E-05	0.03248	0.03255
61	0.03258	0.09762	3.27E-05	0.03254	0.03261
62	0.03264	0.09762	3.17E-05	0.03261	0.03267
63	0.03270	0.09762	3.06E-05	0.03267	0.03273
64	0.03275	0.09762	2.95E-05	0.03272	0.03278
65	0.03281	0.09762	2.85E-05	0.03278	0.03284
66	0.03286	0.09762	2.95E-05	0.03283	0.03289
67	0.03291	0.09762	2.87E-05	0.03289	0.03294
68	0.03297	0.09762	2.79E-05	0.03294	0.03300
69	0.03302	0.09762	2.71E-05	0.03299	0.03305
70	0.03307	0.09762	2.63E-05	0.03304	0.03310
71	0.03312	0.09762	2.55E-05	0.03309	0.03314
72	0.03317	0.09762	2.48E-05	0.03314	0.03319
73	0.03321	0.09762	2.41E-05	0.03319	0.03324
74	0.03326	0.09762	2.34E-05	0.03323	0.03328
75	0.03330	0.09762	2.28E-05	0.03328	0.03332
76	0.03334	0.09762	2.22E-05	0.03332	0.03336
77	0.03338	0.09762	2.15E-05	0.03336	0.03340
78	0.03342	0.09762	2.09E-05	0.03340	0.03344
79	0.03346	0.09762	2.04E-05	0.03344	0.03348
80	0.03350	0.09762	1.98E-05	0.03348	0.03352
81	0.03354	0.09762	2.04E-05	0.03351	0.03356
82	0.03357	0.09762	1.99E-05	0.03355	0.03359
83	0.03361	0.09762	1.95E-05	0.03359	0.03363
84	0.03365	0.09762	1.90E-05	0.03363	0.03367
85	0.03368	0.09762	1.86E-05	0.03366	0.03370
86	0.03372	0.09762	1.82E-05	0.03370	0.03373
87	0.03375	0.09762	1.77E-05	0.03373	0.03377
88	0.03378	0.09762	1.73E-05	0.03376	0.03380
89	0.03381	0.09762	1.69E-05	0.03380	0.03383
90	0.03385	0.09762	1.65E-05	0.03383	0.03386
91	0.03388	0.09762	1.61E-05	0.03386	0.03389
92	0.03391	0.09763	1.58E-05	0.03389	0.03392
93	0.03393	0.09763	1.54E-05	0.03392	0.03395
94	0.03396	0.09763	1.50E-05	0.03395	0.03398
95	0.03399	0.09763	1.47E-05	0.03398	0.03401
96	0.03402	0.09763	1.44E-05	0.03400	0.03403

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
97	0.03405	0.09763	1.41E-05	0.03403	0.03406
98	0.03407	0.09763	1.37E-05	0.03406	0.03408
99	0.03410	0.09763	1.34E-05	0.03408	0.03411
100	0.03412	0.09763	1.31E-05	0.03411	0.03413
101	0.03415	0.09763	1.28E-05	0.03413	0.03416
102	0.03417	0.09763	1.26E-05	0.03416	0.03418
103	0.03419	0.09763	1.23E-05	0.03418	0.03420
104	0.03422	0.09763	1.20E-05	0.03420	0.03423
105	0.03424	0.09763	1.18E-05	0.03423	0.03425
106	0.03426	0.09763	1.15E-05	0.03425	0.03427
107	0.03428	0.09763	1.13E-05	0.03427	0.03429
108	0.03430	0.09763	1.10E-05	0.03429	0.03431
109	0.03432	0.09763	1.08E-05	0.03431	0.03433
110	0.03434	0.09763	1.05E-05	0.03433	0.03435
111	0.03436	0.09763	1.03E-05	0.03435	0.03437
112	0.03438	0.09763	1.01E-05	0.03437	0.03439
113	0.03440	0.09763	9.87E-06	0.03439	0.03441
114	0.03442	0.09763	9.65E-06	0.03441	0.03443
115	0.03444	0.09763	9.45E-06	0.03443	0.03444
116	0.03445	0.09763	9.25E-06	0.03444	0.03446
117	0.03447	0.09763	9.05E-06	0.03446	0.03448
118	0.03449	0.09763	8.86E-06	0.03448	0.03450
119	0.03450	0.09763	8.68E-06	0.03449	0.03451
120	0.03452	0.09763	8.50E-06	0.03451	0.03453
121	0.03453	0.09763	8.34E-06	0.03453	0.03454
122	0.03455	0.09763	8.16E-06	0.03454	0.03456
123	0.03457	0.09763	8.00E-06	0.03456	0.03457
124	0.03458	0.09763	7.85E-06	0.03457	0.03459
125	0.03459	0.09763	7.69E-06	0.03459	0.03460
126	0.03461	0.09763	7.53E-06	0.03460	0.03462
127	0.03462	0.09763	7.37E-06	0.03462	0.03463
128	0.03464	0.09763	7.24E-06	0.03463	0.03464
129	0.03465	0.09763	7.10E-06	0.03464	0.03466
130	0.03466	0.09763	6.96E-06	0.03466	0.03467
131	0.03468	0.09763	6.81E-06	0.03467	0.03468
132	0.03469	0.09763	6.69E-06	0.03468	0.03470
133	0.03470	0.09763	6.56E-06	0.03469	0.03471
134	0.03471	0.09763	6.42E-06	0.03471	0.03472
135	0.03472	0.09763	6.29E-06	0.03472	0.03473
136	0.03474	0.09763	6.16E-06	0.03473	0.03474
137	0.03475	0.09763	6.03E-06	0.03474	0.03475
138	0.03476	0.09763	5.90E-06	0.03475	0.03476
139	0.03477	0.09763	5.77E-06	0.03476	0.03478
140	0.03478	0.09763	5.65E-06	0.03477	0.03479
141	0.03479	0.09763	5.53E-06	0.03479	0.03480
142	0.03480	0.09763	5.42E-06	0.03480	0.03481
143	0.03481	0.09763	5.31E-06	0.03481	0.03482
144	0.03482	0.09763	5.20E-06	0.03482	0.03483

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
145	0.03483	0.09763	5.09E-06	0.03483	0.03484
146	0.03484	0.09763	4.99E-06	0.03484	0.03485
147	0.03485	0.09763	4.88E-06	0.03484	0.03485
148	0.03486	0.09763	4.78E-06	0.03485	0.03486
149	0.03487	0.09763	4.68E-06	0.03486	0.03487
150	0.03488	0.09763	4.58E-06	0.03487	0.03488
151	0.03488	0.09763	4.48E-06	0.03488	0.03489
152	0.03489	0.09763	4.39E-06	0.03489	0.03490
153	0.03490	0.09763	4.30E-06	0.03490	0.03490
154	0.03491	0.09763	4.21E-06	0.03490	0.03491
155	0.03492	0.09763	4.12E-06	0.03491	0.03492
156	0.03492	0.09763	4.03E-06	0.03492	0.03493
157	0.03493	0.09763	3.95E-06	0.03493	0.03494
158	0.03494	0.09763	3.87E-06	0.03493	0.03494
159	0.03495	0.09763	3.80E-06	0.03494	0.03495
160	0.03495	0.09763	3.73E-06	0.03495	0.03496
161	0.03496	0.09763	3.65E-06	0.03496	0.03496
162	0.03497	0.09763	3.60E-06	0.03496	0.03497
163	0.03497	0.09763	4.50E-06	0.03497	0.03498
164	0.03498	0.09763	4.42E-06	0.03498	0.03499
165	0.03499	0.09763	4.49E-06	0.03499	0.03499
166	0.03500	0.09763	4.96E-06	0.03499	0.03500
167	0.03501	0.09763	4.87E-06	0.03500	0.03501
168	0.03502	0.09763	4.78E-06	0.03501	0.03502
169	0.03503	0.09763	4.69E-06	0.03502	0.03503
170	0.03503	0.09763	4.61E-06	0.03503	0.03504
171	0.03504	0.09763	4.52E-06	0.03504	0.03505
172	0.03505	0.09763	4.44E-06	0.03505	0.03506
173	0.03506	0.09763	4.36E-06	0.03505	0.03506
174	0.03507	0.09763	4.28E-06	0.03506	0.03507
175	0.03507	0.09763	4.21E-06	0.03507	0.03508
176	0.03508	0.09763	4.13E-06	0.03508	0.03509
177	0.03509	0.09763	4.06E-06	0.03509	0.03509
178	0.03510	0.09763	3.99E-06	0.03509	0.03510
179	0.03511	0.09763	3.92E-06	0.03510	0.03511
180	0.03511	0.09763	3.85E-06	0.03511	0.03512
181	0.03512	0.09763	3.78E-06	0.03512	0.03512
182	0.03513	0.09763	3.71E-06	0.03512	0.03513
183	0.03513	0.09763	3.65E-06	0.03513	0.03514
184	0.03514	0.09763	3.58E-06	0.03514	0.03514
185	0.03515	0.09763	3.52E-06	0.03514	0.03515
186	0.03515	0.09763	3.46E-06	0.03515	0.03516
187	0.03516	0.09763	3.39E-06	0.03516	0.03516
188	0.03517	0.09763	3.33E-06	0.03516	0.03517
189	0.03517	0.09763	3.28E-06	0.03517	0.03518
190	0.03518	0.09763	3.22E-06	0.03517	0.03518
191	0.03518	0.09763	3.16E-06	0.03518	0.03519
192	0.03519	0.09763	3.11E-06	0.03519	0.03519

