**דוח מעבדה 1 – חלק ב':**

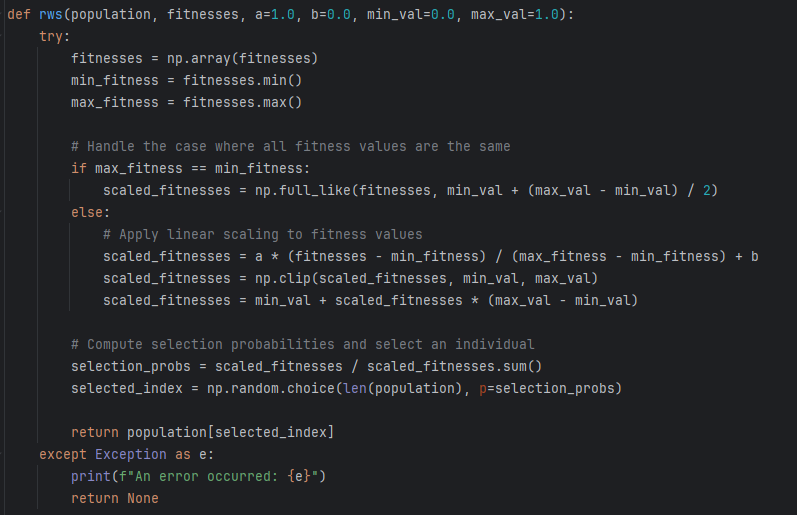
**שאלה 1:**

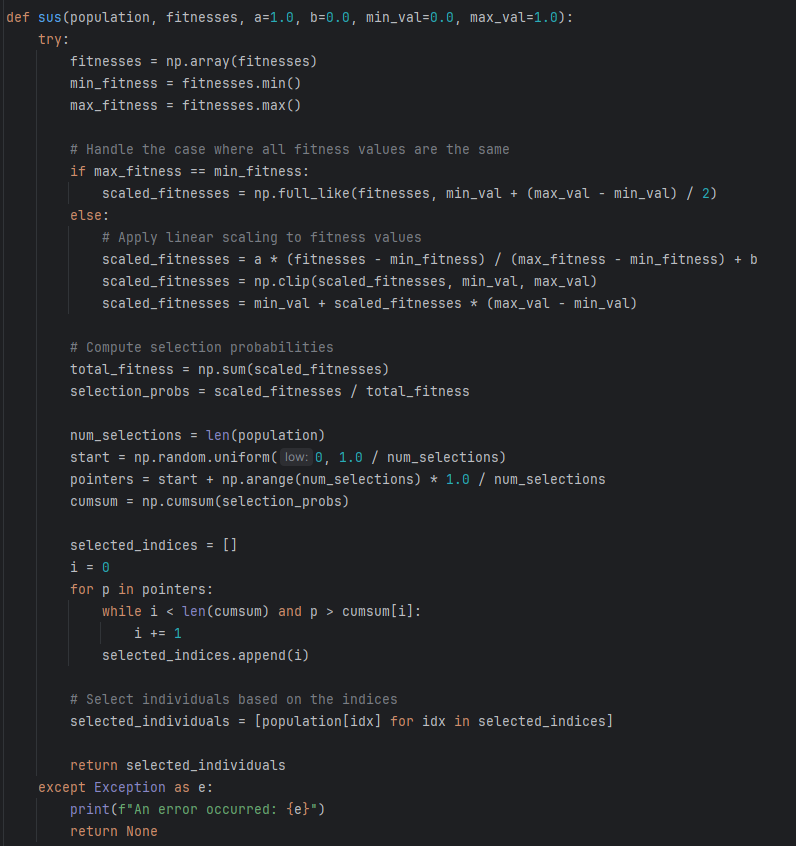
הוסיפו למנוע תמיכה בשיטות הבחירה שונות: parent selection

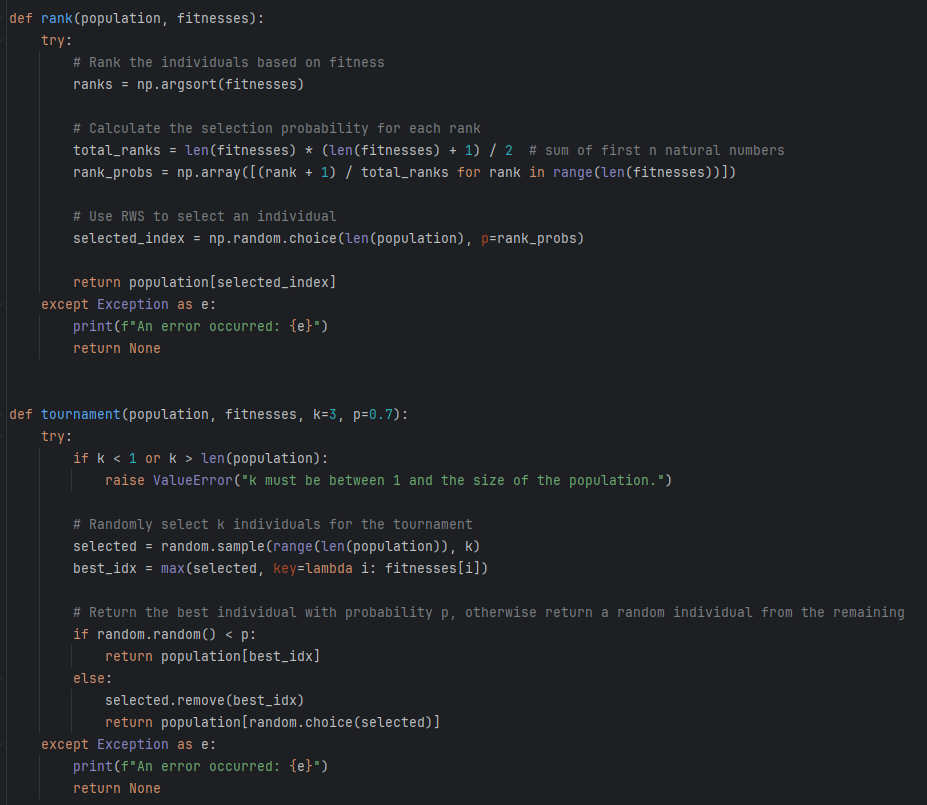
1. RWS + Linear Scaling
2. SUS + Linear Scaling
3. RWS + RANKING
4. טורניר לא דטרמיניסטי עם פרמטריםK P

**פתרון:**

לשאלה זאת ולסעיפים אלו נכתבו הפונקציות הבאות:



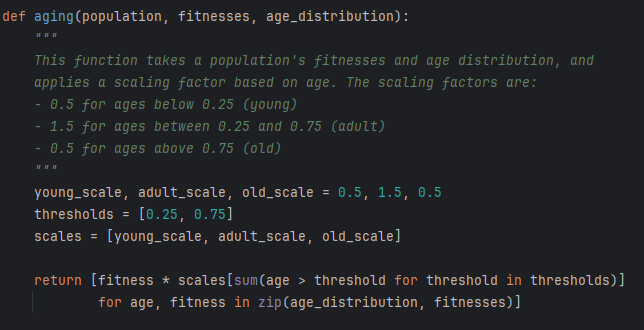




**שאלה 2:**

הוסיפו שיטת שרידות נוספת (Aging).

**פתרון:**



**שאלה 3:**

הוסיפו תמיכה למופע חדש של בעיה – בעיית sudoku.

חידת סודוקו פרושה על גריד 9x9 המחולק ל-9 תת גרידים של 3x3, המטרה היא למלא את הgrid כך שלכל שורה, טור ותת-grid יכילו את כל הספרות 1-9.

אפשרות למצוא דוגמאות לחידות ופתרונותיהן ב: <https://sandiway.arizona.edu/sudoku/examples.html>

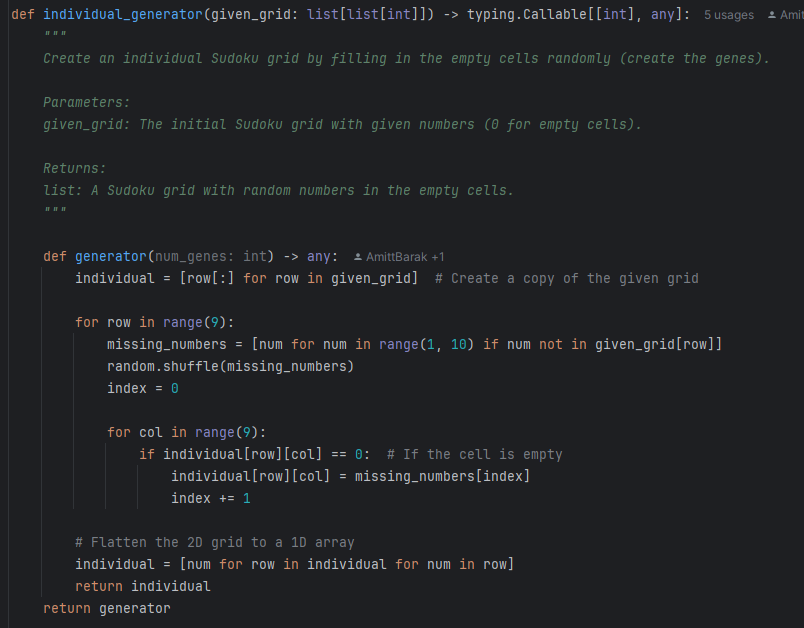
לצורך כך:

1. ממשו ייצוג מתאים לגן באורך N.
2. ממשו 2 אופרטורי שיחלוף לתמורות: PMX,CX.
3. ממשו 2 מוטציות חלופיות לתמורות: היפוך וערבול.
4. בחרו פונקציה פיטנס יעילה ככל הניתן -נמקו את בחירתכם.
5. על המנוע לפתור את כל החידות באתר הנ"ל.

