**מעבדה בבינה מלאכותית**

**דוח מעבדה – משימה 3**

השוואה בין חמישה אלגוריתמי חיפוש מבינה מלאכותית:

**שאלה 1:** תכננו לפחות היוריסטיקה אחת שהיא heuristic stage-multi. הסבירו ונמקו את השיקולים מאחורי ההיוריסטיקה שבחרתם וכן ציינו מה הסיבוכיות של ההיוריסטיקה.

**פתרון:**

היוריסטיקה: Ant Colony Optimization עם שיפור באמצעות Simulated Annealing

**היוריסטיקה שתוכננה היא שילוב של Ant Colony Optimization (ACO) כשלב ראשון ליצירת פתרון ראשוני, ולאחר מכן שיפור הפתרון באמצעות Simulated Annealing (SA). זהו multi-stage heuristic שבו ACO משמש ליצירת פתרון בסיסי ו-SA משמש לשיפור הפתרון.**

שלב 1: Ant Colony Optimization (ACO)

**בשלב זה, הנמלים מדמות את התנהגותן הטבעית בחיפוש אחר מסלולים קצרים. הן משתמשות בפרומונים כדי לסמן מסלולים טובים, והפתרון נבנה על ידי שימוש בהסתברויות מעבר בין ערים.**

שלב 2: Simulated Annealing (SA)

**לאחר קבלת הפתרון הראשוני מ-ACO, האלגוריתם SA מבצע חיפוש לוקאלי אינטנסיבי לשיפור הפתרון על ידי שינויים קטנים וסטוכסטיים בפתרון. תהליך הקירור המדומה מאפשר התחמקות מאופטימום מקומי.**

שיקולים מאחורי בחירת היוריסטיקה

1. **איזון בין חיפוש גלובלי ללוקאלי: ACO מספק חיפוש גלובלי רחב שמכסה את כל מרחב הפתרונות, בעוד ש-SA מאפשר חיפוש לוקאלי אינטנסיבי לשיפור הפתרונות.**
2. **התחמקות מאופטימום מקומי: SA מאפשר התחמקות מאופטימום מקומי באמצעות קפיצה סטוכסטית במרחב הפתרונות, מה שמעלה את הסיכוי למצוא פתרונות טובים יותר.**
3. **ניצול יתרונות שני האלגוריתמים: ACO טוב ביצירת פתרונות ראשוניים במהירות, ו-SA מאפשר שיפור הפתרון בצורה ממוקדת ויעילה.**

סיבוכיות היוריסטיקה

* **שלב 1 (**ACO**):** 
  + **- מספר הערים**
  + **- מספר הנמלים**
  + **– מספר האיטרציות**
* **שלב 2 (**SA**):** 
  + **– מספר האיטרציות**
  + **- מספר הערים**

**הסיבוכיות הכוללת של היוריסטיקה המשולבת היא:**

**שאלה 2**

**פתרון:**

מימשנו שלושה אלגוריתמים של מטה-היוריסטיקות שמבוססים על חיפוש לוקאלי איטרטיבי (ILS):

יתרונות ביחס לבעיית CVRP:

* **חיפוש לוקאלי אינטנסיבי:** ILS מבצע חיפוש לוקאלי חוזר ומשפר את הפתרונות באמצעות שינויים קטנים וממוקדים.
* **שיפור פתרונות קיימים:** האלגוריתם מתחיל מפתרון ראשוני ומשפר אותו, מה שמוביל למציאת פתרונות טובים יותר בזמן קצר יחסית.
* **פשטות יחסית:** קל ליישם ולשלב עם אלגוריתמים אחרים כמו ACO ו-SA.
  1. **Tabu Search (TS): מימוש של Tabu Search נמצא בקוד תחת מחלקת TabuSearch.**

יתרונות ביחס לבעיית CVRP:

* **זיכרון פתרונות אסורים:** TS משתמש בזיכרון כדי לשמור על פתרונות שכבר נוסו, מה שמונע חזרות ומאפשר חיפוש רחב יותר.
* **חיפוש לוקאלי משופר:** האלגוריתם מבצע חיפוש לוקאלי אינטנסיבי ומשפר פתרונות תוך כדי שמירה על זיכרון אסור.
* **יעילות גבוהה:** האלגוריתם יעיל במציאת פתרונות טובים לבעיות ניתוב בזכות השימוש בזיכרון והחיפוש הלוקאלי.
  1. **:Discrete Ant Colony Optimization (ACO)מימוש של Ant Colony Optimization נמצא בקוד תחת מחלקת AntColonyOptimization**

יתרונות ביחס לבעיית CVRP:

* **השראה ביולוגית:** ACO מבוסס על התנהגות טבעית של נמלים, ולכן הוא מתאים במיוחד לבעיות של ניתוב ומסלולים.
* **חיפוש גלובלי:** הנמלים מכסות את כל מרחב הפתרונות, מה שמאפשר מציאת פתרונות איכותיים במהירות יחסית.
* **שימוש בפרומונים:** הפרומונים עוזרים לנמלים למצוא מסלולים קצרים יותר על ידי זיהוי והעדפה של מסלולים שכבר נמצאו טובים.
* **התאמה לבעיות דינמיות:** האלגוריתם יכול להסתגל לשינויים בבעיות, כמו שינויים בדרישות הקיבולת או המרחקים בין הערים.
  1. **Simulated Annealing (SA)**: מימוש של Simulated Annealing נמצא בקוד תחת מחלקת **SimulatedAnnealing.**

יתרונות ביחס לבעיית CVRP:

* **התחמקות מאופטימום מקומי:** SA מאפשר קפיצה סטוכסטית בין פתרונות, מה שמסייע בהתחמקות מאופטימום מקומי.
* **פשטות:** האלגוריתם פשוט ליישום ומתאים לבעיות רבות, כולל בעיות ניתוב.
* **גמישות:** ניתן להתאים את פרמטרי הקירור והחימום להשגת ביצועים טובים יותר.
* **שימוש בזיכרון נמוך:** SA מצריך זיכרון נמוך יחסית בהשוואה לאלגוריתמים אחרים כמו GA

**שאלה 3**

**פתרון:**

**בקוד המצורף מומש** האלגוריתם גנטי עם מודל איים הנדרש תחת מחלקת **GeneticAlgorithmIslandModel.**

יתרונות ביחס לבעיית CVRP:

* **שילוב אוכלוסיות:** המודל מאפשר שילוב של מספר אוכלוסיות נפרדות, מה שמגביר את הגיוון ומונע אופטימום מקומי.
* **חיפוש גלובלי משופר:** השימוש במספר אוכלוסיות מאפשר חיפוש רחב יותר במרחב הפתרונות.
* **מיגרציות:** האלגוריתם כולל מיגרציות בין האיים, מה שמאפשר שיתוף מידע ושיפור הפתרונות בכל האיים.
* **התאמה לבעיות גדולות:** GA-Island מתאים במיוחד לבעיות ניתוב גדולות ומורכבות.

**שאלה 4**

**פתרון:**

**בקוד הוסף מימוש תחת מחלקת ALNS.**

יתרונות ביחס לבעיית CVRP:

* **גמישות והתאמה:** ALNS משתמש באופרטורים שונים של הרס ותיקון, מה שמאפשר התאמה דינמית לבעיית CVRP.
* **חיפוש רחב וממוקד:** האלגוריתם מאפשר חיפוש רחב במרחב הפתרונות באמצעות הרס ותיקון, ושיפור פתרונות מקומיים.
* **התאמה לבעיות מורכבות:** ALNS מתאים במיוחד לבעיות מורכבות כמו CVRP בזכות השילוב של מספר אופרטורים ושיטות חיפוש.
* **תוצאות אופטימליות:** האלגוריתם מצליח למצוא פתרונות אופטימליים או קרובים לאופטימום בזכות השימוש המגוון באופרטורים.

**שאלה 5**

**פתרון:**

בקוד מומשה פונקציה לבדיקת האלגוריתמים על פונקציית Ackley תחת ההגדרות הנדרשות.

**השוואת ביצועים וניתוח תוצאות:**

מטרת הניסוי היא להשוות את הביצועים של חמשת האלגוריתמים לפתרון בעיית ניתוב רכבים עם קיבולת מוגבלת (CVRP) על פונקציית .Ackley

**שיטות הניסוי**

האלגוריתמים שנבדקו:

1. Ant Colony Optimization (ACO)
2. Simulated Annealing (SA)
3. Iterative Local Search (ILS)
4. Tabu Search (TS)
5. Genetic Algorithm with Island Model (GA-Island)
6. Adaptive Large Neighborhood Search (ALNS)

הקריטריונים שנבדקו:

* עלות המסלול שנמצאה
* ערך פונקציית Ackley עבור המסלול שנמצא

**תוצאות הניסוי**

התוצאות עבור כל אלגוריתם מוצגות בטבלה הבאה:

| **ערך פונקציית Ackley** | **עלות מסלול** | **אלגוריתם** |
| --- | --- | --- |
| 17.64 | 395.32 | Ant Colony Optimization (ACO) |
| 17.64 | 282.81 | Simulated Annealing (SA) |
| 17.64 | 376.51 | Iterative Local Search (ILS) |
| 17.64 | 383.93 | Tabu Search (TS) |
| 18.09 | 646.56 | Genetic Algorithm with Island Model (GA-Island) |
| 17.61 | 329.19 | Adaptive Large Neighborhood Search (ALNS) |