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
193	0.03520	0.09763	3.05E-06	0.03519	0.03520
194	0.03520	0.09763	3.00E-06	0.03520	0.03520
195	0.03521	0.09763	2.95E-06	0.03520	0.03521
196	0.03521	0.09763	2.89E-06	0.03521	0.03522
197	0.03522	0.09763	2.84E-06	0.03521	0.03522
198	0.03522	0.09763	2.79E-06	0.03522	0.03523
199	0.03523	0.09763	2.75E-06	0.03523	0.03523
200	0.03523	0.09763	2.71E-06	0.03523	0.03524
201	0.03524	0.09763	2.67E-06	0.03524	0.03524
202	0.03524	0.09763	3.53E-06	0.03524	0.03525
203	0.03525	0.09763	8.88E-06	0.03524	0.03526
204	0.03527	0.09763	8.77E-06	0.03526	0.03527
205	0.03528	0.09763	8.72E-06	0.03527	0.03529
206	0.03530	0.09763	8.61E-06	0.03529	0.03531
207	0.03531	0.09763	8.51E-06	0.03531	0.03532
208	0.03533	0.09763	8.42E-06	0.03532	0.03534
209	0.03535	0.09763	8.33E-06	0.03534	0.03535
210	0.03536	0.09763	8.24E-06	0.03535	0.03537
211	0.03538	0.09763	8.15E-06	0.03537	0.03538
212	0.03539	0.09763	8.11E-06	0.03538	0.03540
213	0.03541	0.09763	8.01E-06	0.03540	0.03541
214	0.03542	0.09763	7.92E-06	0.03541	0.03543
215	0.03544	0.09763	7.82E-06	0.03543	0.03544
216	0.03545	0.09763	7.73E-06	0.03544	0.03546
217	0.03546	0.09763	7.64E-06	0.03546	0.03547
218	0.03548	0.09763	7.55E-06	0.03547	0.03549
219	0.03549	0.09763	7.60E-06	0.03549	0.03550
220	0.03551	0.09763	7.51E-06	0.03550	0.03551
221	0.03552	0.09763	7.43E-06	0.03551	0.03553
222	0.03553	0.09763	7.34E-06	0.03553	0.03554
223	0.03555	0.09763	7.26E-06	0.03554	0.03556
224	0.03556	0.09763	7.18E-06	0.03555	0.03557
225	0.03557	0.09763	7.11E-06	0.03557	0.03558
226	0.03559	0.09763	7.03E-06	0.03558	0.03560
227	0.03560	0.09763	6.95E-06	0.03559	0.03561
228	0.03561	0.09763	6.88E-06	0.03561	0.03562
229	0.03563	0.09763	6.80E-06	0.03562	0.03563
230	0.03564	0.09763	6.73E-06	0.03563	0.03565
231	0.03565	0.09763	6.66E-06	0.03564	0.03566
232	0.03566	0.09763	6.59E-06	0.03566	0.03567
233	0.03568	0.09763	6.54E-06	0.03567	0.03568
234	0.03569	0.09763	6.47E-06	0.03568	0.03569
235	0.03570	0.09763	6.40E-06	0.03569	0.03571
236	0.03571	0.09763	6.34E-06	0.03571	0.03572
237	0.03572	0.09763	6.29E-06	0.03572	0.03573
238	0.03574	0.09763	6.22E-06	0.03573	0.03574
239	0.03575	0.09763	6.15E-06	0.03574	0.03575
240	0.03576	0.09763	6.08E-06	0.03575	0.03576

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
241	0.03577	0.09763	6.02E-06	0.03576	0.03578
242	0.03578	0.09763	5.96E-06	0.03577	0.03579
243	0.03579	0.09763	5.89E-06	0.03579	0.03580
244	0.03580	0.09763	5.83E-06	0.03580	0.03581
245	0.03581	0.09763	5.78E-06	0.03581	0.03582
246	0.03582	0.09763	5.72E-06	0.03582	0.03583
247	0.03583	0.09763	5.67E-06	0.03583	0.03584
248	0.03585	0.09763	5.61E-06	0.03584	0.03585
249	0.03586	0.09763	5.55E-06	0.03585	0.03586
250	0.03587	0.09763	5.50E-06	0.03586	0.03587
251	0.03588	0.09763	5.44E-06	0.03587	0.03588
252	0.03589	0.09763	5.39E-06	0.03588	0.03589
253	0.03590	0.09763	5.50E-06	0.03589	0.03590
254	0.03591	0.09763	5.48E-06	0.03590	0.03591
255	0.03592	0.09763	5.43E-06	0.03591	0.03592
256	0.03593	0.09763	5.41E-06	0.03592	0.03593
257	0.03594	0.09763	5.37E-06	0.03593	0.03594
258	0.03595	0.09763	5.35E-06	0.03594	0.03595
259	0.03596	0.09763	5.30E-06	0.03595	0.03596
260	0.03597	0.09763	5.27E-06	0.03596	0.03597
261	0.03598	0.09763	5.22E-06	0.03597	0.03598
262	0.03599	0.09763	5.20E-06	0.03598	0.03599
263	0.03600	0.09763	5.15E-06	0.03599	0.03600
264	0.03600	0.09763	5.14E-06	0.03600	0.03601
265	0.03601	0.09763	5.10E-06	0.03601	0.03602
266	0.03602	0.09763	5.06E-06	0.03602	0.03603
267	0.03603	0.09763	5.02E-06	0.03603	0.03604
268	0.03604	0.09763	4.98E-06	0.03604	0.03605
269	0.03605	0.09763	4.94E-06	0.03605	0.03606
270	0.03606	0.09763	4.90E-06	0.03606	0.03607
271	0.03607	0.09763	4.86E-06	0.03606	0.03607
272	0.03608	0.09763	4.82E-06	0.03607	0.03608
273	0.03609	0.09763	4.78E-06	0.03608	0.03609
274	0.03610	0.09763	4.74E-06	0.03609	0.03610
275	0.03611	0.09763	4.71E-06	0.03610	0.03611
276	0.03611	0.09763	4.67E-06	0.03611	0.03612
277	0.03612	0.09763	4.62E-06	0.03612	0.03613
278	0.03613	0.09763	4.58E-06	0.03613	0.03614
279	0.03614	0.09763	4.54E-06	0.03613	0.03614
280	0.03615	0.09763	4.50E-06	0.03614	0.03615
281	0.03616	0.09763	4.46E-06	0.03615	0.03616
282	0.03616	0.09763	4.41E-06	0.03616	0.03617
283	0.03617	0.09763	4.37E-06	0.03617	0.03618
284	0.03618	0.09763	4.37E-06	0.03618	0.03619
285	0.03619	0.09763	4.33E-06	0.03618	0.03619
286	0.03620	0.09763	4.29E-06	0.03619	0.03620
287	0.03620	0.09763	4.25E-06	0.03620	0.03621
288	0.03621	0.09763	4.23E-06	0.03621	0.03622

