# קידוד ואלגוריתמים לזיכרונות 236379

## עינב הוברמן, בנימין וורניק ואלישבע רבינוביץ'

**הצגת הבעיה**

בהינתן מילים מאורך ו- מילים מאורך , יש למצוא אלגוריתם יעיל המחשב את הקב': .

כלומר, מציאת אלגוריתם המחשב את החיתוך של מס' כדורי הכנסה ומחיקה.

זהוי הרחבה של עבודה הנעשתה במאמר [The Intersection of Insertion and Deletion Balls](ieeexplore.ieee.org/document/9611515)

**הקדמה**

במאמר מוצג אלגוריתם יעיל עבור המקרה של . נקרא לאלגוריתם זה ID(x,y,n) כאשר x מאורך n-t ו-y מאורך n+t.

ראשית האלגוריתם מחשב את רשימת האינדקסים הימניים-ביותר במילה y אשר תת-המחרוזת של y הנוצרת מאינדקסים אלו נותנת את המילה x. חישוב זה הינו רקורסיבי ומצריך טבלת תכנון דינמי לחישוב ה-lcs של המילים x ו-y וטבלת התאמה של המילים (טבלה בינארית אשר יש בכניסה ה-ij את הערך 1 אם ורק אם ). לאחר מכן האלגוריתם מחשב עבור כל סט של אינדקסים אילו אינדקסים יש להוסיף מ-y כדי לקבל מחרוזת מאורך n.

**סיבוכיות:**

לפי המאמר, הוא חסם על זמן הריצה הריצה הממוצע של האלגוריתם.

נסמן ב- את הסיבוכיות של האלגוריתם. במקרים מסוימים ההקשר יהיה סיבוכיות ממוצעת ובהקשר אחרים סיבוכיות במקרה הגרוע (ההקשר יהיה מובן).

**אופן יצירת קלטים לטסטים**

בכל טסט הגרלנו ווקטור באורך n ויצרנו ממנו באופן רדנומלי את קבוצות המילים הארוכות והקצרות על מנת להבטיח שהחיתוך לא יהיה ריק. עשינו זאת על ידי הגרלת t אינדקסים להכנסת t ערכים אקראיים שערכם 0/1 על מנת ליצור את המילים הארוכות ועל ידי הגרלת t אינדקסים למחיקת ערכים ליצירת המילים הקצרות.

**קישור לעבודה ב-GIT**

<https://github.com/einavhu/intersection-of-multiple-insertion-and-deletion-balls.git>

**אלגוריתם 1 – פתרון נאיבי**

**פסאודו קוד**

קלט: ווקטורים כך שלכל מתקיים *ולכל מתקיים וגם .*

*פלט: .*

*האלגוריתם:*

1. *לכל נחשב:.*
2. *החזר .*

**הסבר**

בהינתן וקטור supersequences מאורך k המכיל מילים מאורך n+t ווקטור subsequences מאורך k המכיל מילים מאורך n-t, נחלק את המילים ל-k זוגות זרים כך שכל זוג מכיל מילה מווקטור ה-supersequences ומילה מווקטור ה-subsequences.

לאחר שקיבלנו k זוגות, נמצא את לכל באמצעות אלגוריתם ID.

לאחר שקיבלנו k קבוצות, נחזיר את החיתוך שלהן.

מציאת החיתוך של הקבוצות נעשה באמצעות מעבר יחיד על כל הקבוצות וספירת מספר המופעים של כל מחרוזת על פני כל הקבוצות. מחרוזת שמופיעה כמספר קבוצות הID תימצא בחיתוך. באופן זה מציאת החיתוך נעשית בזמן ליניארי בסכום גדלי הקבוצות

הערה: אנו מניחים כי שני וקטורי הקלט שלנו מסודרים בצורה אקראית ולכן באיטרציה ה-i ניקח את המילה ה-i מווקטור ה-supersequence ואת המילה ה-i מווקטור ה-subsequence, בהנחה שזה שקול ללקיחת k זוגות באופן אקראי.

אלגוריתם זה הוא הפתרון הנאיבי לבעיה, שנשתמש בו בתור baseline כדי להעריך את הביצועים של האלגוריתמים המופיעים בהמשך.