**פתרון:**

1. **מימוש של ייצוג מתאים לגן באורך N:**

מימשנו ייצוג מתאים לגן באורך N ע"י מימוש הפונקציה הבאה:

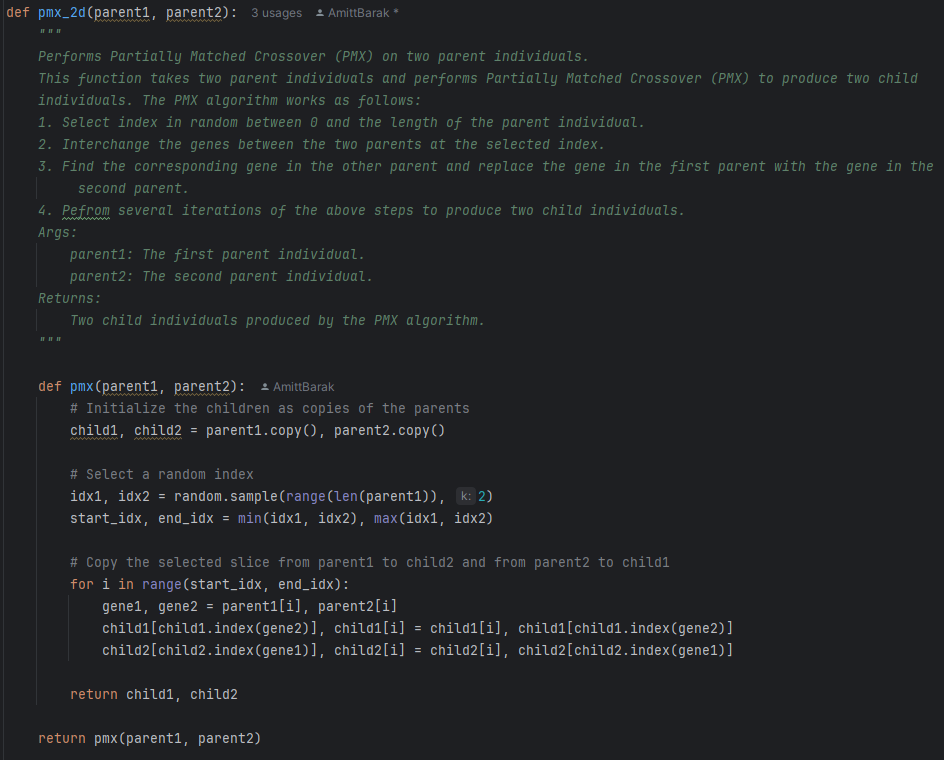


**הסבר:** בפונקציה מימשנו ייצוג מתאים לגן באורךN שבו Nהוא מספר התאים בלוח הסודוקו, כלומר 81 תאים בלוח על ידי השטחת הלוח הדו-ממדי לרשימה חד-ממדית באורך 81.

הלוח הראשוני ניתן כרשימה של רשימות (לוח 9x9) שבו 0 מייצג תא ריק ומספרים מ-1 עד 9 מייצגים תאים מלאים. עוברים על כל תא בלוח וממלאים את התאים הריקים במספרים אקראיים מתוך המספרים החסרים באותה השורה.

לאחר מילוי כל התאים הריקים, ממירים את הלוח הדו-ממדי לרשימה חד-ממדית באורך 81 על ידי איחוד כל השורות לרשימה אחת ארוכה – זהו הייצוג המתאים לגן באורך N שהפונקציה לבסוף מחזירה.

1. **מימוש אופרטור שחלוף :PMX**



**הסבר:** בתחילת התהליך יצרנו עותקים של ההורים שישמשו כבסיס לצאצאים.

לאחר מכן נבחרו שני אינדקסים אקראיים בטווח האורך של הפרט ההורי. אינדקסים אלו קובעים את הקטע של הגנים שיוחלפו בין ההורים.

הקטע שנבחר (בין האינדקסים האקראיים) מועתק מהורה אחד לצאצא השני ולהיפך.

שימור התקינות הגנטית: בלולאה שבודקת את הגנים המוחלפים, התבצעה החלפה של הגנים המוכפלים בצאצאים. כך, אם גן מסוים כבר קיים בקטע שהועתק לצאצא, מוחלפת המופע הנוסף שלו בצאצא עם הגן שמצוי במקום המתאים בהורה המקורי.

**מימוש אופרטור שחלוף CX:**

A computer screen shot of a program code

Description automatically generated

A computer screen shot of a program

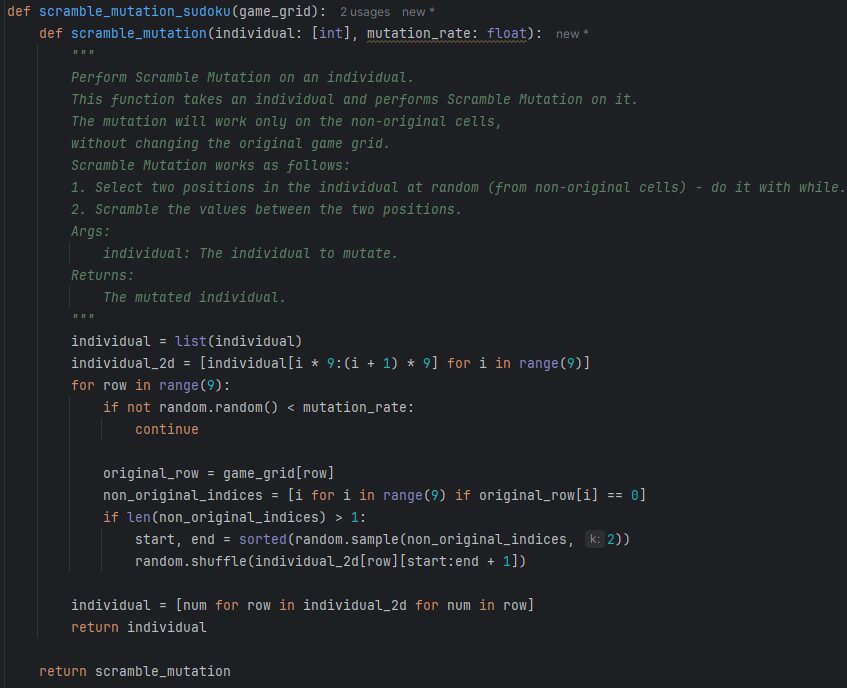
Description automatically generated

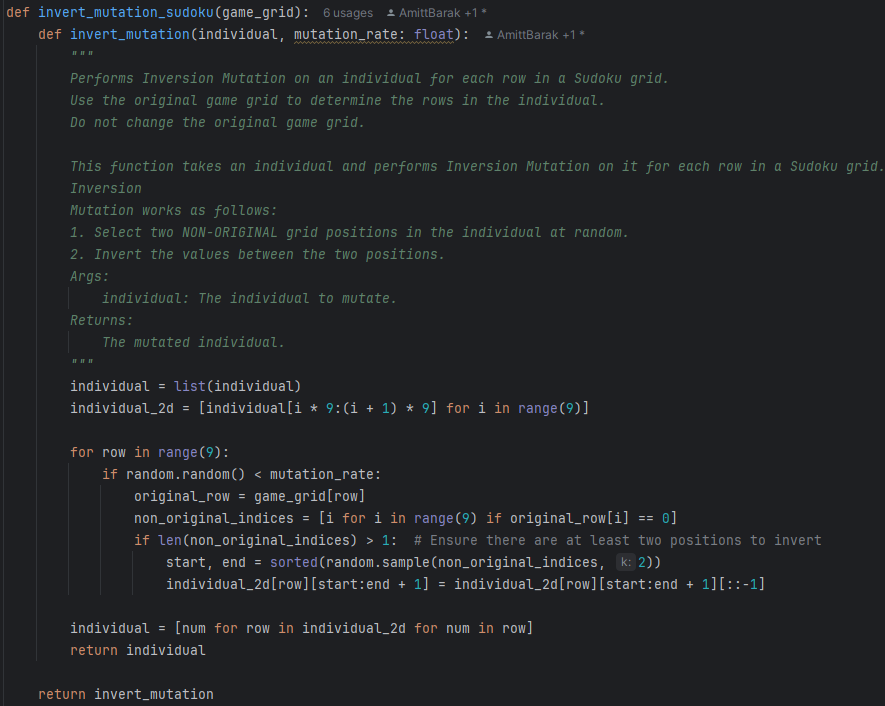
**הסבר:** מוודאים שההורים הם פרמוטציות של אותו סט אלמנטים. יוצרים צאצאים ועוקבים אחרי מיקומים שביקרנו בהם. מזהים מחזורים בהורים ומבצעים החלפה לפי מחזורים.

(פונקציית cycle\_crossover\_2d מתאימה את ביצוע ההחלפות על לוחות סודוקו).

1. **ממשו 2 מוטציות חלופיות לתמורות - היפוך וערבול:**

פונקציית ערבול:



פונקציית היפוך:  


1. פונקציית הפיטנס: נועדה להעריך את איכות הפתרון של לוח סודוקו על ידי מתן ציון לכל פרט. הציון מבוסס על ייחודיות המספרים בשורות, בעמודות ובתתי-הרשתות של לוח הסודוקו, וכן על התאמה לערכים המקוריים בלוח הנתון.

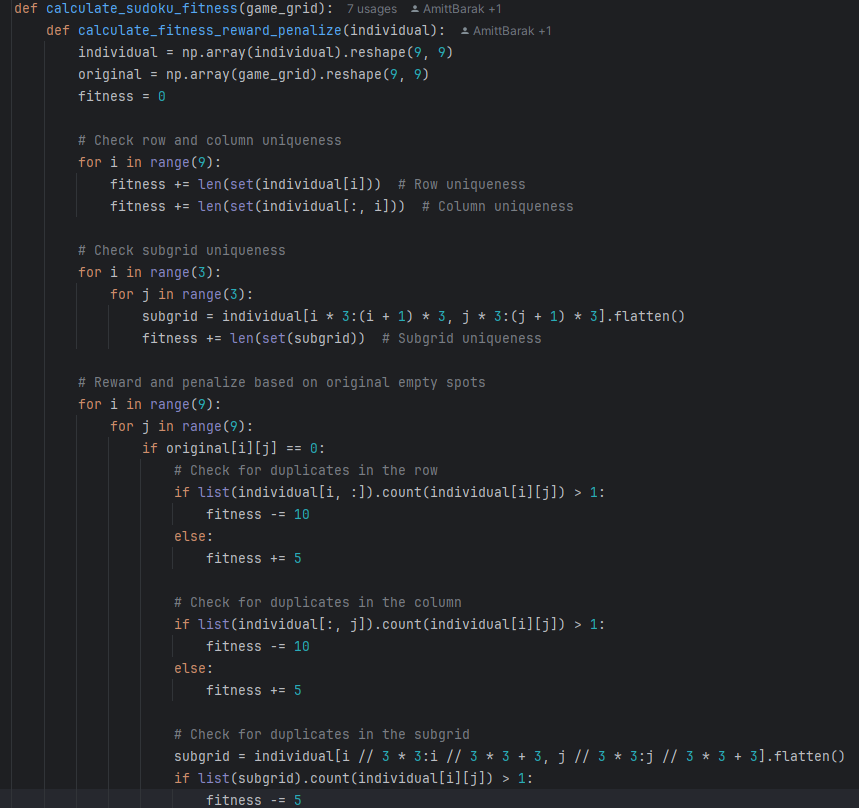
* עוברים על כל שורה ועמודה ובודקים את מספר הערכים הייחודיים. ככל שיש יותר ערכים ייחודיים, הציון עולה.בדיקת ייחודיות בתתי-רשתות:
* עוברים על כל תת-רשת (בלוק 3x3) ובודקים את מספר הערכים הייחודיים. ככל שיש יותר ערכים ייחודיים, הציון עולה.