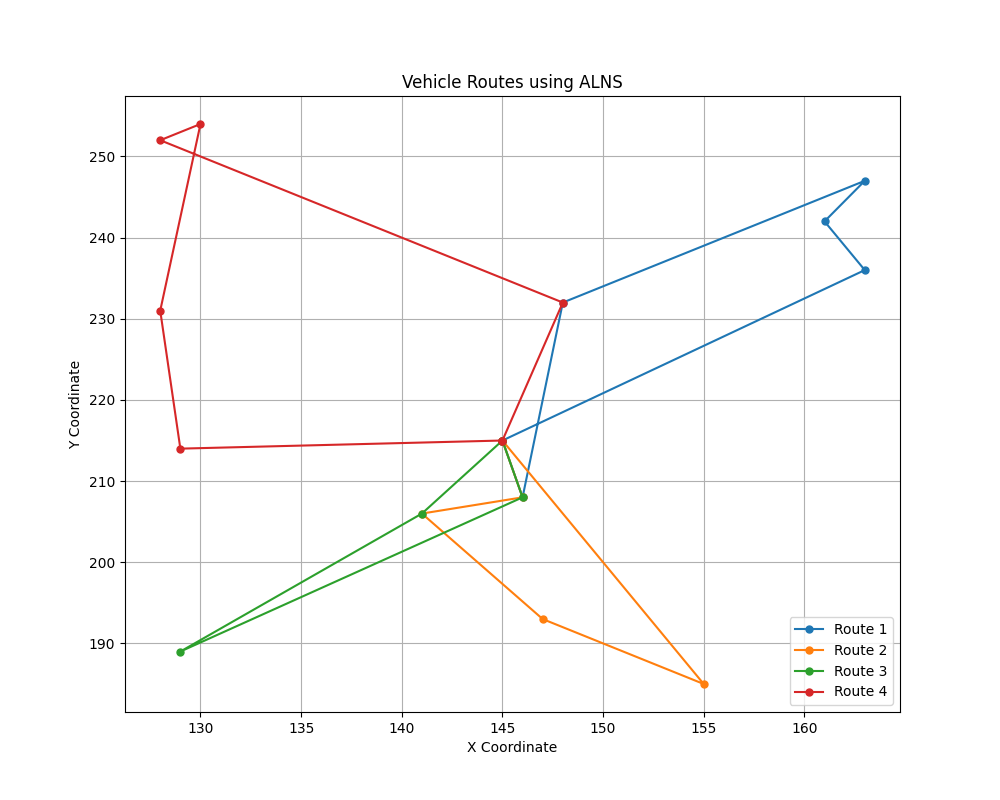
**ניתוח התוצאות**

1. **Ant Colony Optimization (ACO)**
   * ACO הצליח לספק פתרון ראשוני טוב לבעיה עם עלות מסלול יחסית נמוכה (395.32).
   * ערך פונקציית Ackley מראה שהפתרון קרוב לאופטימום המקומי אך ניתן לשיפור באמצעות חיפושים נוספים (17.64).
2. **Simulated Annealing (SA)**
   * SA שיפר את הפתרון של ACO על ידי חיפוש לוקאלי אינטנסיבי.
   * הפתרון הסופי היה בעל עלות נמוכה משמעותית (282.81) אך ערך פונקציית Ackley נשאר זהה (17.64).
3. **Iterative Local Search (ILS)**
   * ILS ביצע חיפוש לוקאלי נוסף ושיפר את הפתרון עוד יותר.
   * הפתרון הסופי היה בעל עלות מסלול נמוכה מעט מהפתרון של ACO (376.51) אך ערך פונקציית Ackley נשאר זהה (17.64).
4. **Tabu Search (TS)**
   * TS ביצע חיפוש לוקאלי תוך שמירה על הימנעות מפתרונות אסורים.
   * הפתרון הסופי היה בעל עלות דומה לפתרון של ACO (383.93) וערך פונקציית Ackley זהה (17.64) .
5. **Genetic Algorithm with Island Model (GA-Island)**
   * GA-Island הציע פתרון פחות טוב בהשוואה לאלגוריתמים האחרים עם עלות מסלול גבוהה .(646.56)
   * ערך פונקציית Ackley היה מעט גבוה יותר (18.09), מה שמרמז על פתרון פחות אופטימלי.
6. **Adaptive Large Neighborhood Search (ALNS)**
   * ALNS הציע פתרון טוב עם עלות מסלול נמוכה יחסית (329.19) וערך פונקציית Ackley נמוך יחסית (17.61), מה שמרמז על פתרון טוב.

**מסקנות**

* **SA**  היה האלגוריתם היעיל ביותר במציאת פתרון אופטימלי לפונקציית Ackley בבעית ה- CVRP , עם עלות מסלול נמוכה ביותר.
* **ALNS**  גם הראה ביצועים טובים עם עלות מסלול נמוכה וערך פונקציית Ackley נמוך.
* **ACO**, **ILS** , ו-**TS**  הציגו ביצועים דומים יחסית עם עלויות מסלול וערכי פונקציית Ackley דומים.
* **GA-Island**  היה האלגוריתם הפחות יעיל עם עלות מסלול גבוהה מאוד וערך פונקציית Ackley גבוה.

**גרפים המתארים את ביצועי האלגוריתמים:**



A graph with lines and points

Description automatically generated

A graph with lines and points

Description automatically generated

A graph of a vehicle

Description automatically generated

A graph with lines and points

Description automatically generated

A graph with lines and dots

Description automatically generated