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
289	0.03622	0.09763	4.19E-06	0.03622	0.03622
290	0.03623	0.09763	4.17E-06	0.03622	0.03623
291	0.03624	0.09763	4.14E-06	0.03623	0.03624
292	0.03624	0.09763	4.10E-06	0.03624	0.03625
293	0.03625	0.09763	4.07E-06	0.03625	0.03626
294	0.03626	0.09763	4.03E-06	0.03625	0.03626
295	0.03627	0.09763	4.01E-06	0.03626	0.03627
296	0.03627	0.09763	3.99E-06	0.03627	0.03628
297	0.03628	0.09763	3.97E-06	0.03628	0.03628
298	0.03629	0.09763	3.93E-06	0.03628	0.03629
299	0.03630	0.09763	3.90E-06	0.03629	0.03630
300	0.03630	0.09763	3.87E-06	0.03630	0.03631
301	0.03631	0.09763	3.84E-06	0.03631	0.03631
302	0.03632	0.09763	3.81E-06	0.03631	0.03632
303	0.03632	0.09763	3.78E-06	0.03632	0.03633
304	0.03633	0.09763	3.75E-06	0.03633	0.03633
305	0.03634	0.09763	3.72E-06	0.03633	0.03634
306	0.03634	0.09763	3.69E-06	0.03634	0.03635
307	0.03635	0.09763	3.67E-06	0.03635	0.03636
308	0.03636	0.09763	3.64E-06	0.03635	0.03636
309	0.03637	0.09763	3.61E-06	0.03636	0.03637
310	0.03637	0.09763	3.58E-06	0.03637	0.03638
311	0.03638	0.09763	3.55E-06	0.03637	0.03638
312	0.03639	0.09763	3.52E-06	0.03638	0.03639
313	0.03639	0.09763	3.50E-06	0.03639	0.03640
314	0.03640	0.09763	3.47E-06	0.03639	0.03640
315	0.03640	0.09763	3.45E-06	0.03640	0.03641
316	0.03641	0.09763	3.41E-06	0.03641	0.03641
317	0.03642	0.09763	3.38E-06	0.03641	0.03642
318	0.03642	0.09763	3.35E-06	0.03642	0.03643
319	0.03643	0.09763	3.33E-06	0.03643	0.03643
320	0.03644	0.09763	3.30E-06	0.03643	0.03644
321	0.03644	0.09763	3.27E-06	0.03644	0.03645
322	0.03645	0.09763	3.25E-06	0.03644	0.03645
323	0.03645	0.09763	3.22E-06	0.03645	0.03646
324	0.03646	0.09763	3.19E-06	0.03646	0.03646
325	0.03647	0.09763	3.19E-06	0.03646	0.03647
326	0.03647	0.09763	3.16E-06	0.03647	0.03647
327	0.03648	0.09763	3.13E-06	0.03647	0.03648
328	0.03648	0.09763	3.13E-06	0.03648	0.03649
329	0.03649	0.09763	3.11E-06	0.03649	0.03649
330	0.03649	0.09763	3.08E-06	0.03649	0.03650
331	0.03650	0.09763	3.66E-06	0.03650	0.03650
332	0.03651	0.09763	3.65E-06	0.03650	0.03651
333	0.03651	0.09763	3.63E-06	0.03651	0.03652
334	0.03652	0.09763	3.62E-06	0.03652	0.03652
335	0.03653	0.09763	3.61E-06	0.03652	0.03653
336	0.03653	0.09763	3.60E-06	0.03653	0.03654

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
337	0.03654	0.09763	3.58E-06	0.03654	0.03654
338	0.03655	0.09763	3.56E-06	0.03654	0.03655
339	0.03655	0.09763	3.55E-06	0.03655	0.03656
340	0.03656	0.09763	3.53E-06	0.03656	0.03656
341	0.03657	0.09763	3.51E-06	0.03656	0.03657
342	0.03657	0.09763	3.49E-06	0.03657	0.03658
343	0.03658	0.09763	3.47E-06	0.03658	0.03658
344	0.03659	0.09763	3.46E-06	0.03658	0.03659
345	0.03659	0.09763	3.46E-06	0.03659	0.03660
346	0.03660	0.09763	3.44E-06	0.03660	0.03660
347	0.03661	0.09763	3.43E-06	0.03660	0.03661
348	0.03661	0.09763	3.41E-06	0.03661	0.03662
349	0.03662	0.09763	3.39E-06	0.03661	0.03662
350	0.03662	0.09763	3.37E-06	0.03662	0.03663
351	0.03663	0.09763	3.35E-06	0.03663	0.03663
352	0.03664	0.09763	3.34E-06	0.03663	0.03664
353	0.03664	0.09763	3.32E-06	0.03664	0.03665
354	0.03665	0.09763	3.30E-06	0.03665	0.03665
355	0.03666	0.09763	3.28E-06	0.03665	0.03666
356	0.03666	0.09763	3.26E-06	0.03666	0.03666
357	0.03667	0.09763	3.24E-06	0.03666	0.03667
358	0.03667	0.09763	3.22E-06	0.03667	0.03668
359	0.03668	0.09763	3.21E-06	0.03668	0.03668
360	0.03669	0.09763	3.19E-06	0.03668	0.03669
361	0.03669	0.09763	3.17E-06	0.03669	0.03669
362	0.03670	0.09763	3.15E-06	0.03669	0.03670
363	0.03670	0.09763	3.14E-06	0.03670	0.03671
364	0.03671	0.09763	3.12E-06	0.03671	0.03671
365	0.03671	0.09763	3.10E-06	0.03671	0.03672
366	0.03672	0.09763	3.08E-06	0.03672	0.03672
367	0.03673	0.09763	3.07E-06	0.03672	0.03673
368	0.03673	0.09763	3.13E-06	0.03673	0.03673
369	0.03674	0.09763	3.12E-06	0.03673	0.03674
370	0.03674	0.09763	3.10E-06	0.03674	0.03675
371	0.03675	0.09763	3.08E-06	0.03675	0.03675
372	0.03675	0.09763	3.07E-06	0.03675	0.03676
373	0.03676	0.09763	3.05E-06	0.03676	0.03676
374	0.03677	0.09763	3.03E-06	0.03676	0.03677
375	0.03677	0.09763	3.02E-06	0.03677	0.03677
376	0.03678	0.09763	3.01E-06	0.03677	0.03678
377	0.03678	0.09763	2.99E-06	0.03678	0.03679
378	0.03679	0.09763	2.97E-06	0.03679	0.03679
379	0.03679	0.09763	2.96E-06	0.03679	0.03680
380	0.03680	0.09763	2.94E-06	0.03680	0.03680
381	0.03680	0.09763	2.93E-06	0.03680	0.03681
382	0.03681	0.09763	2.91E-06	0.03681	0.03681
383	0.03682	0.09763	2.91E-06	0.03681	0.03682
384	0.03682	0.09763	2.90E-06	0.03682	0.03682