מתן בונוסים ועונשים:

**בונוסים:** עבור תאים ריקים בלוח המקורי, מוסיפים בונוס אם המספר לא מופיע פעמיים באותה שורה, עמודה או תת-רשת.

**עונשים:** אם המספר מופיע פעמיים, מפחיתים נקודות.אם התא לא היה ריק במקור והמספר שונה מהמקור, מפחיתים נקודות.

**מימוש:**

A screen shot of a computer

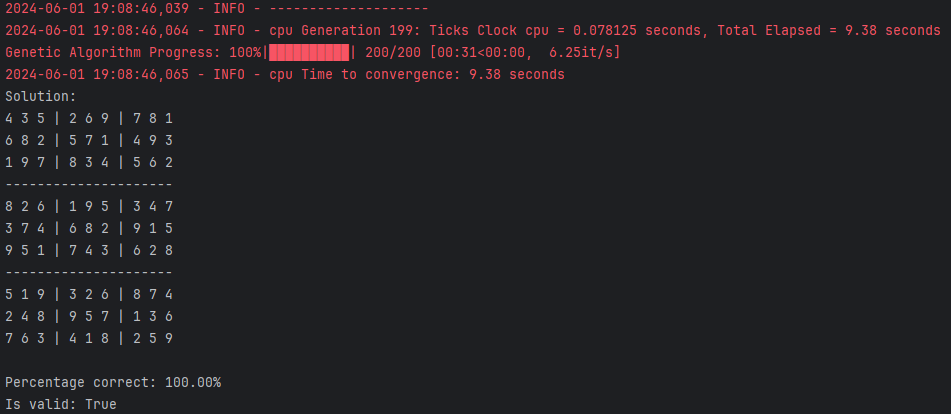
Description automatically generated

**נימוקים לבחירת פונקציית הפיטנס:**

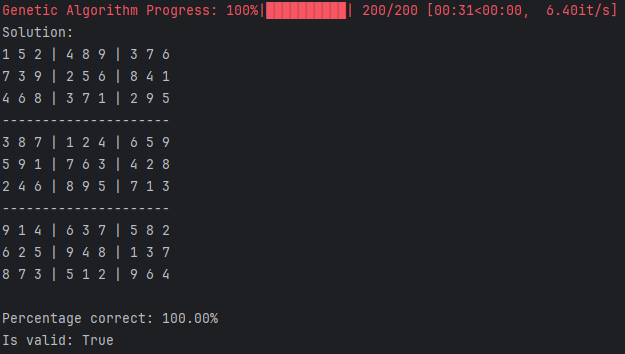
פונקציית הפיטנס שנבחרה עבור פתרון בעיית הסודוקו נועדה להעריך את איכות הפתרון בצורה שמבטיחה את קיום החוקים הבסיסיים של סודוקו: ייחודיות המספרים בכל שורה, עמודה ותת-רשת בלוק. **מדוע בחרנו בפונקציה זו:**

1. בדיקת ייחודיות בשורות ובעמודות: כל שורה וכל עמודה בלוח הסודוקו צריכה להכיל את כל המספרים מ-1 עד 9 ללא כפילויות. הפונקציה מחשבת את מספר הערכים הייחודיים בכל שורה ועמודה ומוסיפה אותם לציון הפיטנס. ככל שיש יותר ערכים ייחודיים, כך הציון גבוה יותר.
2. בדיקת ייחודיות בתתי-רשתות (בלוקים של 3x3): כל תת-רשת בגודל צריכה להכיל את כל המספרים מ-1 עד 9 ללא כפילויות. הפונקציה מחשבת את מספר הערכים הייחודיים בכל תת-רשת ומוסיפה אותם לציון הפיטנס.
3. בונוסים ועונשים על סמך התאמה למקור:
   * התחשבות בתאים הריקים במקור: הפונקציה מעניקה בונוסים אם התא הריק במקור מכיל מספר שלא מופיע פעמיים בשורה, בעמודה או בתת-רשת. אם המספר מופיע פעמיים, ניתנים עונשים.
   * התאמה למספרים המקוריים: אם התא במקור לא היה ריק והמספר הנוכחי שונה מהמספר המקורי, ניתנים עונשים (בכך הפונקציה שומרת על מספרים שהוזנו במקור).
4. איזון בין בונוסים לעונשים:
   * הפונקציה מאזנת בין מתן בונוסים על ייחודיות המספרים ועונשים על כפילויות וחוסר התאמה ללוח המקורי. כך היא מעודדת פתרונות תקינים ומענישה פתרונות שמפרים את חוקי הסודוקו או משנים את הלוח המקורי שלא לצורך.
5. **המנוע פתר את כל החידות באתר הנ"ל, דוגמאות:**

סודוקו 1:



סודוקו 2:



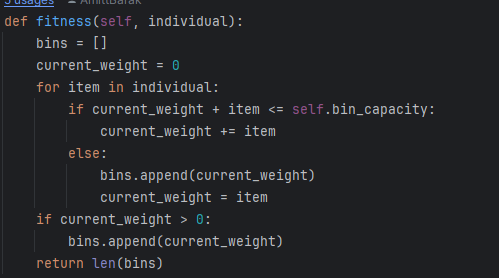
**שאלה 4:**

הוסיפו תמיכה למופע חדש של בעיה בעיית ה PACKING BIN-יש לארוז עצמים בנפחים שונים במספר מכלים בנפח V תוך שימוש במינימום מכלים https://en.wikipedia.org/wiki/Bin\_packing\_problem

1. גם לבעיה זו מצאו ייצוג, פונקציית פיטנס ופונקציית פיטנס אדפטיבית יעילה – נמקו בחירתכם
2. הריצו את האלגוריתם שלכם על חמשת הבעיות הראשונות בקובץ txt1.binpack – הסבר על פורמט באתר http://people.brunel.ac.uk/~mastjjb/jeb/orlib/binpackinfo.html והשוו את ביצועי האלגוריתם שלכם מבחינת מהירות ההתכנסות, איכות הפתרון וזמני ריצה לזה של אלגוריתם FIT FIRST החמדני.

ואת איכות הפתרון שהאלגוריתם מצאו והשוו מול המצוין בקובץ כאופטימלי.

**פתרון:**

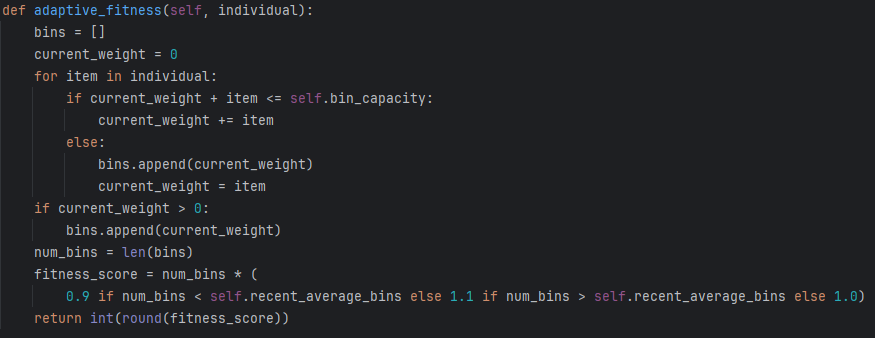
****

1. מבחינת פונקציית הפיטנס בחרנו ללכת על Fixed Fitness function, והסיבה שבחרנו ללכת עליה היא פונקציית הפיטנס נועדה להעריך באופן עקבי את "כושרו" של אינדיבידואל (נגיד גן) בגלל שהחישוב הוא עקבי ולא משתנה הוא מחשב את מספר ה-bins שצריך עבור הפריטים והעקביות הזאת חשובה לנו מכמה סיבות:
2. השוואתיות: מבטיחה שכל הפריטים מוערכים על פי אותם קריטריונים, ולכן מתבצעת השוואה ביניהם האופן הוגן ולא משוחד.
3. חיזוי: יכולת החיזוי של הפונקציה מסייעת בהבנה ובפירוש של התהליך האבולוציוני של היוריסטיקה והתוצאות שמופקות.
4. אמינות: קריטריון הערכה עקביים חיוניים ליציבות התהליך האבולוציוני, הימנעות מהתנהגויות בלתי צפויות.

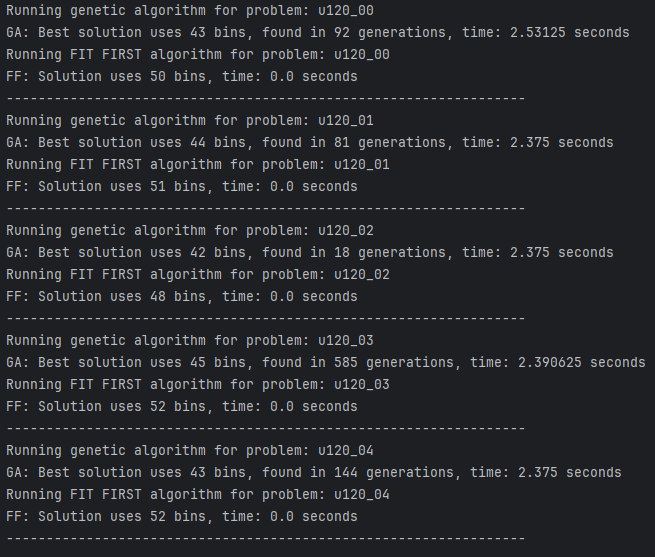
בגלל אופייה של היוריסטיקה הזאת שהיא לא אדפטיבית של פונקציית הפיטנס מביא לכך שהיא איננה משנה את שיטת ההערכה שלה בהתבסס על הביצועים במהלך התהליך האבולוציוני.

יש לכך מספר יתרונות:

1. פשטות: הפונקציה בגלל היותה לא אדפטיבית היא פשוטה יותר וליישום והבנה ובכך היא מורידה את מורכבותה של היוריסטיקה.
2. מיקוד על איכות הפתרון: התמקדות בעיקר במספר התאים ללא הכנסה של משתנים נוספים ובכך יוצר פתרון איכותי.



מבחינת הפונקציה הדינמית בחרנו ללכת על Self adaptive Fitness, פונקציית הפיטנס האדפטיבי מתאימה באופן דינמי על סמך הביצועים הממוצעים האחרונים של האוכלוסייה. להתאמה דינמית ישנה מספר יתרונות מרכזיים:

1. שיפור מתמשך: על ידי התאמת ציוני הפיטנס על סמך המספר הממוצע האחרון של תאים בשימוש, הפונקציה מעודדת שיפור מתמיד. פתרונות המשתמשים בפחות תאים מהממוצע האחרון "מתוגמלים", אבל אלו שמשתמשים ביותר תאים "נענשים".
2. תגובתיות: הפונקציה מגיבה למצב הנוכחי של האוכלוסייה, ומבטיחה שקריטריוני ההערכה מתפתחים עם ביצועי פיטנס באוכלוסייה. זה עוזר לשמור על הרלוונטיות של ציוני הפיטנס לאורך כל התהליך האבולוציוני.
3. משוב: יש בפונקציה מנגנון של משוב עם הענישה/תגמול, ובכך בעצם מעניק סוג של הדרכה לפתרון איכותי יותר.
4. ישנו איזון בין Exploration and Exploitation: על ידי הענשה הפונקציה למעשה מעודדת את חקר כלומר את Exploration ובכך לנסות למצוא פתרונות טובים יותר, על ידי תגמול אנחנו למעשה מבטיחים פתרון טוב יותר ומעודדים Exploitation אנחנו למעשה נלך על הידוע והטוב.
5. לאחר הרצה של האלגוריתם שלנו מול First fit קיבלנו את התוצאות הבאות:  
   

מבחינת איכות הפתרון: האלגוריתם הגנטי תמיד מוצא עם פחות תאים מאשר האלגוריתם של First fit בממוצע ההפרש עומד על 7 תאים פחות מאשר First fit, בנוסף יש לציין שאנו נמוכים גם מהפתרונות האופטימלי בקובץ שעומד על 48-50 תאים.

