# תרגיל 3.1

## אלגוריתם בסיס

### עקרונות האלגוריתם

נסתכל על הפוליהדרונים A,B המורכבים מאוסף משולשים. מרחק hausdorff מוגדר להיות max(d(A,B),d(B,A)) כאשר

בתיאור האלגוריתם נתייחס רק למרחק כאשר חישוב המרחק מתבצע באופן זהה.

נסמן ב- צמד הנקודות כך ש-. ניתן לצמצם את מרחב החיפוש של בעזרת האבחנה הבאה:

***למה 1****: קיימת נקודה אשר מקיימת את אחד התנאים הבאים:*

1. *הינו קודקוד של A ו- הינה הנקודה הקרובה ביותר על B*
2. *הינה נקודת על עקום החיתוך שבין משולש של A למשטח שווה המרחק מבין שני עצמים של B (קודקוד, צלע או פאה)*

מקרה א'

מקרה ב'

B

B

A

A

### תיאור האלגוריתם

*מקרה א' של הלמה מגדיר לנו אוסף של נקודות דיסקרטיות ב-A מהן יש לחשב את המרחק ל-B. חישוב המרחק מקודקודי a לכל אחד ממשולשי B הינה פעולה ב-O(1) ולכן מציאת מתוך אוסף הנקודות הינה בסיבוכיות כאשר n מספר הקודקודים ב-A ו-m מספר הקודקודים ב-B*

במקרה ב' ניתן לחשב את כל המשטחים שווי המרחק ולחתוך עם כל מהמשולשים לחישוב עקומים.

בכל אחד מהעקומים נמצא את הנקודה הרחוקה ביותר מהמשולש הכי קרוב ב-B. עבור העקום נסמן את ריבוע המרחק מהמשולש K ב- ונגדיר את פונקציית המעטפת התחתונה של המרחק מכל המשולשים . עפ"י הגדרות אלו מתקיים כי פונקציית ריבוע המרחק בין העקום ל-B הינה .

הביטוי ניתן לחישוב ב- בכל מקטע של המעטפת התחתונה ולכן סה"כ סיבוכיות החישוב בהינתן המעטפת התחתונה היא כגודל המעטפת התחתונה. סיבוכיות מעטפת תחתונה של פונקציות ממעלה חסומה s הינה האורך המקסיאמלי של סדרת davenport schinzel עם s+1 החלפות כלומר . יש לחשב גם את המעטפת התחתונה, חישוב מעטפת תחתונה הינו בסיבוכיות

בסיכום המקרים, סיבוכיות הריצה של אלגוריתם זה

**שיפור זמני ריצה**

ניתן לסנן את כמות המשלושים הנבדקים אחד מול השני, כאשר מחפשים את הנקודות השייכות למקרה ב' וכל נקודה אפשרית מורכבת ממרחק לקומבינציה של שלושה משולשים כל הפחות משולש נבדק הינה משמעותית.

**שיטה א'**

נשתמש בחסם הבא:

* מכל משולש ב-A נחשב את המרחק הקצר ביותר והארוך ביותר לכל משולש B.

נסמן בהתאמה ,

* לכל משולש ב-A נמצא את המרחק המקסימאלי ממשולש ב-B המינימאלי

,

* M הינו חסם עליון למרחק בין המשולש מ-A ל-B ולכן ניתן למחוק מרשימת צמדים המשולשים המועמדים להכיל את הנקודות המקיימות את המרחק המקסימאלי את כל הצמדים שעבורם

תחת הנחה שחישוב , עבור זוג משולשים הינן פעולות של (מרחק מקסימאלי מתקבל באחד מקודקודי A ומרחק מינמאלי ניתן לחסום מלמטה כיוון שהנגזרת של פונקציית המרחק חסומה בתחום המשולש) ולכן הסינון מתבצע ב-.

נמיין את אוסף המשולשים בסדר עולה של הפונקציה , נחפש את המרחק מהם ל-B בסדר זה ובכך נוכל לעדכן את החסם m לאורך האלגוריתם ולסנן זוגות משולשים נוספים.

