מבוא לבינה מלאכותית - תרגיל בית 1

אלעד אספיס 313597023   אור בלסקי 316406321

30 בנובמבר 2020

שאלה 1

1. נשים לב כי בסוף המסלול אנו חייבים להגיע למעבדה כלשהי (יש m מעבדות לבחירה).
2. בנוסף, לאחר כל דייר שביקרנו יש לנו את האפשרות לבקר במעבדה (m אפשרויות) או לא לבקר במעבדה. כלומר, סה"כ יש $m+1$ אפשרויות. מכיוון שיש k דיירים יש לנו k אפשרויות לבחור.

סה"כ, אנו מקבלים כי שמספר המסלולים שמתקבלים הוא $m \cdot k! \cdot (m+1)^k$.

שאלה 3

k	m	$\#possiblePaths$	$Estimated\ calculation\ time$
7	2	2.2044960×10^7	$18.47\ [secs]$
7	3	2.47726080×10^8	$3.84\ [mins]$
8	3	7.927234560×10^9	$2.25\ [hours]$
8	4	6.3×10^{10}	$19.55\ [hours]$
9	3	$2.8538044416 \times 10^{11}$	$3.69\ [days]$
10	3	$1.1415217766 \times 10^{13}$	$5.33\ [months]$
11	3	$5.0226958172 \times 10^{14}$	$20.76\ [years]$
12	3	$2.4108939923 \times 10^{16}$	$1.06\ [thousand\ years]$
12	4	4.67775×10^{17}	$22.10\ [thousand\ years]$
13	4	3.0405375×10^{19}	$1.52\ [millon\ years]$

שאלה 4

ערכי הקיצון המינימלי - נתאר מצב בו דרגת היציאה היא 0 :

במצב בו עברנו בכל המעבדות על מנת לאסוף מטושים ואנו במעבדה ה-m ואין מספיק מטושים על מנת לעבור ולו בדירה אחת, ולכן אנו נשאר במעבדה הנוכחית ובמצב זה דרגת היציאה היא 0.

ערכי הקיצון המקסימלי - נתאר מצב בו דרגת היציאה היא $k+m$:

כאשר האמבולנס נמצא ב-v_0 (כלומר במצב ההתחלתי) והמטושים שברשותו מספיקים עבור כל d_i הדירות כאשר $1 \leq i \leq k$ - במצב זה האמבולנס יוכל לעבור בכל אחת מ-k הדירות (כי יש לו מספיק מטושים) ובנוסף יוכל לעבור בכל אחת מ-m המעבדות על-מנת לאסוף מטושים נוספים. סה"כ, קיבלנו כי $k+m$.

שאלה 5

לא ייתכנו מעגלים במרחב החיפוש שלנו מכיוון שבמידה ואנו עוברים פעם בדירה מסוימת בהכרח לא נעבור בה שוב.

במקרה עבור המעבדות - נניח בשלילה שקיימים שני מצבים s_i, s_j כך ש-$o_{l_k}(s_i) = o_{l_k}(s_j)$. נניח בה"כ כי $o_{l_k}(s_i)$ מהווה את הביקור הראשון במעבדה. לכן $o_{l_k}(s_j)$ מהווה את הביקור השני. מהגדרת $canVisit$ בהכרח $s_j.Taken \neq \emptyset$. נשים לב כי $s_j.Taken \neq s_i.Taken$ כי כל פעם אנו עוברים בדירות אחרות (או שייתכן ש-$s_i.Taken = \emptyset$). לכן בהכרח מתקיים $s_i.Transferred \cup$ $s_i.Taken \neq s_j.Transferred \cup s_j.Taken$, בסתירה להנחה ש-$o_{l_k}(s_i) = o_{l_k}(s_j)$.

שאלה 6

נשים לב כי מספר המטושים הזמינים אשר יכולים להיות הוא למעשה כל מספר טבעי ולכן ניתן להסיק כי קיימים אינסוף מצבים במרחב זה.

לא כל המצבים ישיגים. לדוגמא, בבעיה הבאה נקבל מצב לא ישיג :

$d_i.rommate = 10$ לכל $1 \leq i \leq k$ ומספר המטושים במעבדה $l_j = 1$ לכל $1 \leq j \leq m$ ו-1 $InitialNrMatoshimAmb$ שווה ל1. המצב$(d_1.loc, \emptyset, \emptyset, 0, \emptyset)$ איננו ישיג - כי עלינו להשיג לפחות 10 מטושים כדי להגיע לדירה d_1 אבל אין מעבדה שיכולה לספק לנו 10 מטושים.

שאלה 7

ייתכנו, כאשר הסתיימו בכל המעבדות המטושים והסתיימו המטושים באמבונלס ונותרו לנו עוד בדיקות לבצע אז לא נוכל לבדוק את כל הדיירים. לכן, נגיע למעבדה כלשהי ונשאר שם כי אין סיבה לצאת לבדוק דיירים ללא מטושים.