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
385	0.03683	0.09763	2.89E-06	0.03682	0.03683
386	0.03683	0.09763	2.88E-06	0.03683	0.03683
387	0.03684	0.09763	2.87E-06	0.03683	0.03684
388	0.03684	0.09763	2.85E-06	0.03684	0.03685
389	0.03685	0.09763	2.84E-06	0.03684	0.03685
390	0.03685	0.09763	2.85E-06	0.03685	0.03686
391	0.03686	0.09763	2.83E-06	0.03686	0.03686
392	0.03686	0.09763	2.82E-06	0.03686	0.03687
393	0.03687	0.09763	2.80E-06	0.03687	0.03687
394	0.03687	0.09763	2.79E-06	0.03687	0.03688
395	0.03688	0.09763	2.78E-06	0.03688	0.03688
396	0.03688	0.09763	2.80E-06	0.03688	0.03689
397	0.03689	0.09763	2.78E-06	0.03689	0.03689
398	0.03689	0.09763	2.77E-06	0.03689	0.03690
399	0.03690	0.09763	2.76E-06	0.03690	0.03690
400	0.03690	0.09763	2.74E-06	0.03690	0.03691
401	0.03691	0.09763	2.73E-06	0.03691	0.03691
402	0.03691	0.09763	2.72E-06	0.03691	0.03692
403	0.03692	0.09763	2.71E-06	0.03692	0.03692
404	0.03692	0.09763	2.70E-06	0.03692	0.03693
405	0.03693	0.09763	2.69E-06	0.03693	0.03693
406	0.03693	0.09763	2.69E-06	0.03693	0.03694
407	0.03694	0.09763	2.68E-06	0.03694	0.03694
408	0.03694	0.09763	2.66E-06	0.03694	0.03695
409	0.03695	0.09763	2.65E-06	0.03695	0.03695
410	0.03695	0.09763	2.64E-06	0.03695	0.03696
411	0.03696	0.09763	2.63E-06	0.03696	0.03696
412	0.03696	0.09763	2.61E-06	0.03696	0.03697
413	0.03697	0.09763	2.60E-06	0.03697	0.03697
414	0.03697	0.09763	2.59E-06	0.03697	0.03698
415	0.03698	0.09763	2.58E-06	0.03698	0.03698
416	0.03698	0.09763	2.57E-06	0.03698	0.03699
417	0.03699	0.09763	2.55E-06	0.03699	0.03699
418	0.03699	0.09763	2.54E-06	0.03699	0.03700
419	0.03700	0.09763	2.53E-06	0.03700	0.03700
420	0.03700	0.09763	2.52E-06	0.03700	0.03700
421	0.03701	0.09763	2.51E-06	0.03700	0.03701
422	0.03701	0.09763	2.50E-06	0.03701	0.03701
423	0.03702	0.09763	2.49E-06	0.03701	0.03702
424	0.03702	0.09763	2.47E-06	0.03702	0.03702
425	0.03703	0.09763	2.46E-06	0.03702	0.03703
426	0.03703	0.09763	2.45E-06	0.03703	0.03703
427	0.03703	0.09763	2.44E-06	0.03703	0.03704
428	0.03704	0.09763	2.43E-06	0.03704	0.03704
429	0.03704	0.09763	2.42E-06	0.03704	0.03705
430	0.03705	0.09763	2.41E-06	0.03705	0.03705
431	0.03705	0.09763	2.40E-06	0.03705	0.03705
432	0.03706	0.09763	2.39E-06	0.03705	0.03706

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
433	0.03706	0.09763	2.38E-06	0.03706	0.03706
434	0.03707	0.09763	2.45E-06	0.03706	0.03707
435	0.03707	0.09763	2.44E-06	0.03707	0.03707
436	0.03707	0.09763	2.43E-06	0.03707	0.03708
437	0.03708	0.09763	2.42E-06	0.03708	0.03708
438	0.03708	0.09763	2.41E-06	0.03708	0.03709
439	0.03709	0.09763	2.41E-06	0.03709	0.03709
440	0.03709	0.09763	2.40E-06	0.03709	0.03710
441	0.03710	0.09763	2.39E-06	0.03709	0.03710
442	0.03710	0.09763	2.40E-06	0.03710	0.03710
443	0.03711	0.09763	2.39E-06	0.03710	0.03711
444	0.03711	0.09763	2.38E-06	0.03711	0.03711
445	0.03711	0.09763	2.37E-06	0.03711	0.03712
446	0.03712	0.09763	2.36E-06	0.03712	0.03712
447	0.03712	0.09763	2.35E-06	0.03712	0.03713
448	0.03713	0.09763	2.36E-06	0.03713	0.03713
449	0.03713	0.09763	2.36E-06	0.03713	0.03713
450	0.03714	0.09763	2.35E-06	0.03713	0.03714
451	0.03714	0.09763	2.35E-06	0.03714	0.03714
452	0.03715	0.09763	2.42E-06	0.03714	0.03715
453	0.03715	0.09763	2.79E-06	0.03715	0.03715
454	0.03716	0.09763	2.84E-06	0.03715	0.03716
455	0.03716	0.09763	2.86E-06	0.03716	0.03716
456	0.03717	0.09763	2.88E-06	0.03716	0.03717
457	0.03717	0.09763	2.90E-06	0.03717	0.03717
458	0.03718	0.09763	2.92E-06	0.03717	0.03718
459	0.03718	0.09763	2.91E-06	0.03718	0.03718
460	0.03719	0.09763	2.91E-06	0.03718	0.03719
461	0.03719	0.09763	2.90E-06	0.03719	0.03720
462	0.03720	0.09763	2.90E-06	0.03719	0.03720
463	0.03720	0.09763	2.89E-06	0.03720	0.03721
464	0.03721	0.09763	2.89E-06	0.03721	0.03721
465	0.03721	0.09763	2.88E-06	0.03721	0.03722
466	0.03722	0.09763	2.88E-06	0.03722	0.03722
467	0.03722	0.09763	2.87E-06	0.03722	0.03723
468	0.03723	0.09763	2.87E-06	0.03723	0.03723
469	0.03724	0.09763	2.86E-06	0.03723	0.03724
470	0.03724	0.09763	2.86E-06	0.03724	0.03724
471	0.03725	0.09763	2.86E-06	0.03724	0.03725
472	0.03725	0.09763	2.85E-06	0.03725	0.03725
473	0.03726	0.09763	2.85E-06	0.03725	0.03726
474	0.03726	0.09763	2.84E-06	0.03726	0.03726
475	0.03727	0.09763	2.84E-06	0.03726	0.03727
476	0.03727	0.09763	2.87E-06	0.03727	0.03727
477	0.03728	0.09763	2.87E-06	0.03727	0.03728
478	0.03728	0.09763	2.86E-06	0.03728	0.03729
479	0.03729	0.09763	2.86E-06	0.03729	0.03729
480	0.03729	0.09763	2.85E-06	0.03729	0.03730

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
481	0.03730	0.09763	2.85E-06	0.03730	0.03730
482	0.03730	0.09763	2.84E-06	0.03730	0.03731
483	0.03731	0.09763	2.84E-06	0.03731	0.03731
484	0.03731	0.09763	2.84E-06	0.03731	0.03732
485	0.03732	0.09763	2.83E-06	0.03732	0.03732
486	0.03732	0.09763	2.83E-06	0.03732	0.03733
487	0.03733	0.09763	2.82E-06	0.03733	0.03733
488	0.03734	0.09763	2.82E-06	0.03733	0.03734
489	0.03734	0.09763	2.84E-06	0.03734	0.03734
490	0.03735	0.09763	2.84E-06	0.03734	0.03735
491	0.03735	0.09763	2.84E-06	0.03735	0.03735
492	0.03736	0.09763	2.83E-06	0.03735	0.03736
493	0.03736	0.09763	2.83E-06	0.03736	0.03736
494	0.03737	0.09763	2.82E-06	0.03736	0.03737
495	0.03737	0.09763	2.82E-06	0.03737	0.03737
496	0.03738	0.09763	2.82E-06	0.03737	0.03738
497	0.03738	0.09763	2.81E-06	0.03738	0.03739
498	0.03739	0.09763	2.81E-06	0.03738	0.03739
499	0.03739	0.09763	2.81E-06	0.03739	0.03740
500	0.03740	0.09763	2.80E-06	0.03740	0.03740
501	0.03740	0.09763	2.80E-06	0.03740	0.03741
502	0.03741	0.09763	2.80E-06	0.03741	0.03741
503	0.03741	0.09763	2.79E-06	0.03741	0.03742
504	0.03742	0.09763	2.80E-06	0.03742	0.03742
505	0.03742	0.09763	2.80E-06	0.03742	0.03743
506	0.03743	0.09763	2.80E-06	0.03743	0.03743
507	0.03743	0.09763	2.79E-06	0.03743	0.03744
508	0.03744	0.09763	2.79E-06	0.03744	0.03744
509	0.03744	0.09763	2.79E-06	0.03744	0.03745
510	0.03745	0.09763	2.79E-06	0.03745	0.03745
511	0.03745	0.09763	2.78E-06	0.03745	0.03746
512	0.03746	0.09763	2.78E-06	0.03746	0.03746
513	0.03747	0.09763	2.78E-06	0.03746	0.03747
514	0.03747	0.09763	2.77E-06	0.03747	0.03747
515	0.03748	0.09763	2.77E-06	0.03747	0.03748
516	0.03748	0.09763	2.77E-06	0.03748	0.03748
517	0.03749	0.09763	2.76E-06	0.03748	0.03749
518	0.03749	0.09763	2.77E-06	0.03749	0.03749
519	0.03750	0.09763	2.80E-06	0.03749	0.03750
520	0.03750	0.09763	2.80E-06	0.03750	0.03750
521	0.03751	0.09763	2.80E-06	0.03750	0.03751
522	0.03751	0.09763	2.79E-06	0.03751	0.03751
523	0.03752	0.09763	2.79E-06	0.03751	0.03752
524	0.03752	0.09763	2.79E-06	0.03752	0.03752
525	0.03753	0.09763	2.79E-06	0.03752	0.03753
526	0.03753	0.09763	2.78E-06	0.03753	0.03753
527	0.03754	0.09763	2.79E-06	0.03753	0.03754
528	0.03754	0.09763	2.81E-06	0.03754	0.03755