מבחינת זמן: האלגוריתם הגנטי רץ בין 2 ל-2.5 שניות בזמן שהתוצאה של First fit לפי הנראה מיידית, מה שמוכיח שהוא מאד מהיר מצד אחד ומצד שני הוא פחות אפקטיבי במציאת הפתרון האופטימלי.

מבחינת דורות: First fit לא בודק דורות כלל, אבל האלגוריתם הגנטי משתנה בין 18 ועד מעל 500 דורות על מנת למצוא את הפתרון האופטימלי.

יש לציין שאם נריץ עוד כמה פעמים התוצאות ישתנו אך הממוצע יישאר זהה.

מסקנה: בהשוואה בין השניים, אלגוריתם גנטי טוב יותר באופן משמעותי מבחינת איכות הפתרון (פחות שימוש בתאים) אבל לוקח יותר זמן לפעול. אלגוריתם FF הוא הרבה יותר מהיר אך פחות יעיל מבחינת השימוש בתאים.

יעילות: אם המטרה העיקרית היא למזער את מספר התאים והזמן אינו בעיה, האלגוריתם הגנטי הוא הבחירה הטובה יותר. אם הזמן הוא גורם קריטי ופתרון מעט גרוע יותר אלגוריתם FF עדיף.

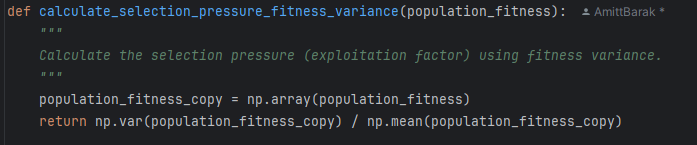
**שאלה 5:**

הוסיפו שיטות למדידת לחץ הבחירה Pressure Selection Exploitation Factor

1. Fitness Variance
2. Top-Average Selection Probability Ratio

דווחו מדדים אלה בכל דור של אבולוציה

**פתרון:**



A computer screen shot of a program code

Description automatically generated

**שאלה 6:**

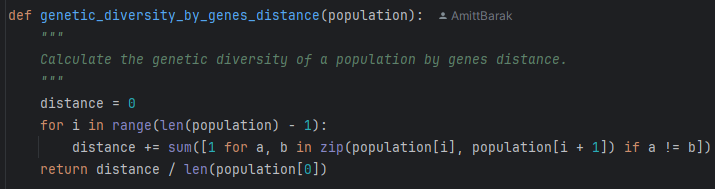
הוסיפו שיטות למדידת הגיוון הגנטי (genetic diversity) [חלק זה מותאם בעיה] – Exploration Factor

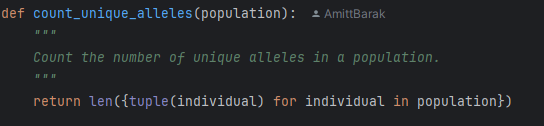
1. המרחק בין גנים.
2. מספר האללים השונים באוכלוסייה.

דווחו מדדים אלה בכל דור של האבולוציה.

**פתרון:**

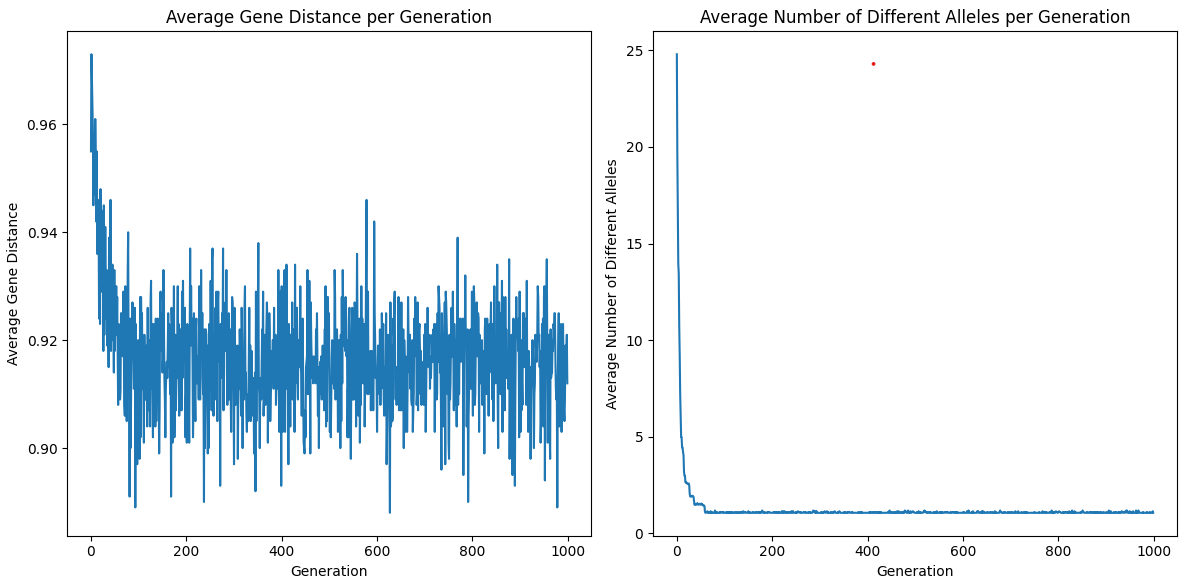
עבור סודוקו, מימשנו את הפונקציות הבאות:



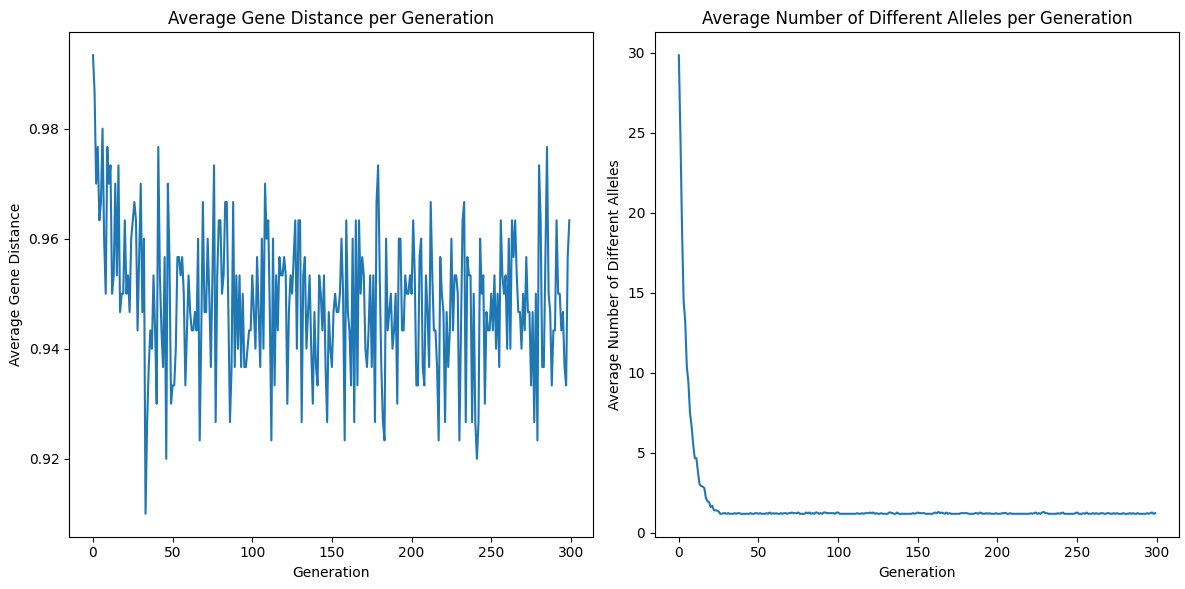


עבור בעיית הbins לא היה צורך מימוש נוסף:

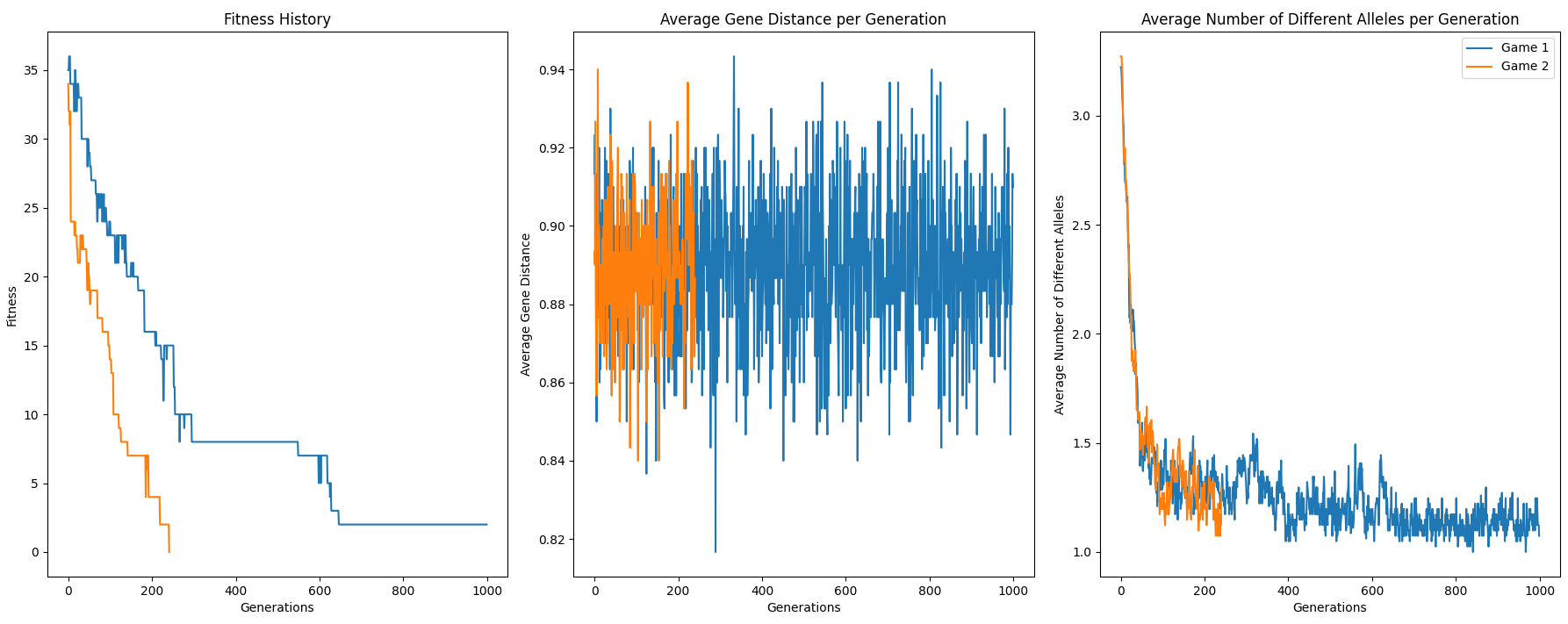
מבחינת פונקציית המרחק בין הגנים היא דינמית ואינה דורשת שינוי וכך גם הפונקציה של מספר האללים השונים באוכלוסייה.

