

מעבדה בבינה מלאכותית
203.3630 ב.1
סמסטר ב' – שנה"ל תשפ"ה

מרצה: שי בושינסקי
shay@cs.haifa.ac.il

ניסוי מספר 3:

MULTI STEP HEURISTICS
META HEURISTICS SWARM INTELLIGENCE
CVRP אינטליגנציה הנחיל

מועד הגשה אחרון:

יום ו' 13 ליוני 2025

מרכיב הציון:

התרגיל הינו חובה

תנאי ההגשה:

העבודה וההגשה בזוגות (ניתן כמובן להגיש ביחידים)

המשימה:

סקרנו בהרצאה multi-step heuristics ואוסף של meta heuristics : Simulated Annealing, Tabu Search, Ant Colony שאותן הכללנו בתוך חיפוש אבסטרקטי מקומי iterative local search. משימתכם הינה להשתמש בהן באופן זה על-מנת לפתור הבעיה הבאה. לצורך פתרונה נדרשת מידה של יצירתיות במימוש ובבחירת הפרמטריזציה של האלגוריתמים הנ"ל.

בעיית ה-CVRP:

ניתוב רכבים (CAPACITATED VEHICLE ROUTING PROBLEM) היא הבעיה הבאה:
נתון צי של רכבים בעלי תכולה קבועה שצריכים לספק ביקוש בערים נתונות הפרושות על מפה דו-מימדית.

עליכם למצוא לצי הקצאת מסלולים אופטימלית (אין צורך להשתמש בכל הרכבים) כך שכל רכב יצא מהמחסן, יבקר בכל עיר פעם אחת וישוב למחסן וכך שבתום הביקורים כל הביקושים יסופקו. פתרון יחשב אופטימלי אם לא ניתן לבצע את הנ"ל במסלול כולל קצר יותר.

מדובר בבעיה פרקטית השייכת למחלקת הבעיות שהן NP-COMPLETE. להלן ההגדרה הפורמאלית של הבעיה:

► n Locations, v Vehicles

► For each location,

– demand d_i and location x_i, y_i

► The capacity of the vehicles c

► The sequence of deliveries of vehicle i , T_i

$$\text{minimize: } \sum_{i \in V} \left(\text{dist}(0, T_{i,0}) + \sum_{\langle j,k \rangle \in T_i} \text{dist}(j, k) + \text{dist}(T_{i,|T_i|-1}, 0) \right)$$

subject to:

$$\sum_{j \in T_i} d_j \leq c \quad (i \in V)$$

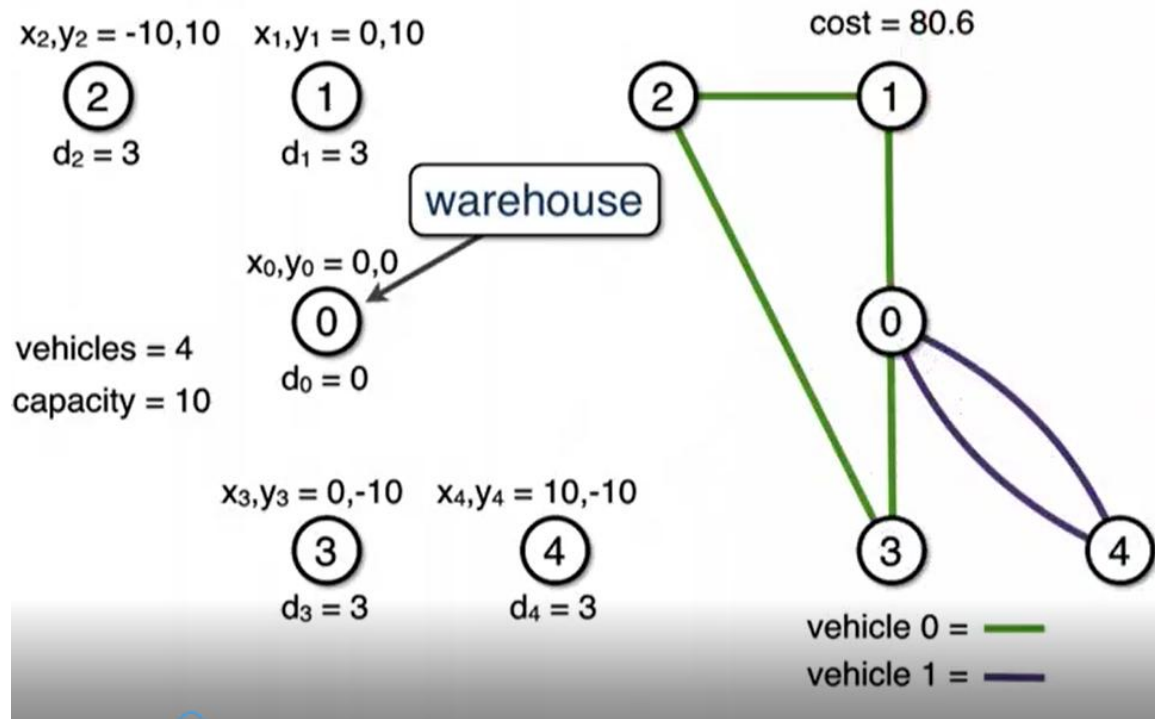
$$\sum_{i \in V} (j \in T_i) = 1 \quad (j \in N \setminus \{0\})$$