**ניתוח סיבוכיות**

1. מציאת ID של כל זוג: . נבצע זאת עבור k זוגות כך שסה"כ נקבל .
2. מציאת החיתוך של הIDים: ליניארי בסכום גדלי הIDים. במקרה הגרוע נקבל ID מסדר גודל של , כפי שראינו במאמר המדובר, ולכן סכום הגדלים הוא לכל היותר .
3. סה"כ: .

אם הוא התוחלת של אז בתוחלת סיבוכיות האלגוריתם תהיה כלומר עבור קבועים וקבוצת מחרוזות אקראיים, האלגוריתם צפוי להיות לינארי ב-.

**תוצאות**

A screenshot of a computer

Description automatically generated with low confidenceזמן ריצה ממוצע של אלגוריתם 1 כאשר מקבעים את ו-

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

Chart, line chart

Description automatically generatedזמן ריצה ממוצע של אלגוריתם 1 כאשר מקבעים את ו-

Chart, line chart, histogram

Description automatically generated

Chart

Description automatically generatedזמן ריצה ממוצע של אלגוריתם 1 כאשר מקבעים את ו-

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

*מסט הגרפים הראשון (כאשר ציר האיקס הוא ) ניתן לראות כי כאשר מקבעים את הגידול בזמן הריצה הוא לינארי כפי שצפינו. עם זאת, ניתן לראות בסט הגרפים השני (כאשר ציר האיקס הוא ) שההתנהגות של זמן הריצה הממוצע הוא אקספוננציאלי ב-, דבר אשר מסתדר עם כך שכאשר מקבעים את ו-, זמן הריצה בממוצע תלוי בגודל של . מוקדם יותר הסברנו כי במקרה הגרוע גודל החיתוך של זוג וקטורים הוא וחסם על המקרה הממוצע הוא , מה שמתיישב עם התוצאות.*

*מסט הגרפים האחרון (כאשר ציר האיקס הוא ) רואים הבדל בהתנהגות עבור -ים גדולים וקטנים. כאשר גדול, זמן הריצה קטן ככל ש- גדל. דבר זה נוגד את האינטואיציה הראשונית שלנו, אך נותן פתח למחקר המשך על ההתנהגות של הגודל הממוצע של החיתוך כפונקציה של ו-. אפשר להסביר זאת באמצעות התבוננות על הביטוי אשר הינו חסם עליון על גודל החיתוך. ביטוי זה הוא וכן . כלומר יש יחסי כוחות מנוגדים בין ו- באקספוננט שיכול להסביר את ההבדל שאנו רואים.*

**אלגוריתם 2**

**פסאודו קוד**

קלט: ווקטורים כך שלכל מתקיים *ולכל מתקיים וגם .*

*פלט: .*

האלגוריתם:

1. בחר.י ע"פ כלל החלטה כלשהו.
2. חשב.י את .
3. אתחל .
4. לכל :
   1. לכל : אם אינו supersequence של עבור לאיטרציה הבאה בשלב 4.
   2. לכל : אם אינו subsequence של עבור לאיטרציה הבאה בשלב 4.
   3. .
5. החזר.י את .

**הסבר**

בהינתן וקטורים מאורך ו- וקטורים מאורך , ניקח מתוכם זוג וקטורים באופן מסוים (ע"פ הגרסה שתפורט בהמשך) כאשר מאורך ו- מאורך נמצא את   
 באמצעות אלגוריתם ID.

לאחר מכן, נעבור על כל הווקטורים ב- ונבדוק עבור כל אחד מהם האם הוא נמצא ב-deletion ball של כל אחד מהווקטורים מאורך בקלט וגם נמצא ב-insertion ball של כל אחד מהווקטורים מאורך בקלט. אם כן, נשאיר אותו. אם לא, נוציא אותו מ-. נסמן את הקבוצה שהתקבלה בסוף ב-

הערה: עבור כל מילה ב- הבדיקה מתבצעת ב-. זאת מכיוון שניתן ב- לבדוק אם מילה אחת היא subsequence של מילה שניה וזה שקול לבדיקה האם המילה היא ב-deletion/insertion ball של מילה אחרת.

סה"כ נקבל שבסוף ריצת האלגוריתם מתקיים: ולכן נחזיר אותה.