שאלה 8

בהנחה ומצב סופי הוא מצב מטרה אז הטווח המינימלי ממצב התחלתי אל המצב הסופי הוא $k+1$ - למשל במצב בו אנו יוצאים מ-v_0 עם מספיק מטושים כדי לבקר בכל הדירות ברצף, עוברים בכל הדירות והולכים למעבדה.

הטווח המקסימלי הוא $2k+m$ - כאשר אנו מתחילים ב-v_0, עוברים בכל המעבדות, עוברים לדירה ולאחר מכן במעבדה על מנת לשים

את הבדיקות במעבדה - באופן הזה נחזור על התהליך k פעמים (מעבדה, דירה, מעבדה, דירה וכו') ולבסוף נחזור למעבדה.

שאלה 9

נגדיר:

$$Succ_{MDA}(s) = \{(d_i.loc, s.Taken \cup \{d_i\}, s.Transferred, s.Matoshim - d_i.roommates, s.VisitedLabs) \,|CanVisit(s, d_i)\}$$

$$\cup \left\{\left(l_i.loc, \emptyset, s.Taken \cup s.Transferred, s.Matoshim + l_i.matoshim \cdot \mathbb{I}_{[l_i \notin s.VisitedLabs]}, s.VisitedLabs \cup \{l_i\}\right) |CanVisit(s, l_i)\right\}$$

שאלה 14

מספר הפיתוחים שהתבצעו באלגוריתם העיוור הוא 17354.

מנגד, מספר הפיתוחים שהתבצעו באלגוריתם היוריסטי הוא 2015.

לכן, ההפרש חלקי מס' הפיתוחים בריצה בלי ההיוריסטיקה הוא $\frac{17354-2015}{17354} = 0.883$

שאלה 16

ניתן לראות שהקו הכחול מסמן את עלות הפתרון כפונקציה של המשקל של הפונקציה היוריסטית.

לעומת זאת, הקו האדום מתאר את מספר הצמתים שפותחו כפונקציה של המשקל של הפונקציה היוריסטית.

נשים לב שבאזור נקודת החיתוך (כאשר $0.53 \leq weight \leq 0.6$) הן מספר הצמתים שמפותחים והן עלות הפתרון הוא נמוך ועל כן זהו אזור כדאי.

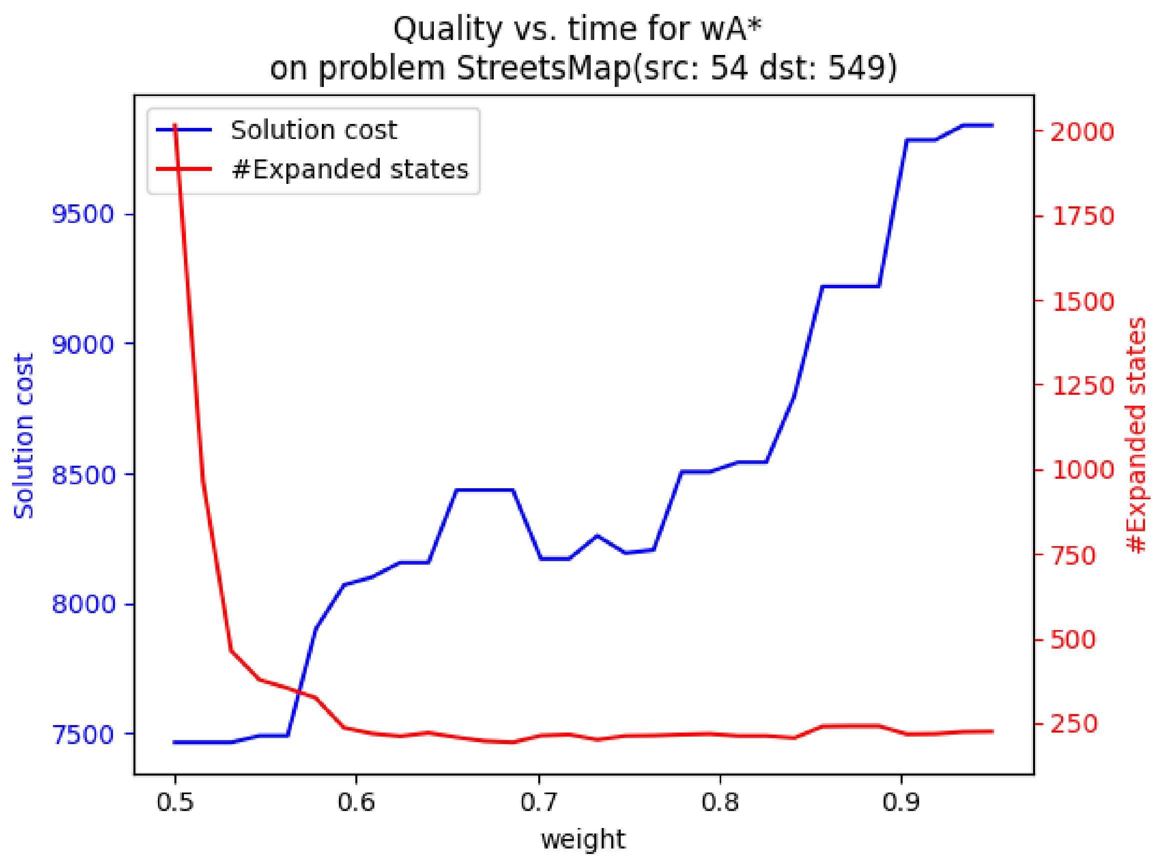
בנוסף, נשים לב כי קצוות הגרף (עבור $weight > 0.9$ או $weight = 0.5$) מתקיים שעלות הפתרון היא גבוה מאוד או מספר הצמתים שמפותחים הוא גבוה מאוד. נבחין כי בהשוואה ל-$weight = 0.9$, ב$weight = 0.7$ אנו מפתחים את אותו מספר הצמתים בערך אבל חוסכים משמעותית ולכן כאשר $weight = 0.9$ המצב איננו כדאי.