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
529	0.03755	0.09763	2.81E-06	0.03754	0.03755
530	0.03755	0.09763	2.80E-06	0.03755	0.03756
531	0.03756	0.09763	2.80E-06	0.03756	0.03756
532	0.03756	0.09763	2.80E-06	0.03756	0.03757
533	0.03757	0.09763	2.80E-06	0.03757	0.03757
534	0.03757	0.09763	2.80E-06	0.03757	0.03758
535	0.03758	0.09763	2.79E-06	0.03758	0.03758
536	0.03758	0.09763	2.79E-06	0.03758	0.03759
537	0.03759	0.09763	2.79E-06	0.03759	0.03759
538	0.03759	0.09763	2.79E-06	0.03759	0.03760
539	0.03760	0.09763	2.81E-06	0.03760	0.03760
540	0.03760	0.09763	2.81E-06	0.03760	0.03761
541	0.03761	0.09763	2.81E-06	0.03761	0.03761
542	0.03761	0.09763	2.81E-06	0.03761	0.03762
543	0.03762	0.09763	2.81E-06	0.03762	0.03762
544	0.03763	0.09763	2.81E-06	0.03762	0.03763
545	0.03763	0.09763	2.80E-06	0.03763	0.03763
546	0.03764	0.09763	2.80E-06	0.03763	0.03764
547	0.03764	0.09763	2.82E-06	0.03764	0.03764
548	0.03765	0.09763	2.86E-06	0.03764	0.03765
549	0.03765	0.09763	2.86E-06	0.03765	0.03765
550	0.03766	0.09763	2.86E-06	0.03765	0.03766
551	0.03766	0.09763	2.86E-06	0.03766	0.03766
552	0.03767	0.09763	2.86E-06	0.03766	0.03767
553	0.03767	0.09763	2.86E-06	0.03767	0.03768
554	0.03768	0.09763	2.86E-06	0.03767	0.03768
555	0.03768	0.09763	2.85E-06	0.03768	0.03769
556	0.03769	0.09763	2.85E-06	0.03769	0.03769
557	0.03769	0.09763	2.85E-06	0.03769	0.03770
558	0.03770	0.09763	2.85E-06	0.03770	0.03770
559	0.03770	0.09763	2.85E-06	0.03770	0.03771
560	0.03771	0.09763	2.85E-06	0.03771	0.03771
561	0.03771	0.09763	2.85E-06	0.03771	0.03772
562	0.03772	0.09763	2.85E-06	0.03772	0.03772
563	0.03773	0.09763	2.85E-06	0.03772	0.03773
564	0.03773	0.09763	2.85E-06	0.03773	0.03773
565	0.03774	0.09763	2.85E-06	0.03773	0.03774
566	0.03774	0.09763	2.85E-06	0.03774	0.03774
567	0.03775	0.09763	2.85E-06	0.03774	0.03775
568	0.03775	0.09763	2.84E-06	0.03775	0.03775
569	0.03776	0.09763	2.84E-06	0.03775	0.03776
570	0.03776	0.09763	2.84E-06	0.03776	0.03777
571	0.03777	0.09763	2.84E-06	0.03776	0.03777
572	0.03777	0.09763	2.84E-06	0.03777	0.03778
573	0.03778	0.09763	2.84E-06	0.03778	0.03778
574	0.03778	0.09763	2.84E-06	0.03778	0.03779
575	0.03779	0.09763	2.84E-06	0.03779	0.03779
576	0.03779	0.09763	2.84E-06	0.03779	0.03780

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
577	0.03780	0.09763	2.84E-06	0.03780	0.03780
578	0.03780	0.09763	2.84E-06	0.03780	0.03781
579	0.03781	0.09763	2.84E-06	0.03781	0.03781
580	0.03781	0.09763	2.84E-06	0.03781	0.03782
581	0.03782	0.09763	2.84E-06	0.03782	0.03782
582	0.03783	0.09763	2.84E-06	0.03782	0.03783
583	0.03783	0.09763	2.84E-06	0.03783	0.03783
584	0.03784	0.09763	2.84E-06	0.03783	0.03784
585	0.03784	0.09763	2.91E-06	0.03784	0.03784
586	0.03785	0.09763	2.96E-06	0.03784	0.03785
587	0.03785	0.09763	2.96E-06	0.03785	0.03785
588	0.03786	0.09763	2.96E-06	0.03785	0.03786
589	0.03786	0.09763	2.96E-06	0.03786	0.03787
590	0.03787	0.09763	2.96E-06	0.03787	0.03787
591	0.03787	0.09763	2.97E-06	0.03787	0.03788
592	0.03788	0.09763	3.40E-06	0.03788	0.03788
593	0.03789	0.09763	3.48E-06	0.03788	0.03789
594	0.03789	0.09763	3.56E-06	0.03789	0.03790
595	0.03790	0.09763	3.56E-06	0.03790	0.03790
596	0.03791	0.09763	3.57E-06	0.03790	0.03791
597	0.03791	0.09763	3.57E-06	0.03791	0.03792
598	0.03792	0.09763	3.58E-06	0.03791	0.03792
599	0.03793	0.09763	4.27E-06	0.03792	0.03793
600	0.03793	0.09763	8.65E-06	0.03792	0.03794
601	0.03795	0.09763	8.71E-06	0.03794	0.03796
602	0.03797	0.09763	8.77E-06	0.03796	0.03797
603	0.03798	0.09763	8.81E-06	0.03797	0.03799
604	0.03800	0.09763	8.88E-06	0.03799	0.03801
605	0.03801	0.09763	8.93E-06	0.03801	0.03802
606	0.03803	0.09763	8.98E-06	0.03802	0.03804
607	0.03805	0.09763	9.03E-06	0.03804	0.03806
608	0.03806	0.09763	9.13E-06	0.03805	0.03807
609	0.03808	0.09763	9.18E-06	0.03807	0.03809
610	0.03810	0.09763	9.24E-06	0.03809	0.03811
611	0.03811	0.09763	9.29E-06	0.03811	0.03812
612	0.03813	0.09763	9.34E-06	0.03812	0.03814
613	0.03815	0.09763	9.40E-06	0.03814	0.03816
614	0.03817	0.09763	9.45E-06	0.03816	0.03818
615	0.03818	0.09763	9.51E-06	0.03817	0.03819
616	0.03820	0.09763	9.57E-06	0.03819	0.03821
617	0.03822	0.09763	9.71E-06	0.03821	0.03823
618	0.03824	0.09763	9.98E-06	0.03823	0.03825
619	0.03826	0.09763	1.01E-05	0.03825	0.03827
620	0.03827	0.09763	1.02E-05	0.03826	0.03828
621	0.03829	0.09763	1.03E-05	0.03828	0.03830
622	0.03831	0.09763	1.04E-05	0.03830	0.03832
623	0.03833	0.09763	1.05E-05	0.03832	0.03834
624	0.03835	0.09763	1.06E-05	0.03834	0.03836