**שיטה ב'**

*ניתן לסנן חיפוש לאורך bisector cuts באופן הבא:*

*נסמן ב-את המרחק של ה-bisector cut על שנוצר מחיתוך המשולש עם עקום שווה המרחק בין האיברים של מערך B מ- . אם אזי המרחק לא יתקיים לאורך העקום וניתן לא לחשב את המעטפת התחתונה לאורכו.*

*לאחר שחישבנו את מרחקי כל קודקודי מ-mesh B ואת הנקודות הרלוונטיות על כל ה-bisector cuts של משולשי B על משולשי A יש לנו חסם m הדוק יותר על המרחק*

# תרגיל 3.2

## תיאור הניסוי

הניסוי בדק את ביצועי השיטה הלוקאלית לפישוט פוליגונים Lindstrom-Turk עבור יחסים שונים של מספר משולשים בין המודל המקורי למפושט. נעשה הפרדה בין ביצועיו בבחירת הקשת האופטימאלית לבין ביצועים בבחירת מיקום הקודקוד ע"י השוואת כל אחת מהפונקציות בנפרד אל מול פונקציות בסיסיות יותר. פונקציית בחירת הקשת הושוותה אל מול בחירת הקשת הקצרה ביותר ופונקציית בחירת מיקום הקודקוד הושוואה אל מול מיקום הקודקוד במרכז הקשת המקורית.

## מקרי הבוחן

תוצאות האלגוריתם נבחנו על חמישה משטחים

1. Dragon

קובץ בגודל: 72 ק"ב ,גאומטריה: concave, מספר צמתים:1257 , מספר משולשים: 2730.

1. icosahedron\_5

קובץ בגודל: 1.3 מ"ב, גאומטריה: convex, מספר צמתים: 10242, מספר משולשים 20480

1. feline

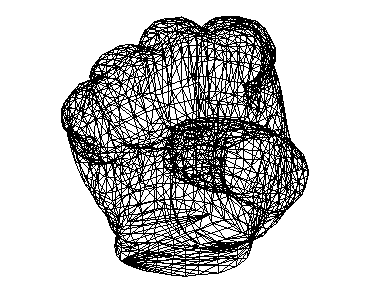
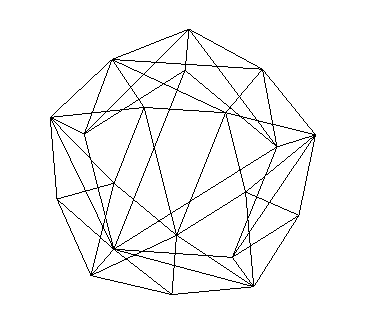
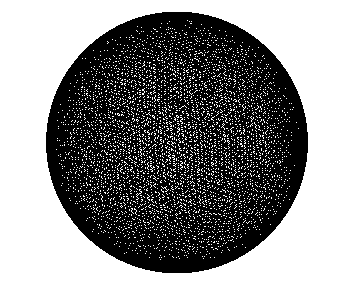
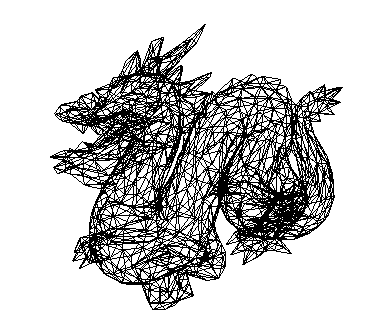
גודל קובץ: 4 מ"ב, גאומטריה: concave, מספר צמתים: 49864, מספר משולשים: 99732

1. dodachedron

גודל קובץ: 869 בייטים, גאומטריה: convex, מספר צמתים: 20, מספר משולשים: 36

1. hand

גודל קובץ, 71.2 ק"ב, גאומטריה: convex, מספר צמתים: 1197, מספר משולשים: 2390.



**(A)**

**(B)**

**(C)**

**(D)**

**(E)**

## ניסוי #1: מדידת ביצוע Lindstrom-Turk

בניסוי זה נעשה שימוש באלגוריתם Lindstrom-Turk המלא כלומר הן בבחירת קשת והן במיקום הקודקוד החדש. הביצועים נבחנו עבור מספר יחסים בין כמות המשולשים ההתחלתית לסופית כתנאי עצירה.  
נשווה מרחק האוסדורף בין חמשת המשטחים שנבחרו כדי לבחון את יעילות האלגוריתם בתנאים שונים.

מרחק האוסדורף

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ratio/model | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| A | 20.1481 | 5.84145 | 3.62342 | 2.7009 | 1.6869 |
| B | 0.030939 | 0.00131355 | 0.000735973 | 0.000599224 | 0.000528822 |
| C | 0.00920327 | 0.00319365 | 0.00319365 | 0.00187766 | 0.000970343 |
| D | 2.49304 | 0.585571 | 0.421382 | 0.192349 | 0.160223 |
| E | 0.030939 | 0.0142608 | 0.0100055 | 0.00761657 | 0.00295408 |

### תוצאות ומסקנות

1. בשניים מתוך שלושת המודלים הקעורים ( feline ו-hand), מתקבל שמרחק האוסדורף קטן יותר עבור אותו תנאי עצירהן, יחסית למודל הקעור השלישי (dragon). הסיבה לכך היא הגאומטריה של מודל ה-dragon, שמצד אחד מכיל מספר קטן יחסית של קשתות (יחסית ל-feline) ומצד שני הגאומטריה שקשתות אלו מתארות היא מסובכת (יחסית ל-hand, שהוא גם מודל קעור שמכיל בערך אותו מספר של קשתות).