כמו כן, נשים לב שכאשר $weight = 0.85$ מספר הצמתים יותר גבוה מאשר $weight = 0.83$ - בניגוד לכלל אצבע לפיו "ככל ש-w קטן יותר כך הפתרון איכותי יותר ומספר הפיתוחים גדול יותר.

דוגמא נוספת לגבי עלות הפתרון, כאשר $weight \approx 0.7$ אנו מקטינים את המשקל ל-0.68 אך עלות הפתרון עולה - בניגוד לכלל האצבע.

בנוסף, בתחום $0.6 \leq weight < 1$ מספר הצמתים נע סביב 250, כלומר מקטינים משמעותית את פונקצית המשקל אך מספר הפיתוחים לא עולה - בניגוד לכלל האצבע.

באופן כללי, בהסתכלות כללית כלל האצבע אכן מתקיים למעט מקרים בודדים אותם פירטנו לעיל.



שאלה 19

חסרון בגישה שכזאת מבחינת יעילות הפתרון הוא גידול משמעותי במספר הצמתים המפותחים. כלומר, עבור כל צומת עוקב במרחב האלגוריתם מחפש מסלול וזה מגדיל את סיבוכיות הזמן. במקרה זה אם נסמן ב-j את מספר הצמתים המקסימלי, נקבל סיבוכיות זמן גדולה בהרבה לעומת סיבוכיות הזיכרון בסעיף 1 אשר תהיה תלויה ב-j. בנוסף, גודל ה-$cache$ ייגדל מאוד - מה שיגדיל את סיבוכיות המקום.

שאלה 20

סעיף א'

שורת הקוד הרלוונטית היא: $@dataclass(frozen = True)$

סעיף ב'

שורה זו אינה מספיקה כי אם השדות הפנימיים של $MDAState$ הם מבני נתונים ואינם מוגדרים כ-$frozen$ אז יהיה ניתן לשנות אותם (למשל הוספת והסרת איבר למבנה הנתונים).

לכן, כל אחד ממבני הנתונים המוגדרים כשדות של $MDAState$ צריכים להיות $frozen$.

מה שעוד מבטיח בקוד שלא יהיה ניתן לשנות את האובייקט ו/או את המבנים שהוא מחזיק זה למשל להגדיר מבני נתונים כ-$frozenset$ כדי שהשדות הפנימיים של $MDAState$ יהיו בלתי ניתנים לשינוי.

סעיף ג'

ייתכן שנפגוש מצב בשנית גם לאחר שפיתחנו אותו.

בשורה הבאה:

```
old_node ← find-state(s,CLOSED)
```

סעיף ד'

הסיבה שאנו רוצים לעשות את זה ספציפית עבור $MDAState$ היא כדי שלא נבצע העתקה $by\ reference$ ואז אם נשנה את השדות של המשתנה החדש שניצור, נשנה את הערכים של המשתנה שהעתקנו ואנו לא רוצים לשנות את המקור אלא רק את האובייקט החדש.

דוגמא למימוש שגוי של המתודה $expand\_state\_with\_costs$ היא:

$$copied\_state = state\_to\_expand$$

$$copied\_state.nr\_matoshim\_on\_ambulance\ = 100$$

כתוצאה מכך, במידה ו-$MDAState$ לא יהיה "קפוא" נקבל שבטעות שינינו את $state\_to\_expand$ מבלי שבאמת רצינו.