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
625	0.03837	0.09763	1.07E-05	0.03836	0.03838
626	0.03839	0.09763	1.08E-05	0.03838	0.03840
627	0.03841	0.09763	1.11E-05	0.03840	0.03842
628	0.03843	0.09763	1.16E-05	0.03842	0.03844
629	0.03845	0.09763	1.18E-05	0.03844	0.03846
630	0.03847	0.09763	1.20E-05	0.03846	0.03849
631	0.03850	0.09763	1.21E-05	0.03848	0.03851
632	0.03852	0.09763	1.23E-05	0.03851	0.03853
633	0.03854	0.09763	1.24E-05	0.03853	0.03855
634	0.03857	0.09763	1.27E-05	0.03855	0.03858
635	0.03859	0.09763	1.28E-05	0.03858	0.03860
636	0.03861	0.09763	1.30E-05	0.03860	0.03863
637	0.03864	0.09763	1.32E-05	0.03862	0.03865
638	0.03866	0.09763	1.34E-05	0.03865	0.03867
639	0.03869	0.09763	1.36E-05	0.03867	0.03870
640	0.03871	0.09763	1.37E-05	0.03870	0.03872
641	0.03874	0.09763	1.39E-05	0.03872	0.03875
642	0.03876	0.09763	1.42E-05	0.03875	0.03878
643	0.03879	0.09763	1.44E-05	0.03877	0.03880
644	0.03881	0.09763	1.46E-05	0.03880	0.03883
645	0.03884	0.09763	1.48E-05	0.03883	0.03886
646	0.03887	0.09763	1.50E-05	0.03885	0.03888
647	0.03890	0.09763	1.52E-05	0.03888	0.03891
648	0.03892	0.09763	1.55E-05	0.03891	0.03894
649	0.03895	0.09763	1.57E-05	0.03894	0.03897
650	0.03898	0.09762	1.60E-05	0.03897	0.03900
651	0.03901	0.09762	1.62E-05	0.03900	0.03903
652	0.03904	0.09762	1.65E-05	0.03903	0.03906
653	0.03907	0.09762	1.67E-05	0.03906	0.03909
654	0.03910	0.09762	1.70E-05	0.03909	0.03912
655	0.03913	0.09762	1.72E-05	0.03912	0.03915
656	0.03917	0.09762	1.75E-05	0.03915	0.03918
657	0.03920	0.09762	1.78E-05	0.03918	0.03922
658	0.03923	0.09762	1.81E-05	0.03921	0.03925
659	0.03927	0.09762	1.84E-05	0.03925	0.03928
660	0.03930	0.09762	1.87E-05	0.03928	0.03932
661	0.03933	0.09762	1.90E-05	0.03932	0.03935
662	0.03937	0.09762	1.93E-05	0.03935	0.03939
663	0.03940	0.09762	1.96E-05	0.03939	0.03942
664	0.03944	0.09762	1.99E-05	0.03942	0.03946
665	0.03948	0.09762	2.02E-05	0.03946	0.03950
666	0.03952	0.09762	2.06E-05	0.03949	0.03954
667	0.03955	0.09762	2.11E-05	0.03953	0.03957
668	0.03959	0.09762	2.14E-05	0.03957	0.03961
669	0.03963	0.09762	2.18E-05	0.03961	0.03965
670	0.03967	0.09762	2.23E-05	0.03965	0.03969
671	0.03971	0.09762	2.26E-05	0.03969	0.03974
672	0.03976	0.09762	2.30E-05	0.03973	0.03978

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
673	0.03980	0.09762	2.36E-05	0.03977	0.03982
674	0.03984	0.09762	2.41E-05	0.03982	0.03987
675	0.03989	0.09762	2.45E-05	0.03986	0.03991
676	0.03993	0.09762	2.50E-05	0.03991	0.03996
677	0.03998	0.09762	2.56E-05	0.03995	0.04000
678	0.04003	0.09762	2.61E-05	0.04000	0.04005
679	0.04007	0.09762	2.67E-05	0.04005	0.04010
680	0.04012	0.09762	2.72E-05	0.04010	0.04015
681	0.04017	0.09762	2.78E-05	0.04015	0.04020
682	0.04022	0.09762	2.84E-05	0.04020	0.04025
683	0.04028	0.09762	2.91E-05	0.04025	0.04031
684	0.04033	0.09762	2.97E-05	0.04030	0.04036
685	0.04039	0.09762	3.03E-05	0.04036	0.04042
686	0.04044	0.09762	3.10E-05	0.04041	0.04047
687	0.04050	0.09762	3.23E-05	0.04047	0.04053
688	0.04056	0.09762	3.30E-05	0.04053	0.04059
689	0.04062	0.09762	3.37E-05	0.04059	0.04065
690	0.04068	0.09762	3.45E-05	0.04065	0.04072
691	0.04075	0.09762	3.53E-05	0.04071	0.04078
692	0.04081	0.09762	3.62E-05	0.04078	0.04085
693	0.04088	0.09762	3.69E-05	0.04084	0.04092
694	0.04095	0.09762	3.79E-05	0.04091	0.04099
695	0.04102	0.09762	3.89E-05	0.04098	0.04106
696	0.04109	0.09762	3.99E-05	0.04105	0.04113
697	0.04116	0.09762	4.10E-05	0.04112	0.04120
698	0.04124	0.09762	4.20E-05	0.04120	0.04128
699	0.04132	0.09762	4.33E-05	0.04127	0.04136
700	0.04140	0.09762	4.45E-05	0.04135	0.04144
701	0.04148	0.09762	4.57E-05	0.04143	0.04152
702	0.04156	0.09762	4.70E-05	0.04152	0.04161
703	0.04165	0.09761	4.83E-05	0.04160	0.04170
704	0.04174	0.09761	4.97E-05	0.04169	0.04179
705	0.04183	0.09761	5.12E-05	0.04178	0.04188
706	0.04193	0.09761	5.27E-05	0.04187	0.04198
707	0.04202	0.09761	5.43E-05	0.04197	0.04208
708	0.04212	0.09761	5.62E-05	0.04207	0.04218
709	0.04223	0.09761	5.82E-05	0.04217	0.04229
710	0.04234	0.09761	6.01E-05	0.04228	0.04240
711	0.04245	0.09761	6.20E-05	0.04239	0.04251
712	0.04256	0.09761	6.42E-05	0.04250	0.04263
713	0.04268	0.09761	6.64E-05	0.04261	0.04275
714	0.04280	0.09761	6.87E-05	0.04274	0.04287
715	0.04293	0.09761	7.10E-05	0.04286	0.04300
716	0.04306	0.09761	7.35E-05	0.04299	0.04314
717	0.04320	0.09761	7.60E-05	0.04312	0.04327
718	0.04334	0.09760	7.89E-05	0.04326	0.04342
719	0.04348	0.09760	8.17E-05	0.04340	0.04357
720	0.04364	0.09760	8.47E-05	0.04355	0.04372