דיווחנו על מדדים אלו בכל דור ולקחנו למעשה 1000 דורות כדגימה וקיבלנו את הגרף הבא:  


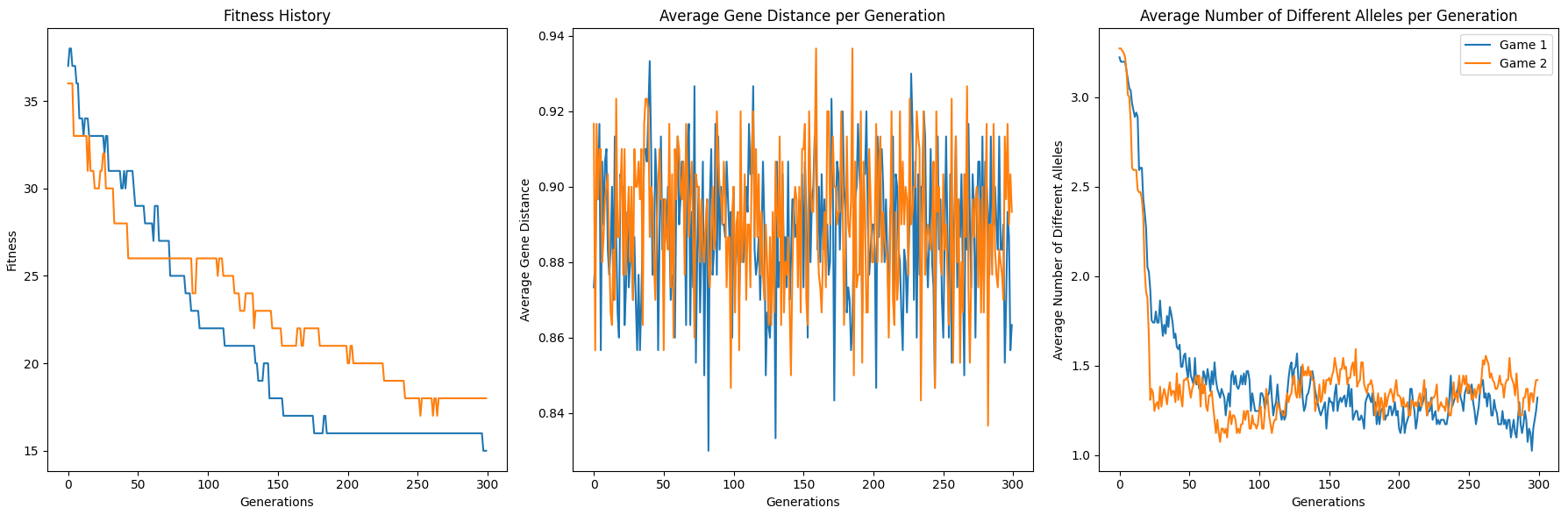
כאשר בדקנו 300 דורות קיבלנו מרחקים דומים ומספר אללים דומה מבחינת הפיזור כמובן:



כאשר בדקנו עבור הסודוקו (2 משחקים ראשונים לא חשבנו שיש צורך לאחרים) קיבלנו ב1000 דורות:



כאשר בדקנו ל300 דורות:



מסקנה שלנו מגרפי בעיית האריזה היא שבין אם 1000 דורות או 300 אינו משנה מרחק בין הגנים נמצא בטווח ברובו של 0.92 ל-0.98 אך מספרר האוללים השונים מתחיל מגבוה מאד (שונים גבוה) ומתאזן.

מבחינת הגרפים של הסודוקו ניתן לראות שמספר האוללים ב1000 דורות במשחק 2 נעצר ואינו מתקדם ולכן ניתן להסיק כי אין יותר שוני לעומת 300 דורות ששם השונים מתמשך וגדל וקטן בהתאם, וניתן להסיק כי הפיטנס יורד גם הוא ב1000 דורות מה שגורם לכך שאין אוללים חדשים לכאורה, ושוב ב300 דורות הפיטנס יורד אך ועולה אך ממשיך.

המרחק בין הגנים ב1000 דורות נעצר בערך בדור 300 אז ניתן שוב לצאת מנקודת הנחה שזה גם אחד הגורמים לכך שאין אוללים שונים או חדשים אך ב300 דורות הפונקציה מודדת ויחסית יציבה.

**שאלה 7:**

בדקו באמצעות סימולציות את רגישות פתרון שתי הבעיות (סודוקו וbin packing)

לפי הקריטריונים של מהירות ההתכנסות , איכות הפתרון וזמני ריצה עפ"י הפרמטרים הבאים:

1. לגודל האוכלוסייה
2. להסתברות למוטציות
3. לאסטרטגיית הבחירה
4. לאסטרטגיית השרידות (aging, elitism)
5. לאסטרטגיית שיחלוף והמוטציה

**פתרון:**

**עבור סודוקו:**

**a**. גודל האוכלוסייה

השפעה על מהירות ההתכנסות:

* גודל אוכלוסייה קטן: כאשר גודל האוכלוסייה קטן, האלגוריתם יתקדם מהר יותר בהתכנסות מכיוון שיש פחות אינדיבידואלים לעבד בכל דור. עם זאת, הסיכון לכניסה לאופטימום מקומי גדל, מה שיכול להאט את ההתכנסות לפתרון נכון.
* גודל אוכלוסייה גדול: גודל אוכלוסייה גדול יכול לסייע במניעת אופטימום מקומי על ידי שמירה על מגוון גנטי רחב יותר. זה עשוי להאט את הזמן שנדרש לכל דור, אך בסופו של דבר להוביל להתכנסות לפתרון נכון יותר.

השפעה על איכות הפתרון:

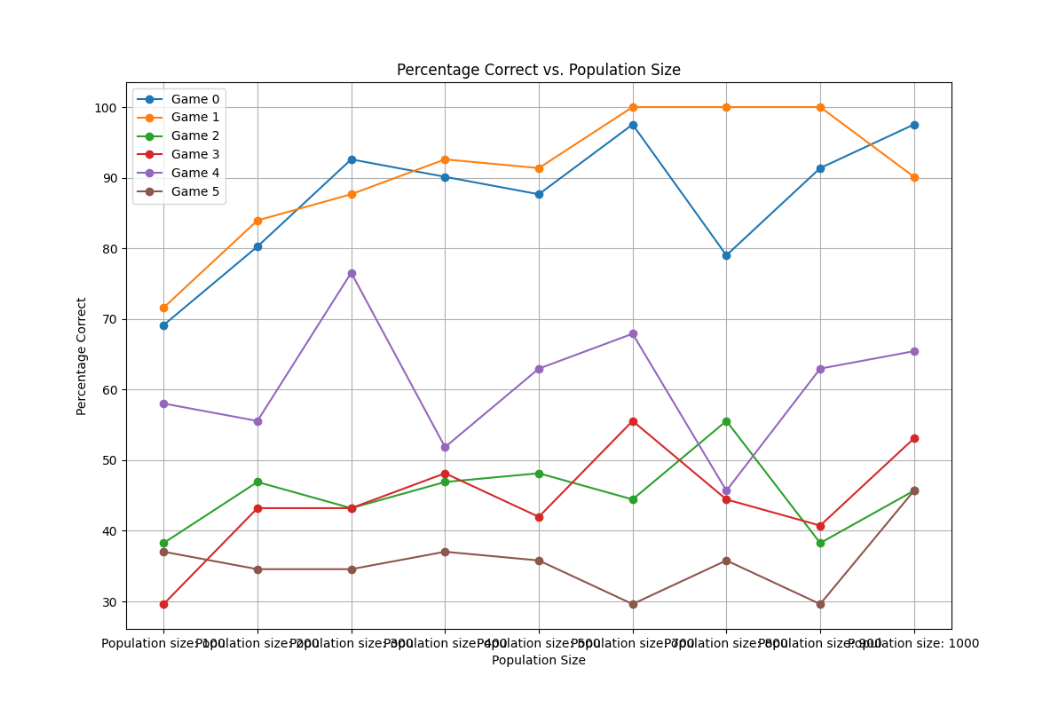
* גודל אוכלוסייה קטן: איכות הפתרון עשויה להיות נמוכה יותר עקב מחסור בגיוון גנטי.
* גודל אוכלוסייה גדול: גודל אוכלוסייה גדול עשוי לשפר את איכות הפתרון מכיוון שיש יותר גנים להתפזר ולבדוק במהלך האבולוציה.

השפעה על זמני ריצה:

* גודל אוכלוסייה קטן: זמני ריצה קצרים יותר לדור בודד.
* גודל אוכלוסייה גדול: זמני ריצה ארוכים יותר לדור בודד אך עשויים להתקזז בגלל התכנסות מהירה יותר לפתרון נכון.

גרף להמחשת גודל האוכלוסייה על נכונות הפתרון:

(עבור ערכי אוכלוסייה [100, 200, 300, 400, 500, 700, 800, 900, 1000])



**b**. הסתברות למוטציות

השפעה על מהירות ההתכנסות:

* הסתברות למוטציה נמוכה: עלולה להוביל להתכנסות מהירה לאופטימום מקומי מבלי למצוא את הפתרון הטוב ביותר.
* הסתברות למוטציה גבוהה: עשויה להאט את ההתכנסות בגלל שינויים תכופים, אך עוזרת למנוע אופטימום מקומי על ידי הוספת מגוון גנטי.

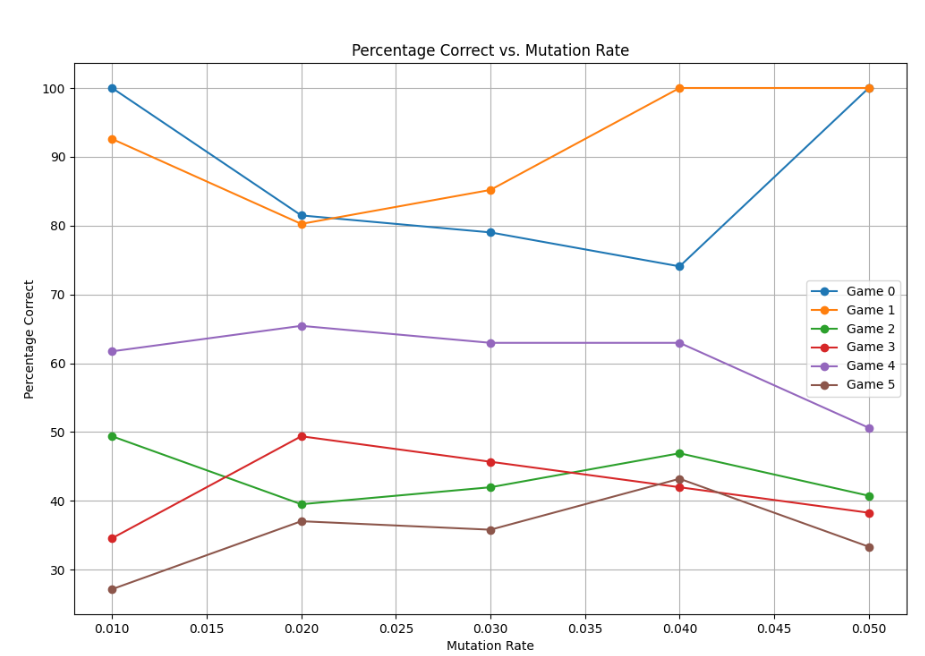
השפעה על איכות הפתרון:

* הסתברות למוטציה נמוכה: עשויה להוביל לפתרונות פחות טובים בשל מחסור במגוון גנטי.
* הסתברות למוטציה גבוהה: עשויה לשפר את איכות הפתרון על ידי הוספת גנים חדשים ושונים.