שאלה 23 - הוכחה

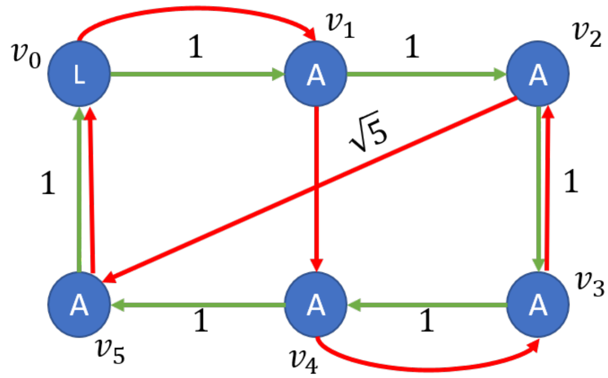
היוריסטיקה $MDAMaxAirDistHeuristic$ היא קבילה. נוכיח זאת:

ראשית, ברור ש-$h\left(s\right) \geq 0$ לכל מצב s כי אנו מחשבים מרחק בין צמתים ומרחק הוא אי-שלילי.

יהי s מצב כלשהו. נראה כי $h\left(s\right)\ \leq\ h^*\left(s\right)$. נשים לב כי $h\left(s\right)$ הוא המרחק האווירי הגדול ביותר של הצומת (נסמנו בקודקוד u) הכי רחוקה מ-u, נסמנה v אשר שייכת ב-$remaining\ junctions$ לפי הגדרת היוריסטיקה. $h^*\left(s\right)$ מחזיר את עלות המסלול האופטימלי מ-s עד מצב המטרה.

נשים לב כי המסלול האופטימלי (עם העלות האמיתית של h^*) מכיל בפרט את המסלול $u\ \rightarrow\ v$ ואולי עוד נקודות שונות בדרך כי חייבים לעבור ב-v בגלל שהוא ב-$remaining\ junctions$. לכן ניתן להסיק כי עבור מצב שרירותי s מתקיים $0 \leq h\left(s\right) \leq h^*\left(s\right)$, כנדרש.

שאלה 26 - הפרכה



מספר המטושים במעבדה L : 10, מספר דיירים בכל דירה : 1, קיבולת האמבולנס : 20, מספר מטושים התחלתי : 10, L מהווה מעבדה ו-A מהווה דירה.

המסלול הירוק מסמל את המסלול האופטימלי מהמצב ההתחלתי (הנקודה שמסומלת ב-L) המהווה נקודת התחלה וסיום, שהיא במקרה המעבדה היחידה במפה.

המסלול האדום הוא המסלול המבוסס על הפונקציה היוריסטית $MDASumAirDistHeuristic$.

מתקיים : $h\left(s\right)=5+\sqrt{5}>6=h^{*}\left(s\right)$ כי המסלול אדום ארוך מהמסלול הירוק, כלומר $h\left(s\right)>h^{*}\left(s\right)$ כאשר s הוא המצב ההתחלתי שמיקומה במפה היא הנקודה המסומלת ב-L, לכן היוריסטיקה איננה קבילה.

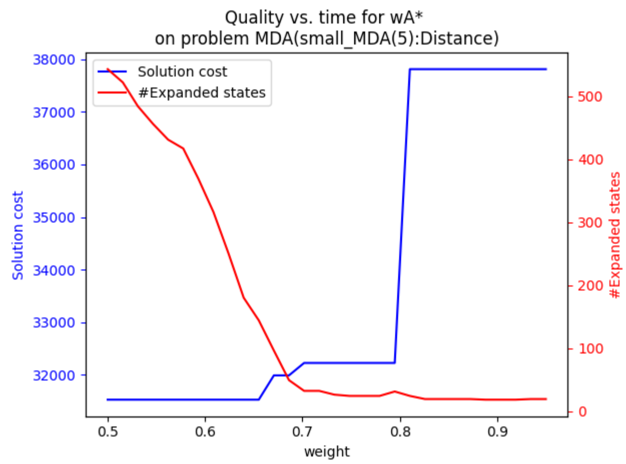
שאלה 29

היוריסטיקה $MDAMSTAirDistHeuristic$ הינה קבילה.

ראשית, ברור ש-$0 \geq h\left(s\right)$ לכל מצב s כי אנו מחשבים מרחק בין צמתים ומרחק הוא אי-שלילי. ולכן משקל העץ לא יכול להיות שלילי כסכום של מספרים אי-שליליים.

בכל צומת יוריסטיקה זו מחשבת העץ הפורש המינימלי של כל הדירות הנותרות לביקור - כלומר כל הדירות ישתתפו במסלול. מבין כל המסלולים נקבל את המסלול המינימלי מבחינת משקל הקשתות (מרחק בין 2 צמתים) - זהו למעשה משקל העפ"מ. בדרך זו קיבלנו את המסלול המינימלי ולכן $0\leq h\leq h^{*}$.