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
721	0.04379	0.09760	8.82E-05	0.04370	0.04388
722	0.04396	0.09760	9.18E-05	0.04386	0.04405
723	0.04413	0.09760	9.56E-05	0.04403	0.04422
724	0.04430	0.09760	9.95E-05	0.04420	0.04440
725	0.04449	0.09760	1.04E-04	0.04438	0.04459
726	0.04468	0.09760	1.09E-04	0.04457	0.04479
727	0.04488	0.09759	1.13E-04	0.04477	0.04499
728	0.04509	0.09759	1.19E-04	0.04497	0.04521
729	0.04531	0.09759	1.24E-04	0.04518	0.04543
730	0.04554	0.09759	1.29E-04	0.04541	0.04567
731	0.04578	0.09759	1.35E-04	0.04564	0.04591
732	0.04603	0.09758	1.42E-04	0.04588	0.04617
733	0.04629	0.09758	1.49E-04	0.04614	0.04644
734	0.04657	0.09758	1.56E-04	0.04641	0.04672
735	0.04685	0.09758	1.64E-04	0.04669	0.04702
736	0.04716	0.09758	1.72E-04	0.04698	0.04733
737	0.04748	0.09757	1.82E-04	0.04729	0.04766
738	0.04781	0.09757	1.92E-04	0.04762	0.04800
739	0.04817	0.09757	2.03E-04	0.04797	0.04837
740	0.04854	0.09756	2.18E-04	0.04833	0.04876
741	0.04895	0.09756	2.34E-04	0.04871	0.04918
742	0.04938	0.09755	2.49E-04	0.04913	0.04963
743	0.04984	0.09755	2.65E-04	0.04957	0.05011
744	0.05033	0.09754	2.84E-04	0.05005	0.05062
745	0.05086	0.09753	3.07E-04	0.05055	0.05116
746	0.05143	0.09752	3.30E-04	0.05110	0.05176
747	0.05204	0.09752	3.59E-04	0.05168	0.05240
748	0.05270	0.09751	3.88E-04	0.05231	0.05309
749	0.05342	0.09750	4.22E-04	0.05300	0.05384
750	0.05420	0.09748	4.60E-04	0.05374	0.05466
751	0.05505	0.09747	5.04E-04	0.05455	0.05556
752	0.05598	0.09745	5.56E-04	0.05543	0.05654
753	0.05701	0.09743	6.15E-04	0.05640	0.05763
754	0.05815	0.09741	6.83E-04	0.05747	0.05883
755	0.05941	0.09739	7.64E-04	0.05865	0.06018
756	0.06083	0.09736	8.60E-04	0.05997	0.06169
757	0.06242	0.09732	9.74E-04	0.06144	0.06339
758	0.06422	0.09728	1.11E-03	0.06311	0.06533
759	0.06627	0.09722	1.28E-03	0.06499	0.06755
760	0.06864	0.09716	1.49E-03	0.06715	0.07012
761	0.07139	0.09707	1.75E-03	0.06964	0.07313
762	0.07462	0.09697	2.08E-03	0.07254	0.07670
763	0.07847	0.09683	2.51E-03	0.07595	0.08098
764	0.08312	0.09665	3.10E-03	0.08002	0.08622
765	0.08885	0.09639	3.89E-03	0.08496	0.09274
766	0.09605	0.09603	5.02E-03	0.09103	0.10107
767	0.10533	0.09551	6.67E-03	0.09866	0.11200
768	0.11766	0.09491	8.57E-03	0.10909	0.12623

i	w(i)	f(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
769	0.13351	0.09386	1.19E-02	0.12165	0.14537
770	0.15545	0.09161	1.89E-02	0.13651	0.17439
771	0.19049	0.08886	2.76E-02	0.16290	0.21808
772	0.24153	0.08769	3.13E-02	0.21028	0.27279

A.3 The BSA results in Chapter 5

The BSA displayed a message every $1,000,000$ steps. Figure presents the output of these messages for (approximately) every $1,000,000,000$ steps. Here, the first column represents progress, in terms of the percentage of the area of the initial box for which the inequality (5.7) is verified. The second column is the iteration number. Figure presents the graphical illustration how progress depends on the number of iterations.

Persentage of r	n
4.5964%	1000000000
9.6894%	2000000000
28.4273%	3000000000
95.7067%	4000000000
99.2349%	5000000000

Table 6.2: The table of percentage of r and n

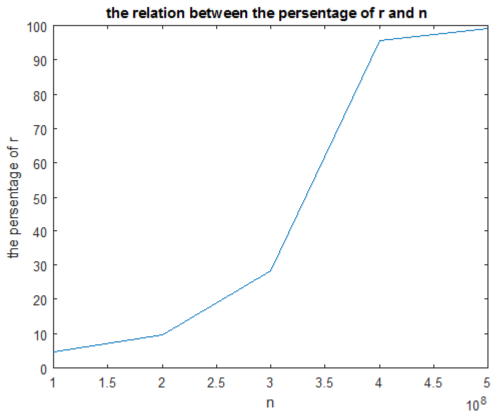

Figure 6.15: The graph of percentage of r and n

A.4 Results for Theorem 5.7 in Chapter 5

>> maincodethm2_2

i	w(i)	F(w(i))	d(i)	w(i)-d(i)	w(i)+d(i)
1	0.0001000	0.0963721	8.25587e-03	-0.00815587	0.00835587
2	0.0153734	0.0967305	7.53910e-03	0.00783426	0.02291246
3	0.0293207	0.0970951	6.80971e-03	0.02251098	0.03613041
4	0.0419187	0.0974260	6.14805e-03	0.03577061	0.04806671
5	0.0532926	0.0977263	5.54746e-03	0.04774510	0.05884002
6	0.0635554	0.0979991	5.00175e-03	0.05855362	0.06855712
7	0.0728086	0.0982460	4.50792e-03	0.06830068	0.07731652
8	0.0811483	0.0984705	4.05894e-03	0.07708931	0.08520719
9	0.0886573	0.0986731	3.65373e-03	0.08500356	0.09231103
10	0.0954167	0.0988572	3.28555e-03	0.09213115	0.09870225
11	0.1014950	0.0990234	2.95312e-03	0.09854185	0.10444809
12	0.1069582	0.0991735	2.65309e-03	0.10430515	0.10961133
13	0.1118665	0.0993096	2.38081e-03	0.10948565	0.11424726
14	0.1162709	0.0994319	2.13611e-03	0.11413484	0.11840705
15	0.1202227	0.0995421	1.91571e-03	0.11830704	0.12213845
16	0.1237668	0.0996420	1.71600e-03	0.12205080	0.12548280
17	0.1269414	0.0997317	1.53666e-03	0.12540473	0.12847806
18	0.1297842	0.0998120	1.37607e-03	0.12840815	0.13116029
19	0.1323299	0.0998842	1.23152e-03	0.13109842	0.13356147
20	0.1346083	0.0999496	1.10090e-03	0.13350737	0.13570916
21	0.1366449	0.1000081	9.83794e-04	0.13566112	0.13762871
22	0.1384649	0.1000604	8.79149e-04	0.13758579	0.13934409
23	0.1400914	0.1000936	8.12756e-04	0.13927861	0.14090412
24	0.1415950	0.1000397	9.20605e-04	0.14067435	0.14251557
25	0.1432981	0.1000635	8.72954e-04	0.14242513	0.14417103
26	0.1449130	0.1000862	8.27536e-04	0.14408551	0.14574058
27	0.1464440	0.1001015	7.96941e-04	0.14564705	0.14724093
28	0.1479183	0.1001288	7.42421e-04	0.14717591	0.14866075
29	0.1492918	0.1001522	6.95512e-04	0.14859630	0.14998732
30	0.1505785	0.1001756	6.48763e-04	0.14992974	0.15122727
31	0.1517787	0.1001965	6.07022e-04	0.15117169	0.15238574
32	0.1529017	0.1002157	5.68515e-04	0.15233319	0.15347022
33	0.1539535	0.1002333	5.33428e-04	0.15342003	0.15448689
34	0.1549403	0.1002494	5.01193e-04	0.15443911	0.15544149
35	0.1558675	0.1002640	4.71907e-04	0.15539560	0.15633941
36	0.1567405	0.1002778	4.44361e-04	0.15629617	0.15718489
37	0.1575626	0.1002907	4.18624e-04	0.15714398	0.15798123
38	0.1583371	0.1003026	3.94887e-04	0.15794217	0.15873194
39	0.1590676	0.1003143	3.71412e-04	0.15869619	0.15943901
40	0.1597547	0.1003252	3.49621e-04	0.15940509	0.16010433
41	0.1604015	0.1003354	3.29133e-04	0.16007238	0.16073064
42	0.1610104	0.1003449	3.10126e-04	0.16070028	0.16132053
43	0.1615841	0.1003535	2.93044e-04	0.16129109	0.16187718
44	0.1621263	0.1003614	2.77218e-04	0.16184905	0.16240349
45	0.1626391	0.1003682	2.63584e-04	0.16237554	0.16290271
46	0.1631268	0.1003758	2.48358e-04	0.16287840	0.16337511
47	0.1635862	0.1003813	2.37337e-04	0.16334888	0.16382355
48	0.1640253	0.1003870	2.26049e-04	0.16379924	0.16425134
49	0.1644435	0.1003934	2.13275e-04	0.16423021	0.16465676
50	0.1648380	0.1003974	2.05240e-04	0.16463280	0.16504328
51	0.1652177	0.1004035	1.93015e-04	0.16502472	0.16541075

References

[1] Ani Adhikari and Jim Pitman. The shortest planar arc of width 1. *The American Mathematical Monthly*, 96(4):309–327, 1989.