השפעה על זמני ריצה:

* הסתברות למוטציה נמוכה: זמני ריצה קצרים יותר.
* הסתברות למוטציה גבוהה: זמני ריצה ארוכים יותר בשל שינויים תכופים.

גרף להמחשת השפעת ההסתברות למוטציות על נכונות הפתרון:



**c**. אסטרטגיית הבחירה

השפעה על מהירות ההתכנסות:

* SUS ורולטה: עשויות להוביל להתכנסות איטית יותר מכיוון שהן יכולות לבחור גם אינדיבידואלים פחות טובים.
* טורניר ודירוג: עשויות להוביל להתכנסות מהירה יותר על ידי בחירה באינדיבידואלים הטובים ביותר.

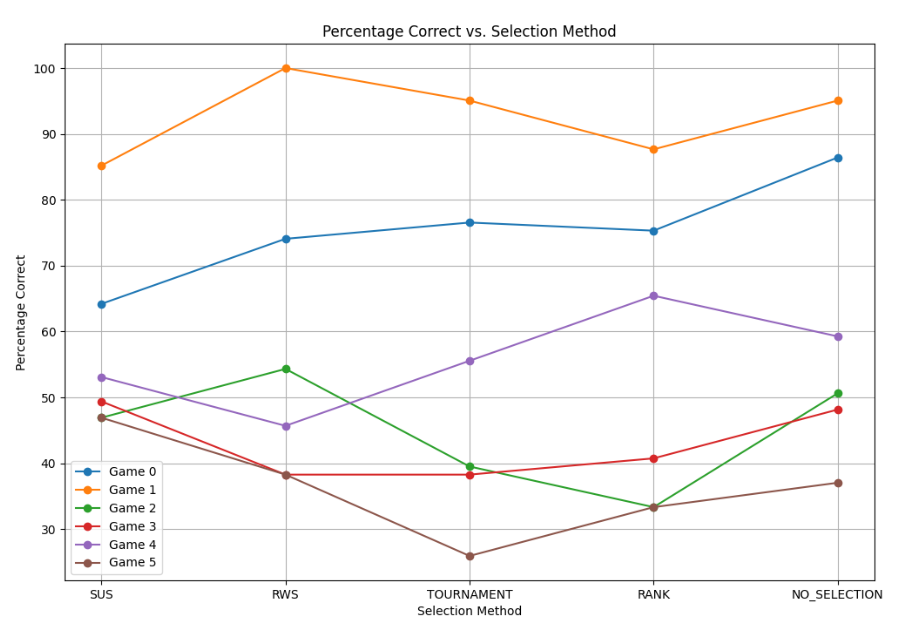
השפעה על איכות הפתרון:

* SUS ורולטה: עשויות לשמור על מגוון גנטי רחב יותר אך עלולות לסבול מאופטימום מקומי.
* טורניר ודירוג: עשויות לשפר את איכות הפתרון אך עלולות להוביל לאופטימום מקומי.

השפעה על זמני ריצה:

* SUS ורולטה: זמני ריצה קצרים יותר לבחירה.
* טורניר ודירוג: זמני ריצה ארוכים יותר לבחירה.

גרף להמחשת השפעת אסטרטגיית הבחירה על נכונות הפתרון:



**d**. אסטרטגיית השרידות (aging, elitism)

השפעה על מהירות ההתכנסות:

* Aging: עשויה להאט את ההתכנסות בשל שמירה על גיוון גנטי.
* Elitism: עשויה להאיץ את ההתכנסות על ידי שמירה על האינדיבידואלים הטובים ביותר.

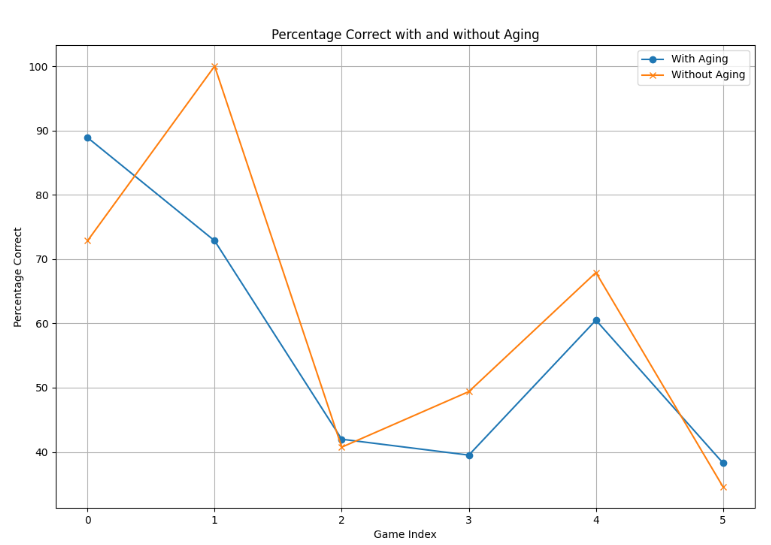
השפעה על איכות הפתרון:

* Aging: עשויה לשפר את איכות הפתרון על ידי מניעת אופטימום מקומי.
* Elitism: עשויה לשפר את איכות הפתרון אך להוביל לאופטימום מקומי.

השפעה על זמני ריצה:

* Aging: זמני ריצה ארוכים יותר בשל חישובים נוספים.
* Elitism: זמני ריצה קצרים יותר.

גרף להמחשת אסטרטגיית השרידות על נכונות הפתרון:



**e**. אסטרטגיית שיחלוף והמוטציה

* אסטרטגיית השחלוף:

השפעה על מהירות ההתכנסות:

* :Cycle Crossoverעשוי להוביל להתכנסות מהירה יותר על ידי שמירה על סדרים קיימים.
* שיחלוף PMX עשוי להוביל להתכנסות איטית יותר בשל חילופי מיקומים רבים יותר.

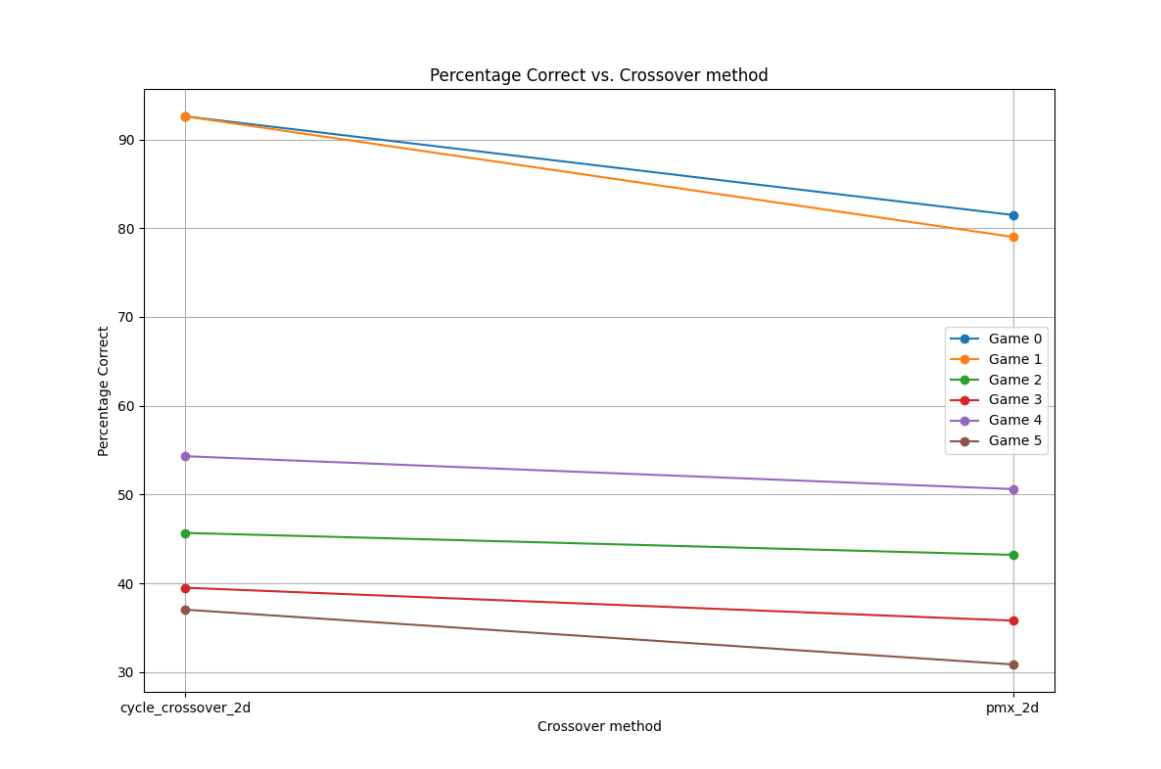
השפעה על איכות הפתרון:

* Cycle Crossover: עשוי לשמור על פתרונות טובים ולשפר את איכות הפתרון.
* שיחלוף :PMX עשוי לשפר את איכות הפתרון על ידי יצירת גנים חדשים ומגוונים.

השפעה על זמני ריצה:

* :Cycle Crossover זמני ריצה קצרים יותר.
* :PMX זמני ריצה ארוכים יותר בשל חילופים רבים.

גרף להמחשת אסטרטגיית שיחלוף על נכונות הפתרון:



* סוג המוטציה:

השפעה על מהירות ההתכנסות:

* היפוך: המוטציה עשויה להאט את מהירות ההתכנסות כי היא משנה את הסדר של חלקים בכרומוזום.
* ערבול: יכולה להאיץ את מהירות ההתכנסות על ידי הוספת גיוון משמעותי יותר מאשר היפוך.