**נכונות**

נראה כי בסוף ריצת האלגוריתם מתקיים ע"י הכלה דו כיוונית:

וגם לכל ו- *מתקיים הוא תת מילה של ו- הוא תת מילה של   
 וגם לכל ו- מתקיים כי וגם   
.*

**ניתוח סיבוכיות זמן**

נסמן ב- את זמן הריצה של כלל ההחלטה.

*.*

**היוריסטיקות לבחירת הזוג**

מניתוח סיבוכיות האלגוריתם ברור כי נרצה להקטין את כמה שניתן. על כן, נציג מספר היוריסטיקות שבחרנו שמטרתן לאפשר ניחוש של "זוג מוצלח" שהחיתוך שלו קטן. בהמשך נשווה בין ההיוריסטיקות מבחינת סיבוכיות (זמן ריצה) וביצועים (עד כמה החיתוך של הזוג שנבחר קטן).

סימונים:

* : אורך ה-alternating sequence המקסימלי ב-.
* : מספר ה-runs ב-.
* : קב' המילים הקצרות ().
* : קב' המילים הארוכות ().
* : וקטור ה- של . כלומר אם האורך של ה- ה- ב- הוא אז .

היוריסטיקות:

1. .
2. .
3. .
4. arg   
   הערה על 4: אם ו- אינם באותו אורך, נרפד את באפסים בסוף, ובנוסף, אם ו- לא מתחילים באותה אות נוסיף ל- 0 בתחילתו כדי להשוות את האורכים.

היוריסטיקה 1 (הסבר):

ההיגיון מאחורי היוריסטיקה זו הוא שכל זוג שהיא פולטת מחיל מגבלה על אופי המחרוזות בחיתוך: כל המחרוזות בחיתוך יצטרכו לקיים שמספר הruns בהם קטן או שווה למספר הruns במחרוזת הארוכה שנבחרה וגדול או שווה למספר הruns במחרוזת הקצרה שנבחרה. נסביר מדוע.

כאשר מבצעים deletion במילה, מס' ה-runs במילה לאחר המחיקות לא יכול לגדול. באופן דומה, כאשר מבצעים insertion במילה, מס' ה-runs במילה לאחר ההכנסות לא יכול לקטון.  
 . יתר על כן, כל מקיים: .  
לכן, נראה לנו שיש סבירות כי היוריסטיקה זו תפלוט בממוצע זוג בעל חיתוך קטן.

היוריסטיקות 2+3 (הסבר):

במאמר מראים כי עבור *שתי מילים ו- , מתקיים כי  
 תלוי באורכי ה* alternating sequences*המקסימליים במילים.*

*מכאן שעבור* t = 1, *מתוך הסתכלות על אורך ה* alternating sequence*המקסימלי במחרוזות, נוכל לשערך את גודל החיתוך שלהם.*

*ערכנו מספר נסיונות כדי לראות האם ואיך ניתן להרחיב קשר זה למקרה כללי ().*

*בחרנו להסתכל גם על ההפרש המינימלי וגם על ההפרש המקסימלי כדי לנסות להבין את ההתנהגות של גדלי החיתוכים.*

*היוריסטיקה 4 (הסבר):*

*נסתכל על ו- של מילים באורך ו- באורך אשר ה- שלהם אינו ריק. אם ו- באותו האורך, אזי בהכרח כל מילה ב- גם כן בעלת אותו אורך של וקטור , ויתר על כן סדר ה--ים של האפסים והאחדים הוא אותו הסדר. ככל שיש יותר כניסות בהן , יהיו פחות דרכים שבהם ניתן להוסיף תווים ל ולהחסיר תווים מ ולקבל את אותה המילה. כניסות שבהן ההפרש בין קטן גם מצמצמות את מספר הדרכים האלה. לכן, אינטואיטיבית, הדמיון בין קשור לגודל הID שלהם.*

*כדי למצוא זוג מילים כך שה שלהן דומה, השתמשנו במרחק אוקלידי. מדובר בפרוקסי קל לחישוב, אך גם לא מדוייק, היות ומרחק אוקלידי קטן יותר לאו דווקא מתקבל עבור וקטורים עם מספר הכניסות הזהה הגבוה ביותר, שאנחנו יודעים שהוא פרמטר שחשוב עבור מדד הדמיון אליו אנו חותרים.*