לדוגמא: בהנתן קואורדינטות המחסן $(0,0)$, קואורדינטות הערים מסומנות ב x,y הביקוש בכל

עיר d מציאת מסלול, (תת-אופטימלי), ל 4 רכבים כשתכולת כל רכב היא 10 בעזרת 2 רכבים בלבד:

(a) הרכב הראשון יוצא מהמחסן מבקר בערים 1,2,3 וחוזר למחסן

(b) הרכב השני יוצא מהמחסן מבקר בעיר 4 וחוזר



עלות המסלול מחושבת: $\sqrt{10^2+20^2}+2*\sqrt{10^2+10^2}+10+10+10$
80.64=

עליכם לפתח תוכנה שתקבל כקלט בעית CVRP ושתחזיר את המסלולים האופטימלים עבור צי הרכבים הנתונים תחת אילוצי הבעיה.

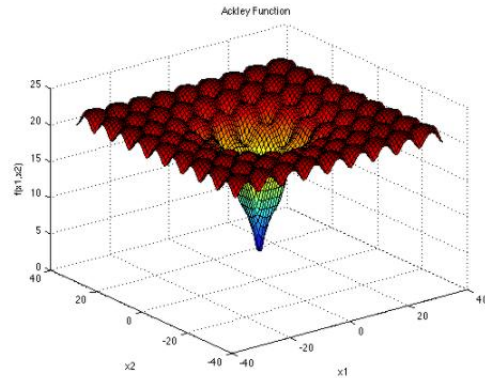
עליכם להשוות בין חמישה אלגוריתמי חיפוש מבינה מלאכותית:

1. תכננו לפחות היוריסטיקה אחת שהיא multi-stage heuristic. הסבירו ונמקו את השיקולים מאחורי ההיוריסטיקה שבחרתם וכן ציינו מה הסיבוכיות של ההיוריסטיקה.
2. ממשו חיפוש לוקאלי איטרטיבי ILS המבוסס על מטה-היוריסטיקות:
 - a. TABU SEARCH להלן TS
 - b. אלגוריתם נחיל: DISCRETE ANT COLONY OPTIMIZATION להלן ACO
 - c. אלגוריתם SIMULATED ANNEALING
3. האלגוריתם הגנטי GA עם ISLAND MODEL שפותח במעבדה הקודמות (אפשר ממטי)
4. אלגוריתם ALNS
5. אלגוריתם Branch and Bound מבוסס היוריסטיקה כדוגמת LDS

6. לבדיקה האלגוריתמים עליכם תחילה למצוא באמצעותם מינימום גלובלי לפונקציה

Ackley

המתוארת להלן (לפי ההיריסטיקה הטבעית):



$$f(\mathbf{x}) = -a \exp \left(-b \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2} \right) - \exp \left(\frac{1}{d} \sum_{i=1}^d \cos(cx_i) \right) + a + \exp(1)$$

תחת ההגדרות הבאות:

- Number of dimensions, $d = 10$
- $X_i \in [-32.768, 32.768]$, for all $i = 1, \dots, d$
- $a = 20$, $b = 0.2$ and $c = 2\pi$
- $f(X^*) = 0.0$ at $X^* = (0.0, \dots, 0.0)$

CVRP: טיפול בקלט/פלט, ניתוח הבעיה ובניית אב-טיפוס לפתרון:

1. לימדו את פורמט הקלט/פלט – (בעיה/פתרון) המוסברים בסעיפים בהמשך
2. התאימו את בעיית הדוגמא לעיל לפורמט הקלט ולכל אלגוריתם שאתם נדרשים לפתח
3. עבור האלגוריתמים השונים פתחו היוריסטיקות שונות שיכולות לסייע בפתרון – מותר כמובן לעשות שימוש בכל ההיוריסטיקות שנלמדו בקורס המבוא עבור בעיות TSP וכן בהיוריסטיקות ספציפיות ל-CVRP שנלמדו בקורס המעבדה.
4. קדדו את האלגוריתמים (ראו סעיף לגבי האלגוריתמים הספציפיים)

5. עבור כל היוריסטיקה בה אתם עושים שימוש הסבירו את יתרונותיה ביחס לבעיית

CVRP

6. שלבו את ההיוריסטיקות באלגוריתמים המתאימים, בצעו "בדיקת שפיות" על הדוגמא

לעיל תוך בחינת ההיוריסטיקות השונות – ותוך כמה זמן. הדפיסו את הפתרון עצמו לפי

מטריצת הפלט ובנוסף את זמני הריצה (CPU ELAPSED).

7. הציגו באמצעות העזרים הויזואלים כמו שהוצגו במעבדה את ביצועי האלגוריתם

הקלט: [/http://vrp.atd-lab.inf.puc-rio.br/index.php/en](http://vrp.atd-lab.inf.puc-rio.br/index.php/en)

רשימת בעיות דוגמא ניתן למצוא באתר ת-Vehicle Routing Data Sets: יש לחלץ את את מס' הרכבים,

מס' הצמתים, תכולת כל רכב, רשימת הקואורדינטות של כל צמת ורשימת הביקושים של כל צמת

בהתאמה. להלן רשימת הבעיות הנדרשות (בכללן יש פתרון משוער)

1. Beginner ($n \leq 30$)

- **P-n16-k8** (Set P)
 - 16 customers, capacity 100, uniform grid + $U(1,30)$ demands
 - Can be solved to optimality in seconds, so students can verify their code.
- **E-n22-k4** (Set E)
 - 22 customers, capacity ≈ 100 , classical Christofides/Eilon instance
 - Introduces slightly non-integer distances and more clustered geography.

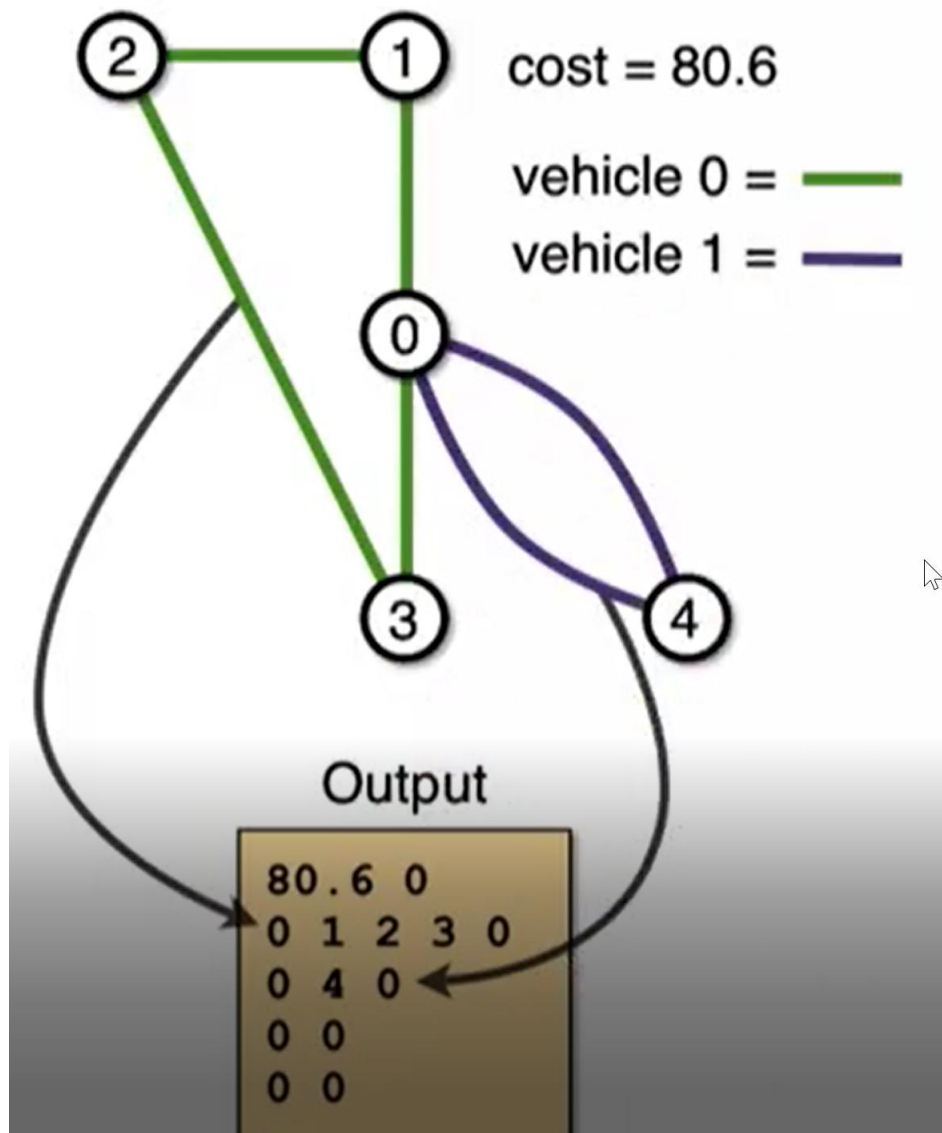
2. Intermediate ($30 < n \leq 80$)

