

בינה מלאכותית תרגיל 1 להגשה רועי ג'הסי

Roei.gehasi@gmail.com

1. *שאלה יבשה*: פיתחו את הקבצים db/israel.csv ותארו את המבנה שלהם במדויק: מה מייצגת כל שורה ומה הפרמטרים בה. לצורך כך, עיינו בקוד של השגרה `load_map_from_csv` בקובץ הקוד המסופק ways/graph.py.

דוגמא לשורה בטבלה:

0,29.4412461,34.841524,1@32@2,906119@15@2

כעת מה משמעות כל פסיק (עמודה) בטבלה:

העמודה הראשונה מסמלת את הצומת הנכחית עליה שאר הפרטים יתייחסו.

העמודה השנייה מסמנת את קו הרוחב (הגאוגרפי).

העמודה השלישית מסמנת את קו האורך (הגאוגרפי).

מהעמודה הרביעית והלאה (לפי כמות הכבישים שיש מאותה צומת) – מסמנת את הכבישים שעוברים באותה הצומת.

כל עמודה שכזו מחולקת באופן הבא:

$1@32@2 \rightarrow$ separate each '@' this way:

$1 \rightarrow$ is the target

$32 \rightarrow$ is the distance

$2 \rightarrow$ is the highway type

8. אם כן, בחרנו בפונקציית יוריסטית כזו שמחשבת את המרחק האווירי מהקודקוד הנוכחי אל המטרה ומחלקת במהירות המירבית על פני כל הכבישים.

על מנת שהפונקציה היוריסטית h תהיה קבילה צריך להתקיים שלכל קודקוד שניקח ולכל פונקציית $h^*(n)$:

$$h(n) \leq h^*(n)$$

ראשית, בחלק הראשון של האלגוריתם בפונקצייה, לקחנו את המרחק האווירי וכבר ראינו בהרצאה כי פונק' יוריסטית המחשבת את המרחק האווירי, היא פונקצייה קבילה.

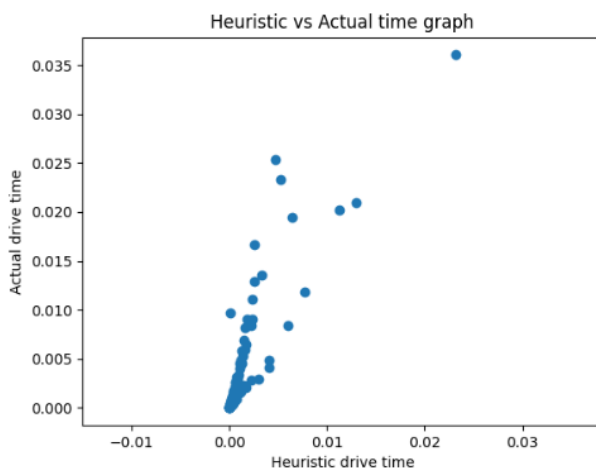
שנית, את המרחק האווירי חילקנו במהירות המירבית על פני כל הכבישים. כעת, מובטח לנו כי לכל מהירות אחרת שניקח, נקבל זמן קצר יותר כאשר נשתמש במהירות המירבית. (ניזכר כי החישוב הוא: $\frac{dist}{speed}$).

לכן, סה"כ הפונקצייה h שבנינו אכן קבילה כנדרש.

10. בהנחה שהרצתם את אלגוריתם A^* בדיוק לפני שהעומס בכביש התחיל, האם בהכרח המסלול המתקבל יהיה אופטימלי? הסבירו את קביעתכם. (התשובה לשאלה זו אינה תכנותית)

המסלול בהכרח יהיה אופטימלי. הסיבה לכך היא שאלגוריתם A^* הוא אופטימלי כאשר h עקבית. מכיוון שחישבנו את h לפי המרחק האווירי (לחלק למהירות המירבית), בהכרח נקבל כי h עקבית. (ראינו זאת בהרצאה). לכן, סה"כ כאשר נריץ את אלגוריתם A^* גם אם זה יהיה בדיוק לפני שהעומס בכביש התחיל, נקבל מסלול אופטימלי.

הציגו בדו"ח גרף ובו נקודה לכל אחת מההרצות הנ"ל. (ציר ה X מייצג את זמן הנסיעה היוריסטי, וציר ה Y מייצג את זמן הנסיעה בפועל) מה ניתן ללמוד מהגרף על הקשר בין המשתנים?



ניתן ללמוד כי ככל שהפונקצייה היוריסטית גדלה, כך גם הזמן האמיתי גדל.

לכן מתיאור הגרף ישנו קשר טוב בין גודל הפונקציה h לבין g .

13. לכל אחד מהאלגוריתמים (IDA^* , A^* , UCS), כתבו את זמן ריצת האלגוריתם הממוצע (ללא טעינת המפה). מי מהאלגוריתם רץ בזמן הקצר ביותר? מדוע?

ממוצע זמני ריצת האלגוריתמים:

```
ucs time : 0.00010
astar time : 0.00000
idastar time : 0.00433
```

כפי שניתן לראות, האלגוריתם הקצר ביותר הוא *astar* (שזה גם הגיוני לפי מה שראינו בהרצאה A^* הוא האלגוריתם המהיר ביותר).

הסיבה לכך ש A^* מהיר ביותר היא שהוא מפתח קודקודים לפי שילוב של פונקציית יוריסטית ופונקציית העבר (לפי מרחק קצר ביותר עד כה מקודקוד ההתחלה). לכן, בוודאי שהוא לפחות זהה ל *ucs*. הסיבה לכך שהוא טוב יותר היא שהוא מתחשב גם במרחק המשוערך אל המטרה ולפי זה הוא מפתח את שאר הקודקודים.

וודאי כי שני האלגוריתמים מהירים מ ida^* הואיל ואלגוריתם זה מפתח לפי גילוי חדש בכל פעם של קודקודים (בהתקדמות לפי *limits*).

לכן, במקרים בהם בכל איטרציה אנו מגלים קודקוד חדש בלבד, נצטרך לפתח שוב מההתחלה את כל ההיררכיה של אותו העומק שכבר העמקנו.

זאת הסיבה שזמן הריצה של אלגוריתם זה יהיה כבד יותר משאר האלגוריתמים.