*מכיוון שלא ניתן להבטיח ש- ו- יהיו מאותו אורך, היה צורך בריפוד של . החלטנו שרירותית לדחוף את הריפוד בסוף מכיוון שאין לנו דרך לדעת בצורה יעילה מאיפה נעלמו ה--ים.   
נשים לב כי זמן הריצה של היוריסטיקה זו ריבועי ב-, לעומת היוריסטיקה 1 שהוא לינארי ב-. לכן, להיוריסטיקה זו יהיה ערך רק אם גודל הקבוצה המתקבלת יהיה משמעותית קטן יותר מאשר גודל הקבוצה המתקבלת מהיוריסטיקה 1.  
בעיקרון, ניתן להשתמש בהיוריסטיקה זו גם כשובר שוויון בהיוריסטיקה 1 במידה ויש יותר מאפשרות אחת, אך כפי שנראה בהמשך אין בכך צורך שכן זמני הריצה לפי היוריסטיקה 1 הם נמוכים ביותר.*

**ניתוח תוצאות ההיוריסטיקות והשוואתן**

ניזכר כי השאיפה שלנו היא למצוא היוריסטיקה שמוצאת לנו את זוג הווקטורים שגודל ה-ID שלהם מינימלי (בכדי להקטין את זמן ריצת אלגוריתם 2, שתלוי בגודל הזה).

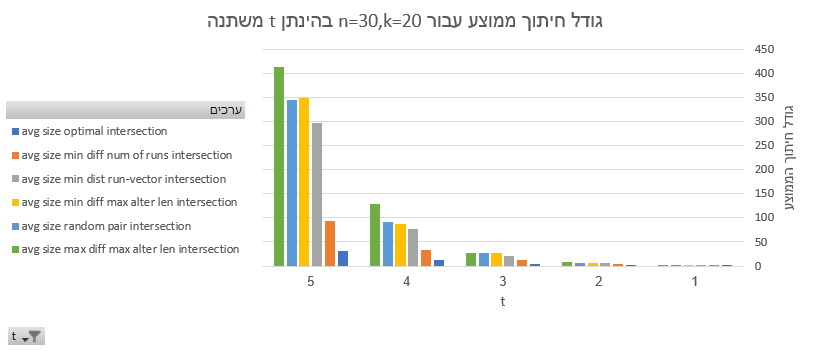
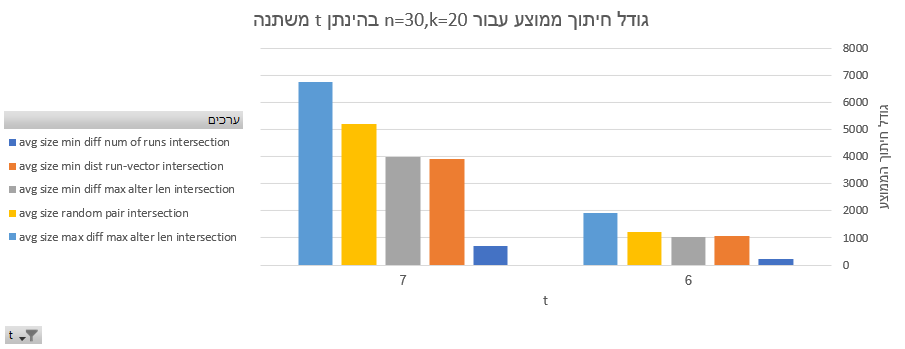
הרצנו את ההיוריסטיקות על קלטים אקראיים זהים ובדקנו את גודל החיתוך שהפלט של כל היוריסטיקה סיפק. ערכנו את הבדיקה על צירופים שונים של הפרמטרים כפי שיפורט להלן. נציג תוצאות המשוות בין גדלי החיתוך הממוצעים שהתקבלו מההיוריסטיקות, ננתח אותן ונסיק מהם מסקנות.

כדי שנוכל להעריך את טיב התוצאות, הצגנו בנוסף לפלט של ההיוריסטיקות גם שני baselines:

1. גודל קבוצת החיתוך המינימלית שיכולה להתקבל מבחירת זוג כלשהו (אותו מצאנו בעזרת חיפוש ממצה פשוט). תוצאה זו מוצגת בצבע כחול כהה בגרפים. השוואת יתר התוצאות למדד זה תראה לנו עד כמה אנחנו רחוקים מהאופטימום.
2. גודל קבוצת החיתוך המתקבלת מבחירה אקראית של מחרוזת קצרה ומחרוזת ארוכה באופן ב"ת זו בזו. בעזרתה נוכל לקבוע את הערך של ההיוריסטיקות: להיוריסטיקה יהיה ערך רק אם תפלוט קבוצה שהגודל שלה קטן מאשר הפלט של random.
3. **בינוניים**

בגלל ההפרשים בגודל החיתוך המתקבל עבור ערכי שונים, התוצאות מוצגות בנפרד עבור , בגרפים בעלות סקלות שונות. בנוסף לגרפים מובאת טבלה המרכזת את הערכים המופיעים בגרפים.