[2] John C Baez, Karine Bagdasaryan, and Philip Gibbs. The lebesgue universal covering problem. *arXiv preprint arXiv:1502.01251*, 2015.

[3] Granville Barnett and Luca Del Tongo. Data structures and algorithms: Annotated reference with examples, 2008.

[4] AS Besicovitch. On arcs that cannot be covered by an open equilateral triangle of side 1. *The Mathematical Gazette*, 49(369):286–288, 1965.

[5] P. Brass and M. Sharifi. A lower bound for lebesgue's universal cover problem. *International Journal of Computational Geometry & Applications*, 15(05):537–544, 2005.

[6] G.D. Chakerian and M.S. Klamkin. Minimal covers for closed curves. *Mathematics Magazine*, 46(2):55–61, 1973.

[7] P Coulton and Y Movshovich. Besicovitch triangles cover unit arcs. *Geometriae Dedicata*, 123(1):79–88, 2006.

[8] GFD Duff. A smaller universal cover for sets of unit diameter. *CR Math. Acad. Sci*, 2:37–42, 1980.

[9] H.G. Eggleston. *Problems in Euclidean space: Application of convexity.* Courier Corporation, 2007.

[10] Gy Elekes. Generalized breadths, circular cantor sets, and the least area ucc. *Discrete & Computational Geometry*, 12(4):439–449, 1994.

[11] I. Fáry and L. Rédei. Der zentralsymmetrische kern und die zentralsymmetrische hülle von konvexen körpern. *Mathematische Annalen*, 122(3):205–220, 1950.

[12] Steven R Finch and John E Wetzel. Lost in a forest. *The American Mathematical Monthly*, 111(8):645–654, 2004.

[13] Z. Füredi and J. Wetzel. Covers for closed curves of length two. *Periodica Mathematica Hungarica*, 63(1):1–17, 2011.

[14] J Gerriets. An improved solution to moser's worm problem. *Unpublished*, 1972.

[15] John Gerriets and George Poole. Convex regions which cover arcs of constant length. *The American Mathematical Monthly*, 81(1):36–41, 1974.

[16] P. Gibbs. An upper bound for lebesgue's covering problem. *arXiv preprint math/0701393*, 2018.

[17] Philip Gibbs. Lost in an isosceles triangle.

[18] Philip Gibbs. Bellman's escape problem for convex polygons, 2016.

[19] PJ Giblin and PA Holtom. The centre symmetry set. *Banach Center Publications*, 50:91–105, 1999.

[20] R Graham. Problem 41, proceedings. In *1963 Number Theory Conference, University of Colorado, Boulder*, 1963.

[21] Bogdan Grechuk and Sittichoke Som-Am. A convex cover for closed unit curves has area at least 0.0975. *arXiv preprint arXiv:1905.00333*, 2019.

[22] Bogdan Grechuk and Sittichoke Som-am. A convex cover for closed unit curves has area at least 0.1. *arXiv preprint arXiv:2004.03063*, 2020.

[23] Oliver Alfred Gross. A search problem due to bellman. 1955.

[24] HC Hansen. Small universal covers for sets of unit diameter. *Geometriae Dedicata*, 42(2):205–213, 1992.

[25] SC Hille and MTM Emmerich. Global optimization for lipschitz continuous expensive black box functions.

[26] JR Isbell. An optimal search pattern. *Naval Research Logistics Quarterly*, 4(4):357–359, 1957.

[27] Joseph A Johnson, George D Poole, and John E Wetzel. A small cover for convex unit arcs. *Discrete & Computational Geometry*, 32(1):141–147, 2004.

[28] JP Jones and J Schaer. The worm problem. *University of Calgary research paper*, (100), 1970.

[29] Heinrich WE Jung. Über den kleinsten kreis, der eine ebene figur einschließt. *Journal für die reine und angewandte Mathematik*, 137:310–313, 1910.

[30] T. Khandhawit, D. Pagonakis, and S. Sriswasdi. Lower bound for convex hull area and universal cover problems. *International Journal of Computational Geometry & Applications*, 23(03):197–212, 2013.

[31] Tirasan Khandhawit and Sira Sriswasdi. An improved lower bound for moser's worm problem. *arXiv preprint math/0701391*, 2007.

[32] Jinhyo Kim. *Iterated grid search algorithm on unimodal criteria*. PhD thesis, Virginia Tech, 1997.

[33] M. Laidacker and G. Poole. On the existence of minimal covers for families of closed bounded convex sets. *Unpublished*, 1986.

[34] Henri Lebesgue. Sur le problème des isopérimètres et sur les domaines de largeur constante. *Bull. Soc. Math. France, CR*, 42:72–76, 1914.

[35] J.M. Marstrand. Packing smooth curves in r^q. *Mathematika*, 26(1):1–12, 1979.

[36] L. Moser. Poorly formulated unsolved problems of combinatorial geometry. *Mimeographed list*, 1966.

[37] Y Movshovich. Besicovitch triangles extended. *Geometriae Dedicata*, 159(1):99–107, 2012.

[38] R. Norwood and G. Poole. An improved upper bound for leo moser's worm problem. *Discrete and Computational Geometry*, 29(3):409–417, 2003.

[39] Rick Norwood, George Poole, and Michael Laidacker. The worm problem of leo moser. *Discrete & Computational Geometry*, 7(2):153–162, 1992.

[40] Charles Stanley Ogilvy. Tomorrow's math: unsolved problems for the amateur. 1962.

[41] Julius Pál. *Ueber ein elementares variotionsproblem*, volume 3. AF Host, 1921.

[42] Chatchawan Panraksa and Wacharin Wichiramala. Wetzel's sector covers unit arcs. *arXiv preprint arXiv:1907.07351*, 2019.

[43] János D Pintér. Global optimization in action. *Scientific American*, 264:54–63, 1991.

[44] George Poole and Rick Norwood. Thirty-fifth anniversary of leo moser's worm problem.

[45] J. Schaer and J. Wetzel. Boxes for curves of constant length. *Israel Journal of Mathematics*, 12(3):257–265, 1972.

[46] Jonathan Schaer. *The broadest curve of length 1*. University of Calgary, Department of Mathematics, 1968.

[47] Houshang H Sohrab. *Basic real analysis*, volume 231. Springer, 2003.

[48] S. Som-am. *An improved lower bound of areas of convex covers for closed unit arcs.* PhD thesis, Master s thesis, Chulalongkorn University, 2010.

[49] R Sprague. Über ein elementares variationsproblem. *Matematisk tidsskrift. B*, pages 96–99, 1936.

[50] Sira Sriswasdi and Tirasan Khandhawit. An improved lower bound for moser's worm problem. *arXiv preprint math.MG/0701391*, 2006.

[51] Rangarajan K Sundaram et al. *A first course in optimization theory.* Cambridge university press, 1996.

[52] W. Wang. An improved upper bound for worm problem. *ACTA MATHEMATICA SINICA-CHINESE EDITION-*, 49(4):835, 2006.

[53] John W Ward. Exploring the bellman forest problem, 2008.

[54] John E Wetzel. Sectorial covers for curves of constant length. *Canad. Math. Bull*, 16(3):367–375, 1973.

[55] John E Wetzel. Fits and covers. *Mathematics magazine*, 76(5):349–363, 2003.

[56] W. Wichiramala. A smaller cover for closed unit curves. *Miskolc Mathematical Notes*, 19(1):691–698, 2018.

[57] Wacharin Wichiramala. A smaller cover for convex unit arcs. *East-West Journal of Mathematics*, 7(2), 2005.

[58] Scott W Williams. Million-buck problems. *The Mathematical Intelligencer*, 24(3):17–20, 2002.

[59] Viktor A Zalgaller. How to get out of the woods. *On a problem of Bellman (in Russian), Matematicheskoe Prosveshchenie*, 6:191–195, 1961.