2. במודל הקמור הגדול (icosahedron\_5), מרחק האוסדורף המתקבל באותם תנאי עצירה הוא קטן משמעותית ממרחק האוסדורף המתקבל עבור המודל הקמור הפשוט (dodecahedron). סביר להניח שעקב המורכבות הגדולה יותר של icosahedron\_5, המשטח המפושט שמור על גאומטריה קרובה גם תחת קריסה של מספר קשתות גדול יותר, יחסית ל-dodecahedron, שמלכתחילה מכיל מספר קטן של קשתות, ולכן מכיל פחות "יתירות".

## ניסוי #2: השוואת ביצועי בחירת קשת

בניסוי זה נערכה השוואה בין ביצועי השימוש באופן בחירת הקשת של Lindstrom-Turk שחושבה בניסוי 1# לבין פונקציית המחיר של squared length אשר בוחרת את הקשת הקצרה ביותר.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| 1 | 20.0278 | 9.8518 | 9.8518 | 3.26897 | 2.48898 |
| 2 | 0.00381135 | 0.00147775 | 0.000877384 | 0.00750223 | 0.000727705 |
| 3 | 0.0120184 | 0.00488645 | 0.00380057 | 0.00269243 | 0.00269243 |
| 4 | 2.21868 | 0.585323 | 0.278619 | 0.214919 | 0.160223 |
| 5 | 0.0735152 | 0.020183 | 0.0134633 | 0.0128823 | 0.0131091 |

### מסקנות:

1. ניתן לראות כי אכן LT נותן ביצועים טובים יותר מהשיטה הנאיבית של squared length
2. לאורך הניסוי נמדדה מהירות חישוב גבוהה יותר (פחות מורכב מבחינה חישובית)
3. ככל שתנאי העצירה יורד (כלומר הרישות המפושט גס יותר) נראה שההבדלים בין אלגוריתם Lindstrom-Turk למחיר squared length מצטמצמים. מסקנה: ניתן לחסוך את זמני החישוב הגבוהים של אלגוריתם Lindstrom-Turk בתנאי עצירה נמוכים ולהשתמש בתנאי הנ"ל. המחיר הוא מרחק האוסדורף גדול יותר.

## ניסוי #3: השוואת תהליך מיקום הקודקוד

שימוש בפוליסת מיקום midpoint, כלומר, מיקום צומת באמצע קשת שמוקרסת והשוואת מרחק האוסדורף לזה שהתקבל בניסוי #1.

מרחק הואסדורף:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| 1 | 21.8983 | 13.2153 | 5.11969 | 2.86957 | 2.12244 |
| 2 | 0.00573522 | 0.00202366 | 0.00105856 | 0.000801658 | 0.000582009 |
| 3 | 0.102094 | 0.00777287 | 0.00769361 | 0.00316001 | 0.00198859 |
| 4 | 0.837999 | 0.660869 | 0.454365 | 0.230334 | 0.200811 |
| 5 | 0.0717937 | 0.0256731 | 0.0139635 | 0.00955728 | 0.00545891 |

### מסקנות

1. מרחקי האוסדורף גרועים יותר עבור אותו יחס עצירה.

2. מהירות חישוב גבוהה יותר (פחות מורכב מבחינה חישובית)

3. ככל שתנאי העצירה יורד (כלומר הרישות המפושט גס יותר) נראה שההבדלים בין מיקום הצמתים באלגוריתםLindstrom-Turk למיקום באמצע הקשת המוקרסת, מצטמצמים. מסקנה: ניתן לחסוך את זמני החישוב הגבוהים של אלגוריתםLindstrom-Turk) בתנאי עצירה נמוכים ולהשתמש בתנאי הנ"ל. המחיר הוא מרחק האוסדורף גדול יותר.

4. עפ"י התוצאות המתקבלות, עבור אותו תנאי עצירה, שימוש בפוליסת המיקום הנ"ל, נתונת תוצאות פחות טובות משימוש בפוליסת המחיר של ניסוי #2. בריצות שהתבצעו, לא הורגשו הבדלים משמעותיים בזמני הריצה בין ניסוי #2 לניסוי #3.