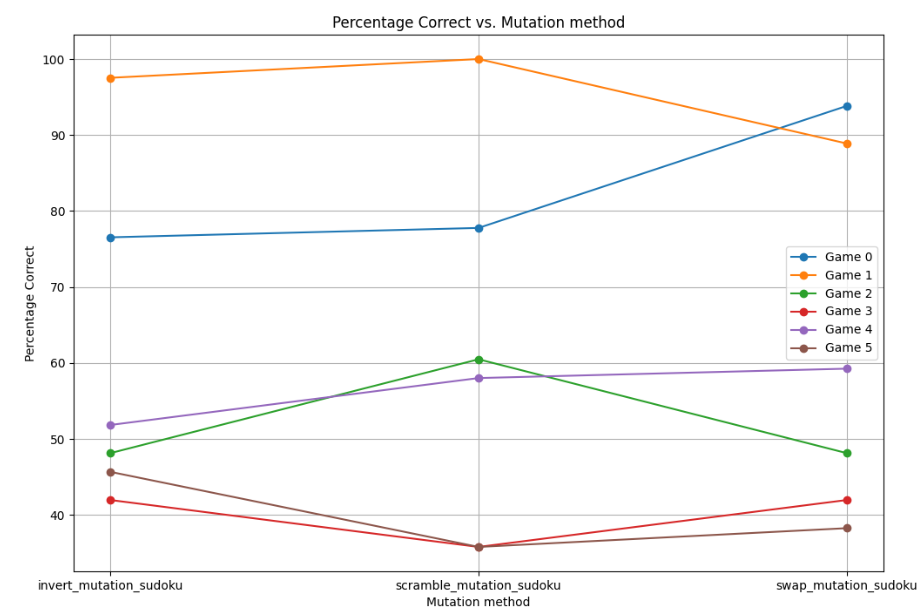
השפעה על איכות הפתרון:

* היפוך: יכולה לשפר את איכות הפתרון על ידי הוספת גיוון לתהליך האופטימיזציה.
* איכות הפתרון: עשויה לשפר את איכות הפתרון, אך אם הערבול יותר מדי רנדומלי, הוא עלול לפגוע באיכות.

השפעה על זמני הריצה:

* היפוך: תהליך ההיפוך הוא בדרך כלל מהיר, כך שזמני הריצה לא יושפעו משמעותית.
* ערבול: זמני הריצה עלולים להיות מעט ארוכים יותר בהשוואה להיפוך, אך לא בצורה משמעותית.

גרף להמחשת סוג המוטציה על נכונות הפתרון:

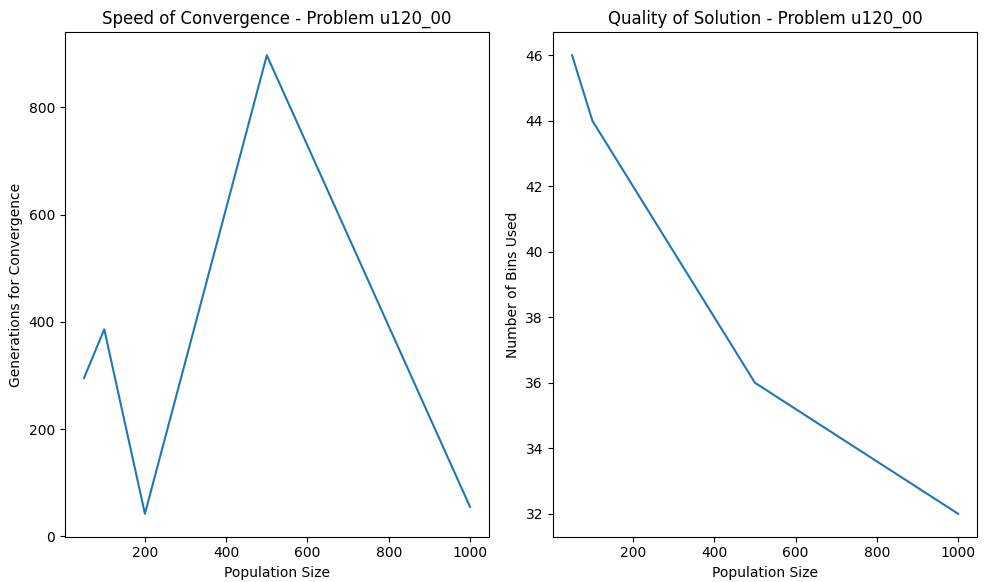


* Swap זוהי המוטציה שמימשנו כבר במעבדה 1 חלק א עבור ה-hello world. התאמנו אותה לסודוקו ע"מ להשוואת אותה ליתר המוטציות גם כן.