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

מתוצאות אלה עולה באופן ברור כי היוריסטיקה 2 (כתום) נותנת את הביצועים הטובים ביותר, ואף גדֵלה בt בצורה מתונה הרבה יותר מיתר ההיוריסטיקות. למשל, הגדלת t מ1 ל5 הגדילה את גודל החיתוך הממוצע המתקבל מהיוריסטיקה 1 בערך פי 45, בעוד שהחיתוך המתקבל מההיוריסטיקה השנייה בטיבה (4) גדל כמעט פי 145.

ניתן לראות שהיוריסטיקות 2 ו3 נותנות ביצועים דומים לבחירה האקראית עבור , ועבור ערכי גדולים יותר היוריסטיקה 3 טובה מעט מrandom בעוד שהיוריסטיקה 2 מעט פחות טובה. התוצאות של שתי היוריסטיקות אלה מובילות לאותה מסקנה, שכדי להקטין את החיתוך, נרצה לקחת מחרוזות בעלות max alternating sequence באורך דומה.

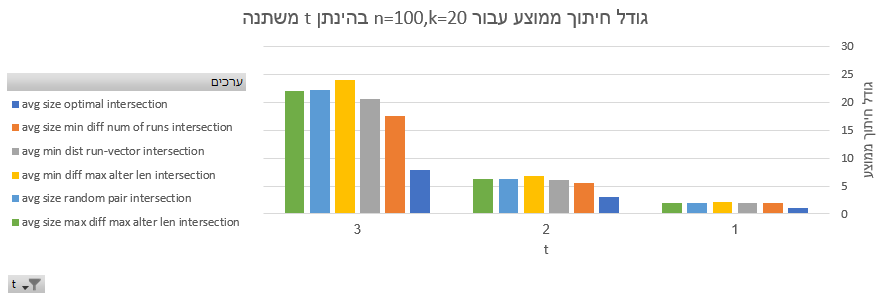
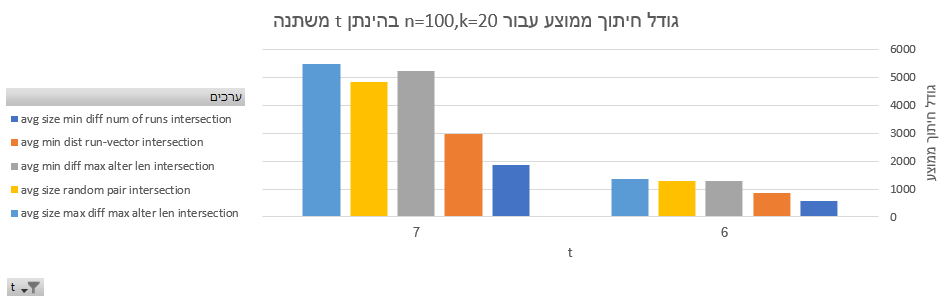
היות ושתי היוריסטיקות אלה נתנו תוצאות שונות רק במעט מהיוריסטיקת הבחירה האקראית, עבור כל הtים שבדקנו, נסיק שבמתכונת הנוכחית שלהם התרומה שלהם לאלגוריתם קטנה עד זניחה.