- **A-n32-k5** (Set A)
 - 32 customers, uniform Euclidean, a well-studied "random" benchmark.
- **B-n45-k9** (Set B)
 - 45 customers, clustered locations—students will see the impact of clustering on route structure.
- **A-n80-k10** (Set A)
 - 80 customers, uniform; known optimum 1 763, so they can gauge optimality gaps.

3. Advanced ($n > 80$)

- **X-n101-k25** (Set X)
 - 101 customers on a 0–1000 grid, mixed/random cluster layout, varied demands.
 - Float distances, non-trivial geographic patterns—great for modern metaheuristics.
- **M-n200-k17** (Set M)

- 200 customers, classic “hard” case;



הפלט:

הערות:

1. בתום ריצת כל אלגוריתם, על הפותר להחזיר את ערך הפונקציה ההיוריסטית ואח"כ עבור רכב שנעשה בו שימוש את רשימת אינדקסי הערים בה ביקר לפי הסדר (יש להתחיל באינדקס 0 - המחסן ולסיים באינדקס 0 - המחסן).
2. מאחר שהאלגוריתמים סטוכסטים במהותם - הקפידו לבצע מספיק ריצות כדי להסיק מסקנות מגובות סטטיסטית לצורך השוואה ראוייה לגבי המדדים.
3. שימו לב שאלגוריתמי META HEURISTICS ניתנים בקלות יחסית למקבול ובכך מאפשרים האצה והעשרה של הפתרונות.
4. בחלק מהאלגוריתמים מומלץ לאתחל בפתרון GREEDY ומשם להתקדם.
5. ACO אתם שולטים על ההיורסטיקה ("η") ועל הפרמטרים השונים של האלגוריתם וגם על מהלך העדכון וטכניקות החלצות מאופטימום לוקאלי.
6. TS אתם מגדירים מהי הסביבה בכל צעד - למשל TS אילו קשתות רלוונטיות. כמו-כן לוויסות פרמטרי הקשיחות של האלגוריתמים SA ב T n ב TS אלפא וביתא ACO וכווננם דינאמית בהתאם למשל לערכי הפונקציה ההיוריסטית.
7. ALNS אתם נדרשים להגדיר סביבה באמצעות אופרטורים של הרס ותיקון כמו ב TTP (לדוגמא 2-OPT). עליכם לכוון את הפרמטרים של אלגוריתם מטרופוליס (M SIMULATED ANNEALING) לקבלה של פתרונות מקלקלים.

ההגשה: במתכונת הסטנדרטים של המעבדה

יש להגיש דו"ח מסודר הכולל:

- א. תוכנת מקור SOURCE - מימוש הנ"ל בשפת תכנות לבחירתך מתועדת
- ב. תוכנות ריצה מתאימות EXE
- ג. מסמך המסכם את תוצאות העבודה