1. הביטוי המתמטי המתאים למספר הסידורים האפשרי הוא: (בעזרת מקדם מולטינומי).

|  |  |  |  |
| --- | --- | --- | --- |
| Calculation time | Log2(#possiblePaths) | #possiblePaths | K |
| sec> | 16.79 |  | 5 |
| 76.11[sec] | 36.25 |  | 8 |
| 3.23[hours] | 43.5 |  | 9 |
| 0.07[years] | 51.07 |  | 10 |
| 16.21[years] | 58.929 |  | 11 |
| 239058[million years] | 92.709 |  | 15 |

1. הערך המקסימלי האפשרי של דרגת יציאה במרחב החיפוש הוא k, מכיוון שבמצב ההתחלתי קיימות k נקודות איסוף ולכן דרגת היציאה תהיה k. לא ייתכן כי דרגת היציאה תהיה גדולה מ-k כי לאחר כל איסוף יורדת נקודת אחת מהתחום של אופרטור האיסוף ונוספת נקודת פריקה אחת לתחום של אופרטור הפריקה. כלומר נוספת אפשרות אחת לדרגת היציאה של הצומת אך גם יורדת אחת. כמו כן לאחר פריקה של חבילה תרד דרגת היציאה ב-1 ולכן k הוא אכן הערך המקסימלי. הערך המינימלי האפשרי הוא 0 וזה אפשרי באחד ממצבי המטרה, בו נפרקו כל החבילות.
2. לא ייתכנו מעגלים מכיוון שנתון כי קבוצות נקודות האיסוף, הפריקה וההתחלה הן קבוצות זרות, ולכן כאשר נאספה או נפרקה חבילה בנקודה מסוימת לא ייתכן שנעבור בנקודה זו שוב, כלומר לא ייתכן מעגל.
3. מצב נוכחי מיוצג על ידי שלישייה של (נקודה נוכחית, קבוצת החבילות שבמשאית, קבוצת החבילות שנפרקו מהמשאית). לכן הביטוי עבור מספר המצבים האפשרי הוא:

הסבר- בכל רגע נתון יש לנו p חבילות שנאספו וכרגע על המשאית (טרם נפרקו), ולכן מספר האפשרויות לחבילות שנאספו הוא k בחר p. מתוך החבילות שאינן על המשאית (או שנפרקו או שלא נאספו) יש לנו d חבילות שכבר נפרקו, ולכן מספר האפשרויות לחבילות שנפרקו הוא k-p בחר d. לפי הבחירות שביצענו לעיל, כל מצב נוכחי יכול להיות במיקום של כל נקודה שנבחרה, לכן נכפיל ב-(p+d) עבור מספר האפשרויות של המיקום הנוכחי.

כל המצבים אכן ישיגים. מרחב המצבים שתיארנו וגודלו נבנו רק לפי מגבלות הבעיה, לכן כל מצב עוקב הוא חוקי וישיג מהמצב הנוכחי.

1. לא ייתכנן בורות ישיגים שאינם מצבי מטרה. אם הגענו לבור משמע לא קיימות קשתות יוצאות מהמצב, כלומר אין חבילות שנותרו לאיסוף, או חבילות שעל המשאית שצריכות להגיע לנק' פריקה שלהם. אך זהו בדיוק מצב מטרה, שכל החבילות נאספו ונפרקו. נוסיף כי ייתכן תחת הגבלת TruckCapacity שיווצרו בורות ישיגים שהם לא מצב מטרה, כאשר במצב בו המשאית ריקה, ונותר לנו חבילות לאסוף אך כל נקודת איסוף שטרם עברנו בה מספר החבילות בה גדול מהTruckCapacity.
2. כל מסלול מנק' ההתחלה אל מצב סופי באורך 2k+1. אנו מניחים שכל נק' האיסוף והפריקה זרות, ועל מנת לאסוף ולפרוק את כל החבילות צריך לעבור בכל נק' האיסוף והפריקה, לכן בספירת המצב ההתחלתי, נעבור ב2k+1 מצבים עד שנגיע למצב סופי – בו אספנו ופרקנו את כל החבילות.
3. מרחב המצבים מתואר כעץ, כאשר עומק העץ הוא אורך המסלול, מכיוון שכל רמה מייצגת נקודת איסוף או פריקה ולכן לפי סעיף 7 עומק העץ הוא 2k. הגרף הינו , גרף חסר מעגלים*.*

פונקציית העוקב פועלת באופן הבא:

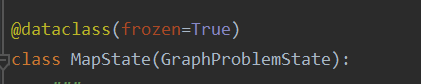
* אם ההובלה ה – i לא נמצאת על המשאית ולא סופקה בעבר ויש מספיק מקום פנוי במשאית עבור ההעמסת כל החבילות של ההובלה הזו – קיים מצב עוקב עבורו נאסוף את החבילה ה-i. המיקום הנוכחי יהיה di.pick, ונוסיף את החבילה למשאית.
* אם ההובלה ה – i נמצאת כרגע על המשאית – קיים מצב עוקב עבורו נפרוק את החבילה ה-i. המיקום הנוכחי יהיה di.drop ונסיר את החבילה מהמשאית.

1. **סעיף f:**

פלט הריצה המתוקנת:

StreetsMap(src: 54 dst: 549) UniformCost time: 0.49 #dev: 17354 |space|: 17514 total\_g\_cost: 7465.52560 |path|: 137 path: [ 54 ==> 55 ==> 56 ==> 57 ==> 58 ==> 59 ==> 60 ==> 28893 ==> 14580 ==> 14590 ==> 14591 ==> 14592 ==> 14593 ==> 81892 ==> 25814 ==> 81 ==> 26236 ==> 26234 ==> 1188 ==> 33068 ==> 33069 ==> 33070 ==> 15474 ==> 33071 ==> 5020 ==> 21699 ==> 33072 ==> 33073 ==> 33074 ==> 16203 ==> 9847 ==> 9848 ==> 9849 ==> 9850 ==> 9851 ==> 335 ==> 9852 ==> 82906 ==> 82907 ==> 82908 ==> 82909 ==> 95454 ==> 96539 ==> 72369 ==> 94627 ==> 38553 ==> 72367 ==> 29007 ==> 94632 ==> 96540 ==> 9269 ==> 82890 ==> 29049 ==> 29026 ==> 82682 ==> 71897 ==> 83380 ==> 96541 ==> 82904 ==> 96542 ==> 96543 ==> 96544 ==> 96545 ==> 96546 ==> 96547 ==> 82911 ==> 82928 ==> 24841 ==> 24842 ==> 24843 ==> 5215 ==> 24844 ==> 9274 ==> 24845 ==> 24846 ==> 24847 ==> 24848 ==> 24849 ==> 24850 ==> 24851 ==> 24852 ==> 24853 ==> 24854 ==> 24855 ==> 24856 ==> 24857 ==> 24858 ==> 24859 ==> 24860 ==> 24861 ==> 24862 ==> 24863 ==> 24864 ==> 24865 ==> 24866 ==> 82208 ==> 82209 ==> 82210 ==> 21518 ==> 21431 ==> 21432 ==> 21433 ==> 21434 ==> 21435 ==> 21436 ==> 21437 ==> 21438 ==> 21439 ==> 21440 ==> 21441 ==> 21442 ==> 21443 ==> 21444 ==> 21445 ==> 21446 ==> 21447 ==> 21448 ==> 21449 ==> 21450 ==> 21451 ==> 621 ==> 21452 ==> 21453 ==> 21454 ==> 21495 ==> 21496 ==> 539 ==> 540 ==> 541 ==> 542 ==> 543 ==> 544 ==> 545 ==> 546 ==> 547 ==> 548 ==> 549]

**סעיף g:**

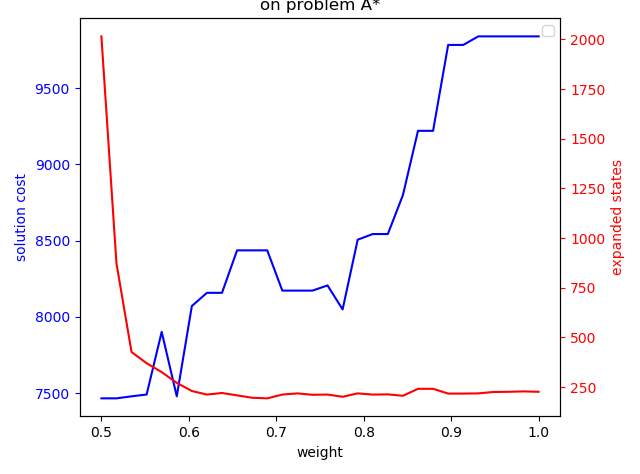


**חסר הסבר**

1. להלן:

StreetsMap(src: 54 dst: 549) A\* (h=AirDist, w=0.500) time: 0.10 #dev: 2015 |space|: 2229 total\_g\_cost: 7465.52560 |path|: 137 path: [ 54 ==> 55 ==> 56 ==> 57 ==> 58 ==> 59 ==> 60 ==> 28893 ==> 14580 ==> 14590 ==> 14591 ==> 14592 ==> 14593 ==> 81892 ==> 25814 ==> 81 ==> 26236 ==> 26234 ==> 1188 ==> 33068 ==> 33069 ==> 33070 ==> 15474 ==> 33071 ==> 5020 ==> 21699 ==> 33072 ==> 33073 ==> 33074 ==> 16203 ==> 9847 ==> 9848 ==> 9849 ==> 9850 ==> 9851 ==> 335 ==> 9852 ==> 82906 ==> 82907 ==> 82908 ==> 82909 ==> 95454 ==> 96539 ==> 72369 ==> 94627 ==> 38553 ==> 72367 ==> 29007 ==> 94632 ==> 96540 ==> 9269 ==> 82890 ==> 29049 ==> 29026 ==> 82682 ==> 71897 ==> 83380 ==> 96541 ==> 82904 ==> 96542 ==> 96543 ==> 96544 ==> 96545 ==> 96546 ==> 96547 ==> 82911 ==> 82928 ==> 24841 ==> 24842 ==> 24843 ==> 5215 ==> 24844 ==> 9274 ==> 24845 ==> 24846 ==> 24847 ==> 24848 ==> 24849 ==> 24850 ==> 24851 ==> 24852 ==> 24853 ==> 24854 ==> 24855 ==> 24856 ==> 24857 ==> 24858 ==> 24859 ==> 24860 ==> 24861 ==> 24862 ==> 24863 ==> 24864 ==> 24865 ==> 24866 ==> 82208 ==> 82209 ==> 82210 ==> 21518 ==> 21431 ==> 21432 ==> 21433 ==> 21434 ==> 21435 ==> 21436 ==> 21437 ==> 21438 ==> 21439 ==> 21440 ==> 21441 ==> 21442 ==> 21443 ==> 21444 ==> 21445 ==> 21446 ==> 21447 ==> 21448 ==> 21449 ==> 21450 ==> 21451 ==> 621 ==> 21452 ==> 21453 ==> 21454 ==> 21495 ==> 21496 ==> 539 ==> 540 ==> 541 ==> 542 ==> 543 ==> 544 ==> 545 ==> 546 ==> 547 ==> 548 ==> 549]

מספר פיתוחי המצבים היחסי שחסכנו לעומת הריצה העיוורת הוא – 7.61

1. 

קיבלנו גרף בו העקומה הכחולה מתארת את טיב הפתרון כתלות במשקל, ואילו העקומה האדומה מתארת את כמות הצמתים שפותחו כתלות במשקל. ניתן לראות כי ככל שהמשקל עולה, עלות הפתרון עולה ואילו כמות הצמתים שמפתחים קטנה. ניתן לראות כי העקומה האדומה מפסיקה לרדת החל מ- w=0.6 וניתן להבין כי זה מכיוון שעבור ערך זה מספר הצמתים המפותחים הוא אופטימלי.

הכלל אצבע "ככל ש-w קטן יותר כך הפתרון איכותי יותר ומס' הפיתוחים גדול יותר" נכון לרוב המשקלים המיוצגים בגרף, אך לדוג' עבור W~0.59 מתקיים שטיב הפתרון יותר טוב ממשקל של W~0.57 וגם כמות הצמתים שפותחו יותר קטנה, אך המשקל יותר גדול בניגוד לכלל האצבע.

המשקל W~0.59 הוא המשקל בו היינו בוחרים לאור תוצאות הגרף, כיוון שבשכלול 2 הפרמטרים (טיב הפתרון וכמות המצבים שפותחו) הפתרון האופטימלי שמתקבל הוא בערך זה.

1. נוכיח כי ההיוריסטיקה TruckDeliveriesMaxAirDistHeuristic הינה קבילה. נסמן ב את תוצאת ההיוריסטיקה הנ"ל ונראה כי מתקיים . h מחשבת מרחק אווירי מקס' בין כל זוג צמתים ע"י שימוש בפונק' get\_air\_distance\_between\_junctions. כיוון שהמרחק אווירי תמיד אי שלילי, הפונק' הנ"ל מחזירה את המרחק האווירי המקס' שהיא מוצאת אז בפרט היא מחזירה מספר אי שלילי, ולכן מתקיים בהכרח .