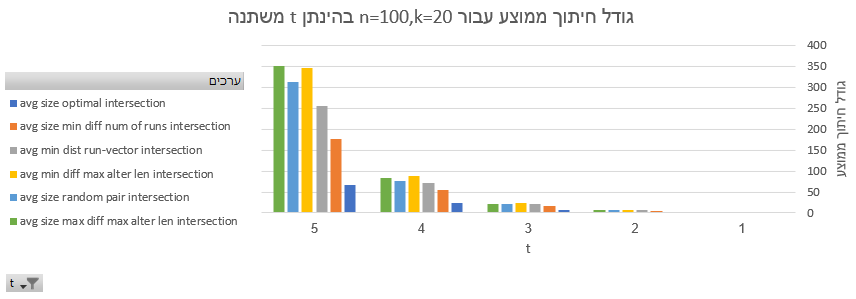
היוריסטיקה 4 משמעותית טובה יותר מהיוריסטיקות 2,3 עבור . עם זאת, היוריסטיקה 1 טובה ממנה בפער שהולך וגדל עם . בגלל הדמיון בין הגדרת היוריסטיקות אלה, ניחוש מושכל מורה ששימוש בהיוריסטיקה 1 מייתר את השימוש בהיוריסטיקה 4 כך ששילוב שלהם לא יהווה שיפור משמעותי.

1. **ההשפעה של הגדלת**

כדי לבחון איך הגדלת משפיעה על הביצועים של ההיוריסטיקות השונות, נבדוק את הביצועים שלהם על גדול פי 3.333 מזה שהשתמשנו בו, ונתבונן במגמת השינוי.

בגלל ההפרשים בגודל החיתוך המתקבל עבור ערכי שונים, התוצאות מוצגות בנפרד עבור , בגרפים בעלות סקלות שונות. האלגוריתם האופטימלי לא הורץ על היות וזמן הריצה היה ארוך מדי.

בנוסף לגרפים מובאת טבלה המרכזת את הערכים המופיעים בגרפים.



תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

נסתכל על תוצאות אלו בהשוואה לתוצאות שהתקבלו עבור .

ראשית, הפער בין היוריסטיקה 1 ויתר ההיוריסטיקות קטֵן – למשל, בעוד שעבור הגדלת הגדילה את החיתוך שהתקבל מהיוריסטיקה 1 יותר מפי 2, החיתוך שהתקבל מהיוריסטיקות 3,4 *קטן* עבור רוב ערכי *.* אך, נשים לב כי גם החיתוך המתקבל מהבחירה האקראית קטן, כך שלכאורה פשוט מדובר בפרמטרים "טובים" יותר, שמסיבה כלשהי פוגעים ביכולות של היוריסטיקה 1. יתר על כן, כעת גם היוריסטיקה 3 פחות טובה מבחירה אקראית החל מ, כך שלא באמת ניתן לראות את הקטנת החיתוך כשיפור בביצועים, היות וגם הbaseline השתנה.   
לא ניכר שהשינוי בn השפיע על היוריסטיקה 2.

סה"כ, הדירוג היחסי בין ההיוריסטיקות נשמר – הגדלת לא גרמה לאף היוריסטיקה "לעקוף" את אחת ההיוריסטיקות האחרות. לכן נוכל שוב להסיק שהיוריסטיקה 1 היא הטובה ביותר, אך רואים שבעוד שהבחירה האקראית המהווה baseline נתנה חיתוכים קטנים יותר כשהגדלנו את (וכן יתר ההיוריסטיקות), היוריסטיקה 1 נתנה חיתוכים גדולים יותר, כך שהפער ביניהם הצטמצם. כדי להמשיך לבדוק את המגמה, שוב השארנו את כמו שהוא והגדלנו מאוד את ((. בטבלה הבאה מוצגות התוצאות:

*תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי*:

התוצאות מתאימות למגמה שראינו, ומחדדות את מה שקורה כאן: נראה שכאשר מאוד גדול **כל** ההיוריסטיקות נותנות תוצאות מאוד דומות. בשלב הזה עוד יש יתרון קטן להיוריסטיקה 1, אך הוא הצטמצם מאוד. נוכל לנחש שכאשר ההיוריסטיקות כולן יתנהגו באופן זהה לבחירה רנדומית, ולכן במקרים בהם מאוד גדול עדיף לוותר על ההיוריסטיקות שלנו ופשוט לבחור זוג אקראי.

1. **ההשפעה של הגדלת**

כדי לבחון את ההשפעה של הגדלת , נשווה בין התוצאות עבור   
ועבור .

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי:

:

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

אנו מבחינים שבדומה למקרה של גדול, גם בגבול היוריסטיקה 2 שואפת לאלגוריתם הבחירה הרנדומי. לעומת זאת, היוריסטיקות 1,4 שומרת על היתרון היחסי שלהן והגדלת לא משפיעה עליהן כלל! כמו כן נראה שהגדלת משפיעה על היוריסטיקה 3 לרעה.