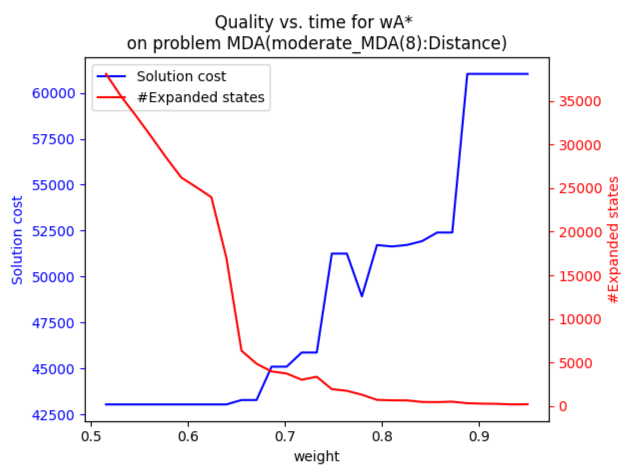
שאלה 30



נשים לב שבאזור נקודת החיתוך (כאשר $0.67 \leq weight \leq 0.7$) הן מספר הצמתים שמפותחים והן עלות הפתרון הוא נמוך ועל כן זהו אזור כדאי.

בנוסף, נשים לב כי עבור $weight > 0.8$ עלות הפתרון היא גבוה מאוד ועבור $weight = 0.5$ מספר הצמתים שמפותחים הוא גבוה מאוד.

נבחין כי בהשוואה ל-$weight = 0.82$, ב$weight = 0.77$ אנו מפתחים את אותו מספר הצמתים בערך אבל חוסכים משמעותית בעלות הפתרון ולכן כאשר $weight = 0.82$ המצב איננו כדאי.



נשים לב שבאזור נקודת החיתוך (כאשר $0.69 \leq weight \leq 0.7$) הן מספר הצמתים שמפותחים והן עלות הפתרון הוא נמוך ועל כן זהו אזור כדאי ולכן נבחר $weight$ בתחום זה.

בנוסף, נשים לב כי עבור $weight > 0.8$ אנו כמעט ולא חוסכים בפיתוחי הצמתים אך המחיר במגמת עלייה. יתר על כן, כאשר ב-$weight \approx 0.88$ יש עלייה דרסטית בעלות הפתרון ולכן אין טעם לבחור ב-$weight > 0.8$.

שאלה 31

$MDAMSTAirDistHeuristic$	$MDASumAirDistHeuristic$	$MDAMaxAirDistHeuristic$	
לא	לא	אל	$cost_{MDA}^{test\ travel}$
לא	לא	לא	$cost_{MDA}^{monetary}$

שאלה 32

נשים לב שכאשר השתמשנו בפונקציית העלות $cost_{Monetary}$ אזי הערך של $total\_g\_cost$ בבעיה של $small\_MDA$ הוא 42.04962 ועבור הבעיה של $moderate\_MDA$ הוא 77.20101, בניגוד למקרה בו השתמשנו בפונקצית העלות $cost_{Distance}$ שערכה של $total\_g\_cost$ בבעיה של $small\_MDA$ היה 31528.65909.

שאלה 34

היוריסטיקה $MDATestsTravelTimeToNearestLabHeuristic$ היא קבילה. נוכיח זאת:

ראשית, ברור ש-$h\left(s\right) \geq 0$ לכל מצב s כי אנו מחשבים מרחק למעבדה הקרובה ומרחק הוא אי-שלילי.

נשים לב כי $h^*\left(s\right)$ אופטימלי ביחס לפונקציה $cost_{MDA}^{test\ travel}$ כאשר אנו רוצים לנסוע את המרחק המינימלי כשיש באמבולנס בדיקות. לכן, לאחר כל בדיקה בדירה ניגש למעבדה הקרובה על-מנת להעביר את הבדיקות למעבדה ונמשיך לדירה הבאה - בדרך זו נקבל את המסלול האופטימלי. נשים לב כי זה מה שפונקציה $h\left(s\right)$ מעריכה.

יהי s מצב. במידה והמצב הנוכחי הוא דירה, אז $h\left(s\right)$ הוא העלות למעבדה הקרובה ביותר. נשים לב כי $h^*\left(s\right)$ ביחס לפונקציה $cost_{MDA}^{test\ travel}$ מכיל לפחות את העלות של נסיעה למעבדה הקרובה וביקור בדירות נוספות אם קיימות ב-$remain\ junctions$ ולכן מתקיים במצב זה $h\left(s\right) \leq h^*\left(s\right)$.

אחרת, אם s הוא מעבדה, אז $h\left(s\right) = 0$ כי s היא המעבדה הקרובה לעצמה ולכן מתקיים במצב זה $h\left(s\right) = 0 \leq h^*\left(s\right)$.