הפונק' מחזירה מרחק בין 2 צמתים במסלול שנותר למשאית לעבור, שהמרחק האווירי ביניהם הוא מקס' מכל צמדי הצמתים האפשריים במסלול, נסמן את זוג הצמתים הללו a,b.

תהי ההיוריסטיקה המושלמת, אזי המסלול שהיא מחשבת חייב לעבור בשני הצמתים הללו, בה"כ הוא עובר בa לפני b, ויש שתי אפשרויות למעבר.

אפשרות ראשונה שמהצומת a נעבור ישירות לצומת b, ובמקרה הזה המרחק אשר תחזיר יהיה שווה למרחק ש תחזיר ויתקיים .

אפשרות שניה היא לעבור בין a לb בצמתים נוספים ולא במעבר ישיר, ואז במצב הזה המרחק שנעבור בהכרח יותר גדול מהמרחק האווירי בין a לb, ולכן *.*

*בסה"כ קיבלנו כי מתקיים*   *ולכן ההיוריסטיקה קבילה.*

1. *נראה כי ההיוריסטיקה* TruckDeliveriesSumAirDistHeuristic אינה קבילה ע"י דוגמא נגדית. נסתכל על המסלול שנותר לעבור הבא, כאשר a הצומת הנוכחית: (כל הצמתים שצריך לעבור בהם והכבישים המחברים אותן)

**1**

**5**

**2**

**3**

**3**

**2**

נסמן ב את תוצאת ההיוריסטיקה הנ"ל*. המסלול שיבחר לפי החישוב שלה הוא* a->b>c>d *ונקבל ש . אך קיים מסלול אופטימלי עובר בצמתים a->c->b->d ועבורו נקבל , כלומר ולכן ההיוריסטיקה אינה קבילה.*

1. נוכיח כי ההיוריסטיקה TruckDeliveriesMSTAirDistHeuristic הינה קבילה. נסמן ב את תוצאת ההיוריסטיקה הנ"ל ונראה כי מתקיים *.* h *מחשבת את משקל העפ"מ של גרף הצמתים שנותר לנו לעבור בהם, כאשר משקל כל קשת הוא המרחק האווירי בין 2 צמתי הקשת. נסמן את הגרף* G *ואת משקל העפ"מ* w(T)*. מרחק אווירי בין כל זוג צמתים בבעיה הוא אי שלילי, ולכן סכום משקלי קשתות (כל קב' של קשתות בגרף) הוא אי שלילי,* ולכן מתקיים בהכרח .

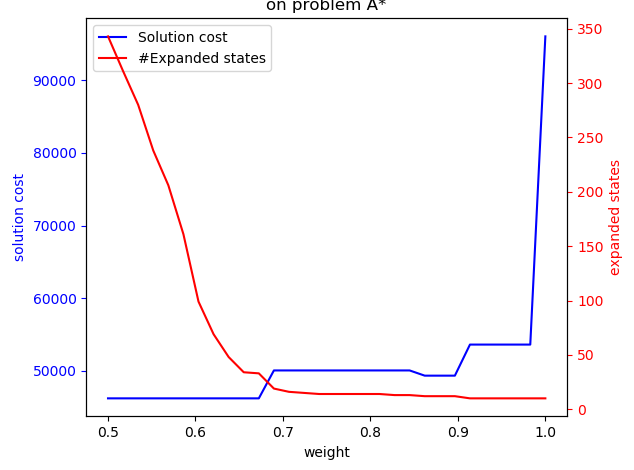
נניח בשלילה כי  *אינה קבילה, משמע קיים פתרון אופטימלי המקיים . הפתרון הוא משקל עפ"מ של גרף* G. *נסמן את משקל העפ"מ הנ"ל* w(T')*, כלומר קיבלנו ש* w(T) < w(T'). *אך זה בסתירה לכך שמחשבת משקל של עץ פורש מינימלי של* G*.*

*ולכן אכן מחשבת את הפתרון האופטימלי, משמע ההיוריסטיקה קבילה.*

1. *להלן הגרף עבור הרצת run\_astar\_for\_weights\_in\_range() לבעיה small\_delivery\_problem\_with\_distance\_cost כאשר ההיוריסטיקה לפיה החישוב מתבצע היא TruckDeliveriesMSTAirDistHeuristic:*