**סיכום התוצאות:**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| היוריסטיקה 1 טובה משמעותית מבחירה אקראית ויתר ההיוריסטיקות (עבור כל הים). | כל ההיוריסטיקות שואפות לתוצאה של בחירה אקראית. | אין השפעה על בחירה אקראית והיוריסטיקות 1,4  היוריסטיקה 2 שואפת לתוצאה של בחירה אקראית |
| **דירוג לפי ביצועים (לכל ערכי (**: | | |

בהמשך, בעקבות המסקנות מחלק זה, התמקדנו בזמני הריצה של אלגוריתם 2 המתבצעת באמצעות היוריסטיקה 1, היות ומדובר בהיוריסטיקה הטובה ביותר שמצאנו.

**ניתוח תוצאות עבור אלגוריתם 2 עם היוריסטיקה 1 (הפרש מינימלי בין מספר ה-)**

Chart, line chart

Description automatically generatedזמן ריצה ממוצע של כאשר מקבעים את ו-

Chart, line chart

Description automatically generatedA picture containing text, sky, different

Description automatically generated

Chart, line chart

Description automatically generatedזמן ריצה ממוצע של כאשר מקבעים את ו-

Chart, line chart, histogram

Description automatically generated

Chart, line chart

Description automatically generatedזמן ריצה ממוצע של כאשר מקבעים את ו-

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated

*מסט הגרפים הראשון (כאשר ציר האיקס הוא k ומקבעים את t,n) ניתן לראות שיפור בביצועים ככל ש-k גדל. מכיוון שההיוריסטיקה היא בחירת ההפרש המינימלי של מספר ה-, ככל שיש יותר מילים הזוג הנבחר יהיה בעל הפרש קטן יותר, מה שיקטין את הגודל של . ניתן לראות זאת גם בסט הגרפים השני (כאשר ציר האיקס הוא ) שעבור -ים גדולים יותר זמן הריצה אינו גדל אקספוננציאלית. למעשה, יש כאן טריידאוף בין ו-. כאשר מגדילים את , גודל החיתוך הממוצע של זוג וקטורים גדל, אך אם מסתכלים על גודל החיתוך הממוצע הנבחר* ***לפי ההיוריסטיקה הנ"ל*** *אז משחק תפקיד מאוד משמעותי וניתן לראות שאף משמעותי יותר מ- עצמו. לבסוף, מסט הגרפים האחרון ניתן לראות כי כאשר מגדילים את זמן הריצה גדל גם כן, וכאשר גדל זמן הריצה ב- גדל לינארית (עבור -ים קטנים ההתנהגות של זמן הריצה רועשת יותר). נשים לב כי הגודל של עדין משחק תפקיד גדול יותר מ- בזמן הריצה. ניתן לראות זאת בכך שנאלצנו להציג את הגרפים עבור בנפרד מאשר עבור שכן זמני הריצה הם בסדרי גודל שונים.*

***השוואת תוצאות של אלגוריתם 2 עם היוריסטיקה 1 ועם בחירת זוג רדנומלי***

*גרפים נבחרים עבור אלגוריתם 2 עם בחירה של זוג רנדומלי:*

*זמן ריצה ממוצע כאשר מקבעים את :*

Chart, line chart, histogram

Description automatically generated

Chart, line chart

Description automatically generated

נשים לב כי כאשר הזוג הנבחר הוא רנדומלי, כבר אין שיפורים משמעותיים בזמני הריצה עבור -ים גדולים יותר, ואנו חוזים שוב זמן ריצה שגדל אקספוננציאלית כאשר מגדילים את . זה כמובן מסתדר עם כל מה שהסברנו על גודל החיתוך של זוג וקטורים רנדומלי. יתר על כן, מכיוון שכל הקלט רנדומלי, כבר אינו משחק תפקיד משמעותי במקרה הממוצע. אנו כן עדים שוב לתופעה שזמני הריצה קטנים כאשר גדל בדומה לאלגוריתם 1.

**השוואת זמני ריצה של כל האלגוריתמים עבור (עבור יתר המקרים התוצאות דומות):**

הערה: בחרנו שלא להציג את זמני הריצה של אלגוריתם 2 עם יתר ההיוריסטיקות שכן ראינו כי היוריסטיקה 1 בעלת התוצאות הטובות ביותר, וכן מלבד בחירת זוג רנדומלי היא גם בעלת סיבוכיות זמן הריצה הנמוך ביותר.