סה"כ, קיבלנו כי לכל מצב s מתקיים $0 \leq h\left(s\right) \leq h^*\left(s\right)$.

שאלה 35

נבצע את ההשוואה ע"י טבלה

	$MaxAirDist$	$SumAirDist$	$MSTAirDist$	$TestsTravelTimeToNearestLab$
$cost_{MDA}^{test\ travel}$	176505.013	176505.013	176505.013	131265.15303

נשים לב כי אכן רואים את המזעור במדד הרלוונטי (הפרש בעלות של כ-45 אלף מטר), כנדרש.

שאלה 36 - הוכחה

הטענה נכונה. נשים לב ש-C^*_{dist} היא העלות המינימלית לאחר ריצת שלב (i). יהא P המסלול המתאים לעלות של C^*_{dist}. נסמנו $P = s_0 \overset{\tilde{o_0}}{\to} \dots \overset{\tilde{o_{d-1}}}{\to} s_d$. נשים לב כי לכל $0 \le t \le d$ מתקיים עבור $o = \tilde{o}_t$: $\sum_{i=0}^{t-1} cost^{dist}_{MDA}(s_i, \tilde{o}_i) \le cost^{dist}_{MDA}(s_t, \tilde{o}_t) +$ ולכן מתקיים $(1+\varepsilon) \cdot C^*_{dist}$ $cost\left(\left\langle s_0 \overset{\tilde{o_0}}{\to} \dots \overset{\tilde{o_{d-1}}}{\to} s_d \right\rangle\right) = cost^{test\ travel}_{MDA}(s_d, 0) < \infty$ ולכן אלגוריתם $\mathcal{A}_1$ בהכרח מחזיר פתרון כלשהו - את הפתרון הנ"ל או אחד אחר יותר טוב.

שאלה 37 - הוכחה

ראשית, נשים לב כי לפי ההגדרה: $cost^{test\ travel}_{MDA}(s,o) = \left(\sum_{d \in s.Taken} d.roomates\right) \cdot cost^{dist}_{MDA}(s,o)$.

נבחין בכל צומת לפיתוח s בדרך לפתרון הערך $\left(\sum_{d \in s.Taken} d.roomates\right)$ קבוע. אנו יודעים שיש פתרון עם עלות מינימלית ביחס ל-$cost^{dist}_{MDA}$ השמור ב-C^*_{dist} - לכן לפחות עבורו התנאי מתקיים. מכיוון שהוא מינימלי ביחס ל-$cost^{dist}_{MDA}$ הוא גם מינימלי ביחס ל-$cost^{test\ travel}_{MDA}(s,o)$ בגלל היחס הלינארי ביניהם.

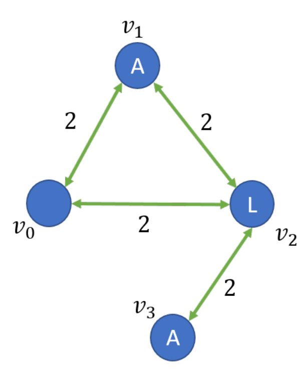
שאלה 38

	$MDAMSTAirDistHeuristic$	$MDATestsTravelTimeToNearestLabHeuristic$
$Cost^{dist}_{MDA}$	43034.794	93226
$Cost^{cost-travel}_{MDA}$	176505.013	131265.153
$Cost^{combined}_{MDA}$		67525.188

נשים לב כי $0.569 = \frac{67525.188}{43034.79} - 1 = \frac{DistCost(ReturnedSolution)}{C^*_{dist}} - 1$.

כלומר ערך ה-ε נשמר.

שאלה 39 - הפרכה



נניח כי מספר הדיירים ב-v_1 וב-v_3 הוא 2 והתחלנו ב-v_0 עם 2 בדיקות במקרה.

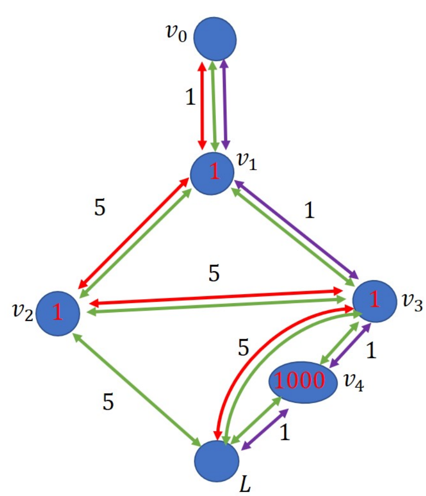
ראשית, נשים לב שהפתרון האופטימלי ביחס ל-$cost^{dist}_{MDA}$ הוא $v_0 \to v_1 \to v_2 \to v_3 \to v_2$ ועלותו היא $2+2+2+2=8$. לכן, נסיק כי $C^*_{dist}=8$.