Quality vs. time for wA\*

on problem A\*

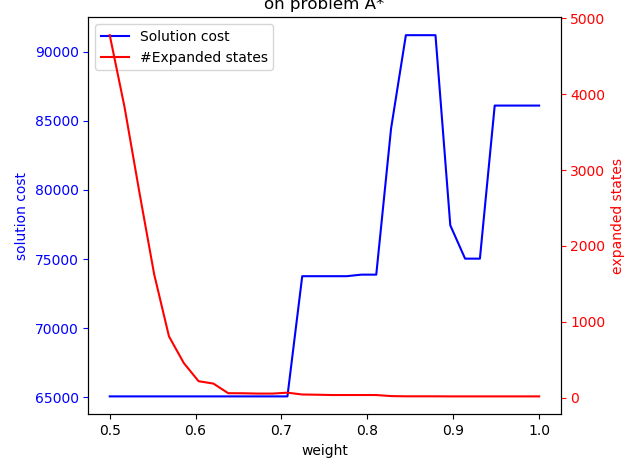


ניתן לראות כאן כי ככל שהמשקל של הפונקציה היוריסטית גדל, כך יורדת כמות הפיתוחים בגרף ואיכות הפתרון יורדת. מגמת הגרף תואמת ברובה את "כלל האצבע" אך כפי שהוסבר בסעיף 14 נציין כי זו המגמה הכללית והכלל לא נכון לכל נקודה בעקום באופן גורף. היינו בוחרים ערך של w בטווח של [0.67,0.68] מכיוון שבשכלול 2 הפרמטרים (טיב הפתרון וכמות המצבים שפותחו) הפתרון האופטימלי שמתקבל הוא בטווח זה.

*להלן הגרף עבור הרצת run\_astar\_for\_weights\_in\_range() לבעיה moderate\_delivery\_problem\_with\_distance\_cost כאשר ההיוריסטיקה לפיה החישוב מתבצע היא TruckDeliveriesSumAirDistHeuristic:*

Quality vs. time for wA\*

on problem A\*



בגרף זה ניתן לראות כי המגמה של טיב הפתרון אינה עולה באופן קבוע ככל שה-w גדל, למרות שמגמת פיתוח הצמתים היא מגמה יורדת כפי שציפינו. ניתן להסביר את ההתנהגות החריגה של הגרף כתוצאה מכך שלפי סעיף 22 ההיורסטיקה הנוכחית אינה קבילה. היינו בוחרים ערך של W~0.65, משקל עבורו 2 המדדים מתלכדים בערך אופטימלי.

1. ניתן לראות כי הפתרון המתקבל בכל הרצה מחשב לפי המדד הרלוונטי בעזרת פונקציית העלות המתאימה. כלומר, בחישוב הבעיה בעזרת פונקציית הזמן, עלות הזמן שהתקבלה היא 30 דקות אבל לעומת זאת בעזרת פונקציית המחיר, עלות הזמן שהתקבלה היא 35 דקות. באותו אופן, בחישוב הבעיה בעזרת פונקציית המחיר, עלות המחיר שהתקבלה היא 104 ₪ אבל לעומת זאת בעזרת פונקציית הזמן, עלות המחיר שהתקבלה היא 152 ₪ . כצפוי קיבלנו פתרון טוב יותר לפי מדד מסוים כאשר השתמשנו בפונקציית העלות המתאימה עבור מדד זה.
2. כמות הפיתוחים שחסכנו היא 37 צמתים. ידענו לצפות כי כמות הפיתוחים תהיה טובה יותר מכיוון שלפי עקרון האלגוריתם של A\*eps אנחנו מוכנים לוותר עד כדי אפסילון על איכות הפתרון בתמורה לקבלת פתרון מהיר יותר. נציין כי במקרה זה איכות הפתרון כלל לא נפגעה לעומת חישוב ה-A\* הרגיל.
3. לפי חלק א' עבור הרצת A\* על הבעיות moderateDelivery, bigDelivery, בעיות בעלות ערך k=8,15 בהתאמה, זמן החישוב הנדרש לעבור על כל הסידורים האפשריים מאוד גדול עבור k=8 ואילו עבור k=15 הזמן הנדרש אינו אפשרי ליישום. Anytime A\* מאפשר לנו להגביל את זמן החישוב על חשבון טיב הפתרון. במקרה שלנו אכן זמן החישוב ירד (לעומת חלק א) וקיבלנו פתרון מסוים, שניתן להניח כי הוא אינו הטוב ביותר.