זמני ריצה ממוצעים עבור קבוע ו- משתנה:

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generated*זמני ריצה ממוצעים עבור קבוע ו- משתנה:*

*בבירור ניתן לראות כי ביצוע אלגוריתם 2 הם משמעותית טובים יותר מאשר של אלגוריתם 1, גם כאשר הזוג הנבחר באלגוריתם 2 הוא רנדומלי כפי שצפינו.*

***סיכום:***

*מצאנו אלגוריתם הנותן את קבוצת החיתוכים של הinsertion & deletion balls בזמן יעיל. הביצועים של האלגוריתם שלנו טובים באופן משמעותי מהאלגוריתם הנאיבי, ועומדים על פחות מעשירית השנייה במקרה הממוצע, ופחות מחצי שניה עבור tים גדולים (שבמקרה זה הפלט עצמו מאוד גדול).*

*למרות הקשר הישיר בין אורך הmax alternating sequence לגודל הID עבור t = 1, מצאנו כי הקשר עבור tים גדולים יותר הוא חלש, כך שתכונה זו לא מהווה אינדיקציה מספקת בפני עצמה לגודל הID.*

***כיווני המשך:***

* *שילוב היוריסטיקות: לא בחנו את האפשרות להשתמש ביותר מהיוריסטיקה אחת לשם בחירת הזוג באלגוריתם 2. יכול להיות ששימוש באחת היוריסטיקות החלשות יותר בתור שובר שיוויון, או נתינת ציון ממושקל המבוסס על כלל ההיוריסטיקות, יתן תוצאות טובות יותר. נציין כי הרצת עוד היוריסטיקות תגדיל את זמן הריצה, ולכן גם אם נמצא ששילוב כלשהו שלהם מוצא ID קטן יותר בממוצע, אין זה הכרחי שבסופו של דבר זמן הריצה יהיה טוב יותר.*
* *בתוצאות שלעיל ראינו שבמקרים מסויימים, הגדלת n בלי להגדיל את t מקטינה את זמן הריצה. נראה שיש כאן קשר מורכב בין גודלו של n לזמן הריצה, ויכול להיות מעניין לבחון קשר זה עוד.*
* *כפי שהסברנו, הגדרת היוריסטיקה 4 הגיעה מתוך ניסיון למדוד דמיון בין וקטורי runs. מרחק אוקלידי במקרה הזה לא עונה בצורה מיטבית על הצרכים שלנו, בין השאר כי אורכי הוקטורים יכולים להיות שונים. ניתן לחשוב על מטריקות אחרות שיביעו בצורה טובה יותר את האינפורמציה הגלומה בהשוואת וקטורי runs.*
* *היות ועבור tים גדולים (גדולים אפילו מ5) גודל החיתוך הסופי גדול מאוד, לא התייחסנו לתחום ערכים זה בתוצאות שלנו, היות וזמן הריצה היה ארוך מדי להרצת טסטים. בהינתן המשאבים הנדרשים להרצה בסדר גודל כזה, יהיה מעניין לבדוק מעין שילוב של אלגוריתם 1 ו2 במקרה בו t גדול: במקום למצוא את הID של זוג אחד בלבד כמו באלגוריתם 2, נמשיך למצוא IDים של זוגות נוספים ולחתוך ביניהם, כל עוד גודל הקבוצה שבידנו גדולה מסף מסויים. כאשר גודל הקבוצה קטן מספיק, נעבור לבדיקת הוקטורים בקבוצה ביחס לsupersequence/subsequences הנותרים, כפי שעשינו בשלב השני של אלגוריתם 2.*
* *אפשרויות להכללת הבעיה:*
  + *הרחבה למקרה בו מספר הinsertions ומספר הdeletions אינו שווה*
  + *הרחבה למקרה בו מספר הsupersequences שונה ממספר הsubsequences.*
  + *הרחבה למקרה בו ישנה הסתברות קטנה עבור כל אחת מהמילים הנתונות שהיא* ***אינה*** *supersequence/subsequence של המילה המקורית. כלומר שנפלו שגיאות בחלק ממילות הקלט.*