כעת, נפעיל את שלב (iii) באלגוריתם $\mathcal{A}_2$.

$current\_state$	$open$	$close$	g
v_0	$[v_2, v_1]$	$[]$	0
v_2	$[v_1, v_3]$	$[v_0]$	4
v_1	$[v_2, v_3]$	$[v_0]$	4
v_2	$[v_3]$	$[v_1, v_0]$	8
v_3	$[v_2]$	$[v_1, v_0]$	8
v_2	$[]$	$[v_1, v_0, v_3]$	12

בשלב האחרון שהגענו ל-v_2 בסוף הקריטריון המשולב לא מתקיים מכיוון ש-$cost\left(v_0 \to v_2 \to v_1 \to v_2 \to v_3 \to v_2\right) = 2+2+2+2+2 = 10 > (1+0.2)\cdot 8 = 1.2 \cdot 8 = 9.6$ ולכן הצומת v_2 נמחקת (זוהי המעבדה היחידה בבעיה שלנו) והוא מצב המטרה ולכן נקבל שלא קיים פתרון.

שאלה 40 - הפרכה



להלן, טבלת מעקב של המסלול האדום :

$current\_state$	$open$	$close$	$Cost_{MDA}^{test\ travel}$
v_0	$[v_1]$	$[]$	0
v_1	$[v_2, v_3]$	$[v_1]$	0
v_2	$[v_2, v_3]$	$[v_0]$	5
v_3	$[v_3]$	$[v_1, v_0]$	$5 + 10 = 15$
L	$[]$	$[v_1, v_0, v_3]$	$5 + 10 + 15 = 30$

להלן, טבלת מעקב של המסלול הסגול:

$current\_state$	$open$	$close$	$Cost_{MDA}^{test\ travel}$
v_0	$[v_1]$	$[]$	0
v_1	$[v_2, v_3]$	$[v_1]$	0
v_3	$[v_2, v_3]$	$[v_0]$	1
v_4	$[v_3]$	$[v_1, v_0]$	$1 + 2 = 3$
L	$[]$	$[v_1, v_0, v_3]$	$1 + 2 + 1002 = 1005$

סימון - המספרים השחורים מסמנים את $cost_{MDA}^{dist}$ מצומת v_i ל-v_j כלשהו.

v_0 זוהי נקודת התחלה. בנוסף, v_4, v_3, v_2, v_1 הם כולם דירות והמספרים האדומים בתוך הצמתים מסמנים את מספר הדיירים בדירה.

L מסמלת את המעבדה היחידה במפה וחשוב לציין שאין בה מטושים לאיסוף.

נניח כי מספר המטושים באמבולנס הוא 1002 ובנוסף $AmbulanceTestsCapacity = 1500$.

נשים לב ש-$C^*_{dist} = 4$. נבחר $\varepsilon = 0.1$. נחשב את העלות של המסלול האופטימלי ביחס לקריטריון המשולב.

נתאר את אופן הפעולה של האלגוריתם.

נפתח את v_1 ונקבל ש-$open = [v_2, v_3]$.

במידה ופיתחנו את v_2 קודם - נבחין כי בחישוב התנאי על הצומת v_2 נקבל כי $1 + 5 \overset{(*)}{>} (1 + 0.1) \cdot 4$ ולכן לפי האלגוריתם $\mathcal{A}_2$ נסיר את v_2 מ-$open$.

במידה ופיתחנו צמתים אחרים לפני שפיתחנו את צומת v_2 כמובן שה-$cost$ הכולל רק יילך ויגדל (אגף שמאל ב-$(*)$) רק יילך ויגדל ובטח שלא יקיים את התנאי המבוקש).

מכיוון שהמסלול הסגול הוא בעל העלות המינימלית ביחס ל-$cost_{MDA}^{dist}$ וכמו כן מקיים את התנאי שהעלות עד צומת הנוכחית קטנה מ-$(1 + \varepsilon) \cdot C^*_{dist}$ לכל צומת במסלול זה נקבל שהפתרון של המסלול הסגול יתקבל. נשים לב שפתרון זה איננו האופטימלי אלא דווקא זה שעובר שצומת שנמחק (צומת v_2). נראה זאת.

אורך הפתרון של המסלול הסגול הוא: $1 \cdot 1 + 2 \cdot 1 + 1002 \cdot 1 = 1005$.

אורך הפתרון של המסלול האדום הוא: $1 \cdot 5 + 2 \cdot 5 + 3 \cdot 5 = 30$.

נשים לב שאנו עושים חישוב זה עבור המסלול האדום כי למרבה הצער אין מספיק מטושים על מנת לעבור ב-v_4 - יש 999 מטושים שם פנויים אבל בדירה v_4 יש 1000 דיירים, לכן לא נותר אלא להמשיך למעבדה ולסיים את המסלול.

לכן, נסיק כי הפתרון של המסלול השמאלי אופטימלי יותר אבל הוא לא הוחזר כי v_2 נמחק מה-$open$.

שאלה 41

היתרון של $\mathcal{A}_2$ על $\mathcal{A}_1$ הוא שאנו לא שומרים צמתים מיותרים ב-$open$ אשר לא עומדים בתנאי הבא: ערך ה-$cost_{MDA}^{dist}$ קטן או שווה מ-$(1+\varepsilon)\cdot C_{dist}^*$ ובכך אנו חוסכים מקום רב בזיכרון.

נשים לב ש-$\mathcal{A}_1$ מכיל מספר מסלולים גדול מאשר $\mathcal{A}_2$ עקב מחיקת הצמתים.

שאלה 44

להלן התוצאות:

```
MDA(small_MDA(5):Distance)          A* (h=MDA-MST-AirDist, w=0.500)    time:  18.45    #dev: 543     |space|: 877      total_g_cost: 31528.65909
MDA(small_MDA(5):Distance)          A*eps (h=MDA-MST-AirDist, w=0.500)    time:    2.55    #dev: 490     |space|: 818      total_g_cost: 31528.65909
```

נשים לב כי כאשר השתמשנו ב-$A^*\varepsilon$ חסכנו ב-53 צמתים. כאשר השתמשנו ב-$A^*\varepsilon$ נחשפנו לצמתים שהדרך לפתרון קרובה יותר והערך היוריסטי שלהם אולי יותר גבוה מאחרים אשר אלגוריתם A^* לא היה בוחר אותו אבל $A^*\varepsilon$ בחר בגלל הגמישות של f_score.

חלק י

סעיף א'

המדד הביצועי עבורו אלגוריתם IDA^* עדיף תמיד על פני אלגוריתם A^* הוא מדד הזיכרון.

אלגוריתם IDA^* משתמש בפונקציית f כמו ב-A^* אבל בחיפוש לעומק ולכן עם דרישות זיכרון נמוכות.

אלגוריתם IDA^* מבצע העמקה הדרגתית שמעלה בכל איטרציה את הגבלת פונקצית f.

סעיף ב'

1. פיתוחי צמתים

2. בכל איטרציה באלגוריתם IDA^* קיימת הגבלה על פונקציית ה-f ולא נזכור מה פיתחנו באיטרציות הקודמות, בניגוד ל-A^* הכולל רשימות של $open$ ו-$close$.

3. ראשית, נשים לב כי ייתכן שהאלגוריתמים IDA^* ו-A^* יפתחו צמתים שכבר פותחו בעבר. בנוסף, אנו יודעים כי $ID-DFS$ מפתח צמתים שכבר פותחו על מנת לחסוך בזכרון, לעומת BFS **שבכלל לא מפתח צמתים שכבר פותחו**. במקרה של A^* לעומת IDA^* ייתכן ששניהם מפתחים צמתים שכבר פותחו בכל מקרה ולכן המדד נפגע פחות מאשר מ-$ID-DFS$ לעומת BFS.

סעיף ג'

(i) - ראשית, נשים לב כי בכל איטרציה ההפרש בין $nextFlimit$ ל-$prevFlimit$ הוא לפחות $\frac{1}{k}$. לכן, כל שעלינו להעריך הוא את ההפרש בין $f-limit$ בהתחלה ו-f_limit בסוף (כאשר ה-f_limit בסוף הוא למעשה $Q_k\left(C^*_{\mathcal{S}}\right)$) ולבסוף נחלק בהפרש הגרוע ביותר שבין שני איטרציות עוקבות שהוא $\frac{1}{k}$, דהיינו: $\frac{\left[\left(Q_k(C^*_{\mathcal{S}})+\frac{1}{k}\right)-(Q_k(I))\right]}{\frac{1}{k}}=k\cdot Q_k\left(C^*_{\mathcal{S}}\right)+1-k\cdot(f-limit)$

(ii) - נסתכל באיטרציה אחת לפני האיטרציה שמצאנו את הפתרון. כעת, נשים לב כי $nextFLimit=\max\left\{Q_k\left(C^*_{\mathcal{S}}\right),Q_k\left(prevFlimit\right)+\frac{1}{k}\right\}$. מסעיף הקודם, אנו יודעים כי ההפרש בין $nextFlimit$ ל-$prevFlimit$ הוא לפחות $\frac{1}{k}$. לפי הגדרת $\varepsilon\left(\mathcal{A},\mathcal{S}\right)$ נסיק כי $\varepsilon\left(\mathcal{A},\mathcal{S}\right)\leq\frac{1}{k}$, כלומר זהו החסם על ההפרש בין $C^*_{\mathcal{S}}$ לעלות הפתרון שמצאנו.