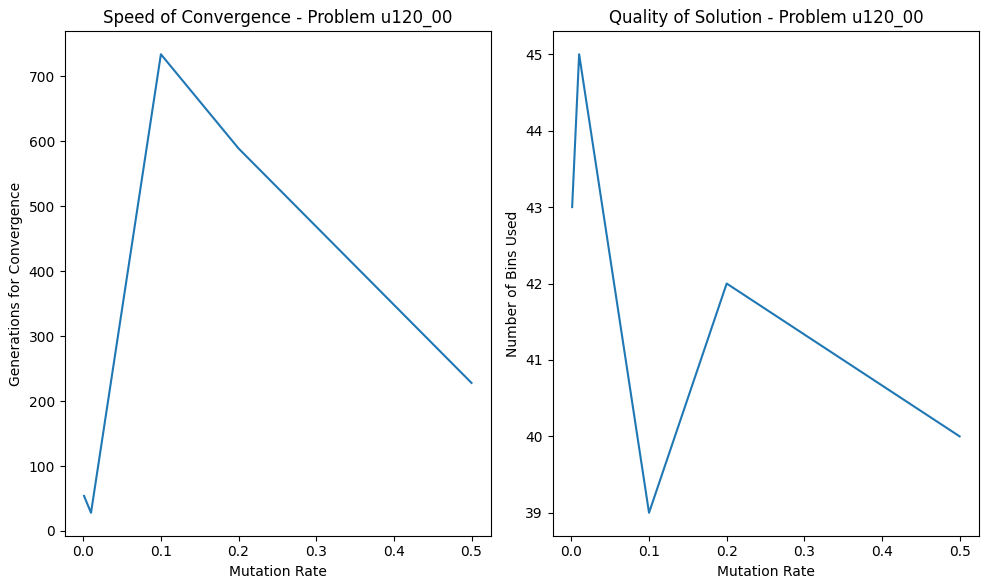
**עבור bin packing:**

**Bin Packing**

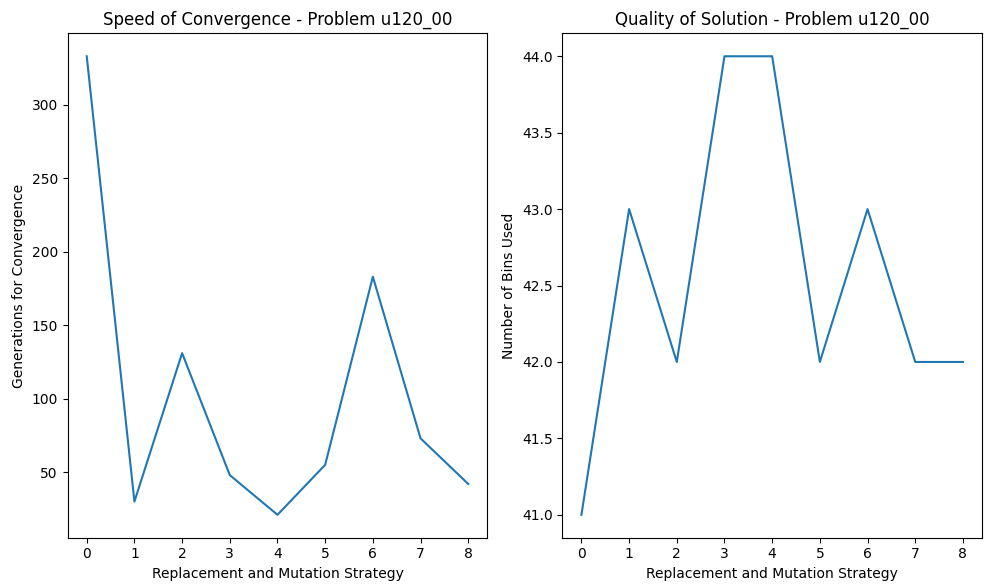
**לפי גודל האוכלוסייה:**

****

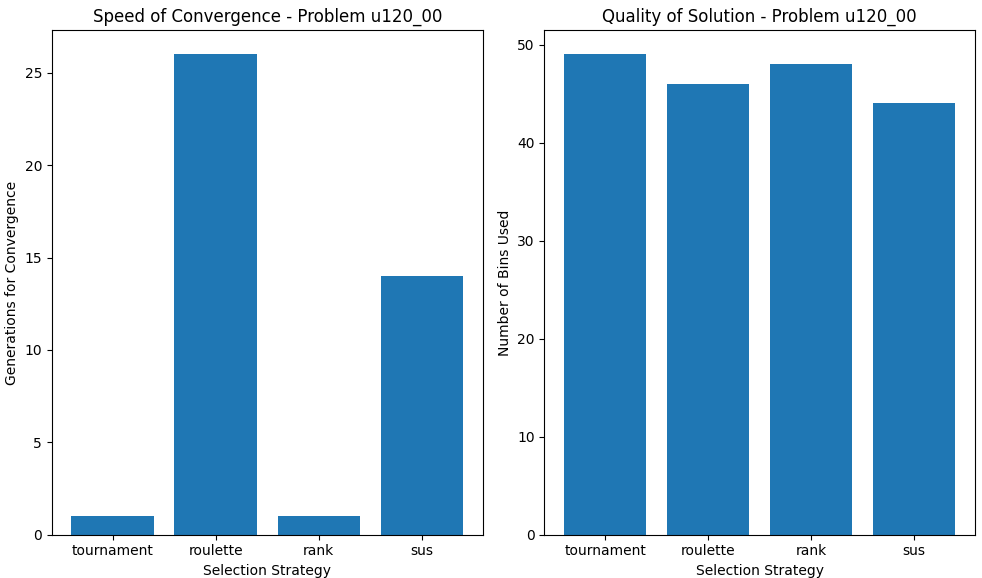
**לפי הסתברות למוטציות:**

****

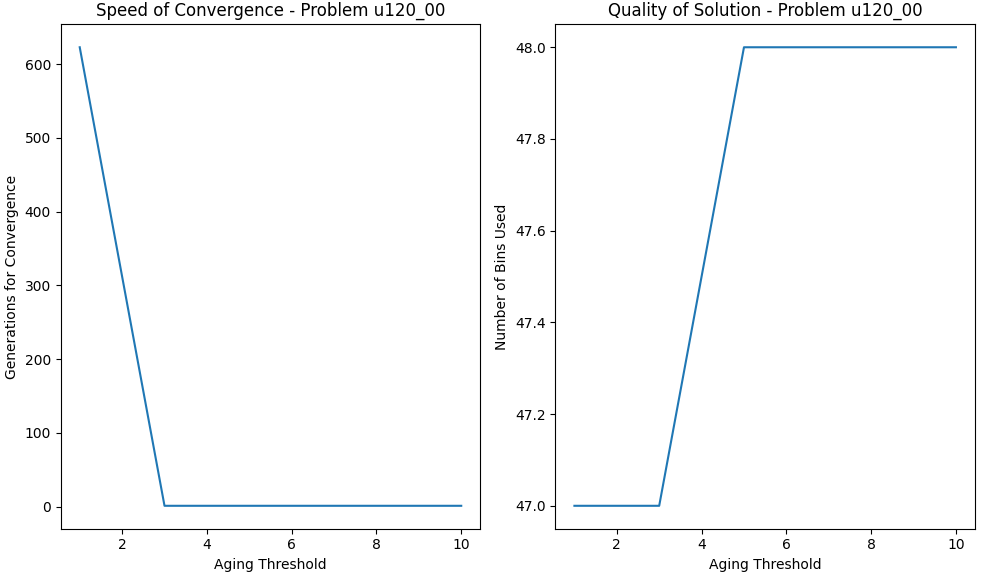
**לפי חלופיות ומוטציות:**

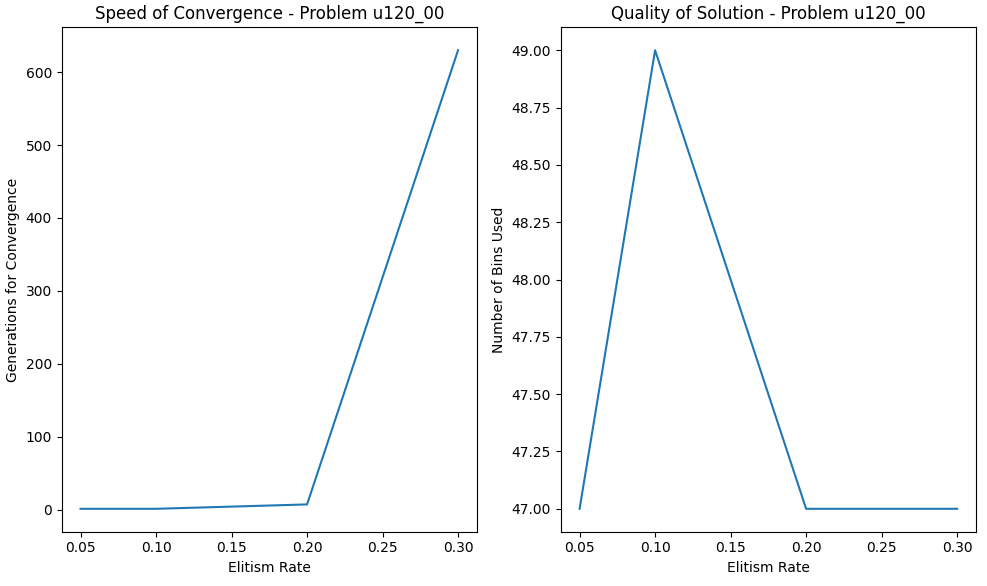
****

**לפי בחירה:**

****

**לפי aging/elitism:**

****

****

**שאלה 8:**

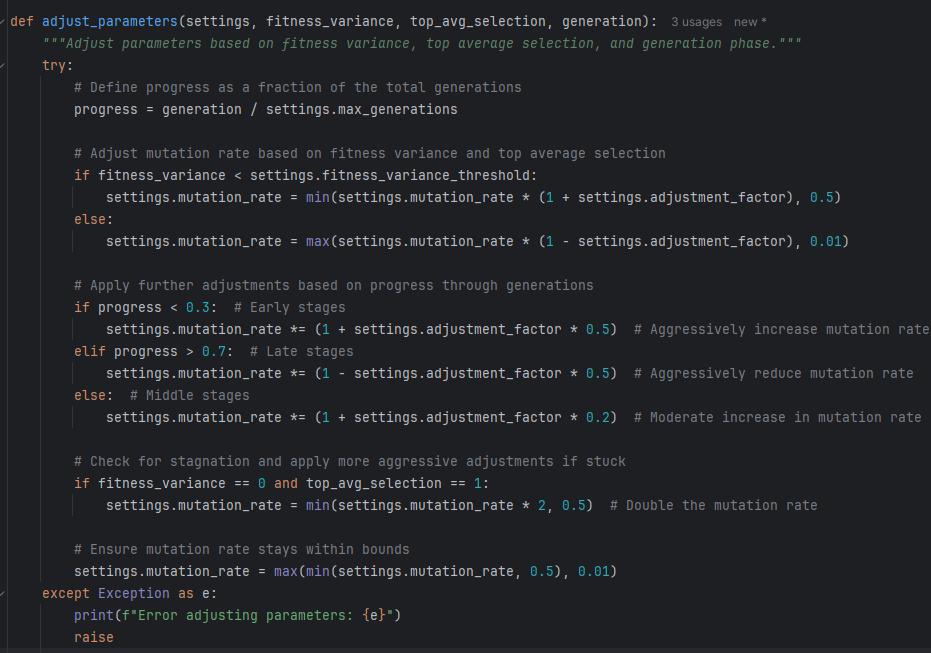
חלצו והציגו את הפרמטריזציה המיטבית עבור בעיות אלו

1. הסבירו כיצד נעזרתם במדדי הלחץ והגוון למצוא פרמטריזציה זו.
2. מה המסקנות שעולות מממצאים אלה.

**פתרון:**

**עבור סודוקו:**

1. **נעזרנו במדדי הלחץ והגוון למצוא את הפרזנטציה באופן הבא:**
   * מדד השונות בכשירות: שונות גבוהה מצביעה על גיוון רב באוכלוסייה, דבר שחשוב בשלב ההתחלתי. על כן, העלינו את שיעור המוטציה כדי לשמר גיוון. שונות נמוכה מצביעה על קונברגנציה, שעשויה להצביע על קיפאון או התקרבות לפתרון. הפחתנו את שיעור המוטציה כדי לכוונן את הפתרונות.
   * **יחס ההסתברות לבחירת האינדיבידואלים הטובים ביותר:** יחס גבוה מצביע על ניצול יתר של הפתרונות הטובים ביותר. העלינו את שיעור המוטציה כדי להוסיף גיוון. יחס נמוך מצביע על איזון בין חיפוש לניצול. הפחתנו את שיעור המוטציה כדי להתמקד יותר בניצול הפתרונות הטובים.



1. **המסקנות שעולות ממצאים אלו הינם:** 
   * שיעור המוטציה: צריך להיות מותאם דינמית לפי השונות בכשירות והתקדמות הדורות. בשלב המוקדם נדרשת עלייה חדה במוטציה לשימור גיוון, ובשלב המאוחר נדרשת הפחתה לשיפור הפתרונות.
   * גודל האוכלוסייה: גודל גדול שומר על גיוון גנטי ומונע קונברגנציה מקומית, אך מגדיל את זמן הריצה.
   * שיטת הבחירה: שיטות כמו SUS ו-RWS מספקות איזון טוב בין חיפוש לניצול, בעוד שיטות כמו טורניר ודירוג מתמקדות יותר בניצול הפתרונות הטובים.
   * אסטרטגיית שימור: שימוש ב-Aging ו-Elitism משפיע על האיזון בין גיוון לניצול, כאשר Aging שומר על גיוון ו-Elitism משמר את הטובים ביותר.

הערות:

* + 1. גרפים המצורפים (בשאלה 7 לעיל) ממחישים את ההשפעה של פרמטרים שונים על איכות הפתרון וזמן ההתכנסות.
    2. הפרזנטציה המיטבית לכל בעיית סודוקו מוזנת בקוד המצורף.

**עבור בעיית Bin packing:**

בדקנו עם בחירה/aging/elitism וגודל אוכלוסייה שנוכל להשוות למה שמצאנו קודם לכן.

A group of blue lines

Description automatically generated

A graph of a graph

Description automatically generated with medium confidence

A group of graphs showing different types of data

Description automatically generated with medium confidence

A group of blue lines

Description automatically generated

A group of blue lines

Description automatically generatedA graph of a graph

Description automatically generated with medium confidence

A graph of a graph

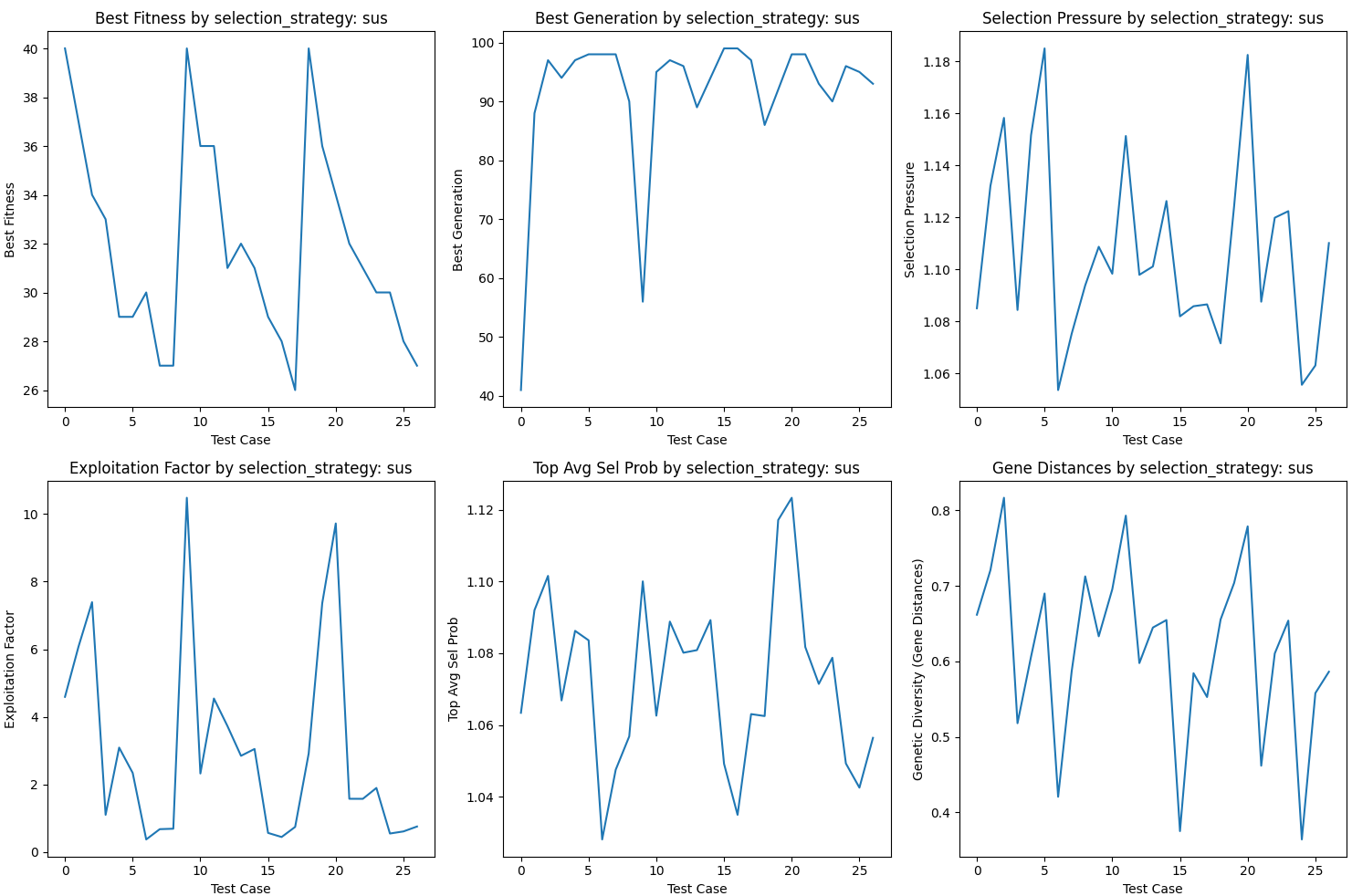
Description automatically generated with medium confidence

A graph of a graph

Description automatically generated with medium confidence

A group of blue lines

Description automatically generated

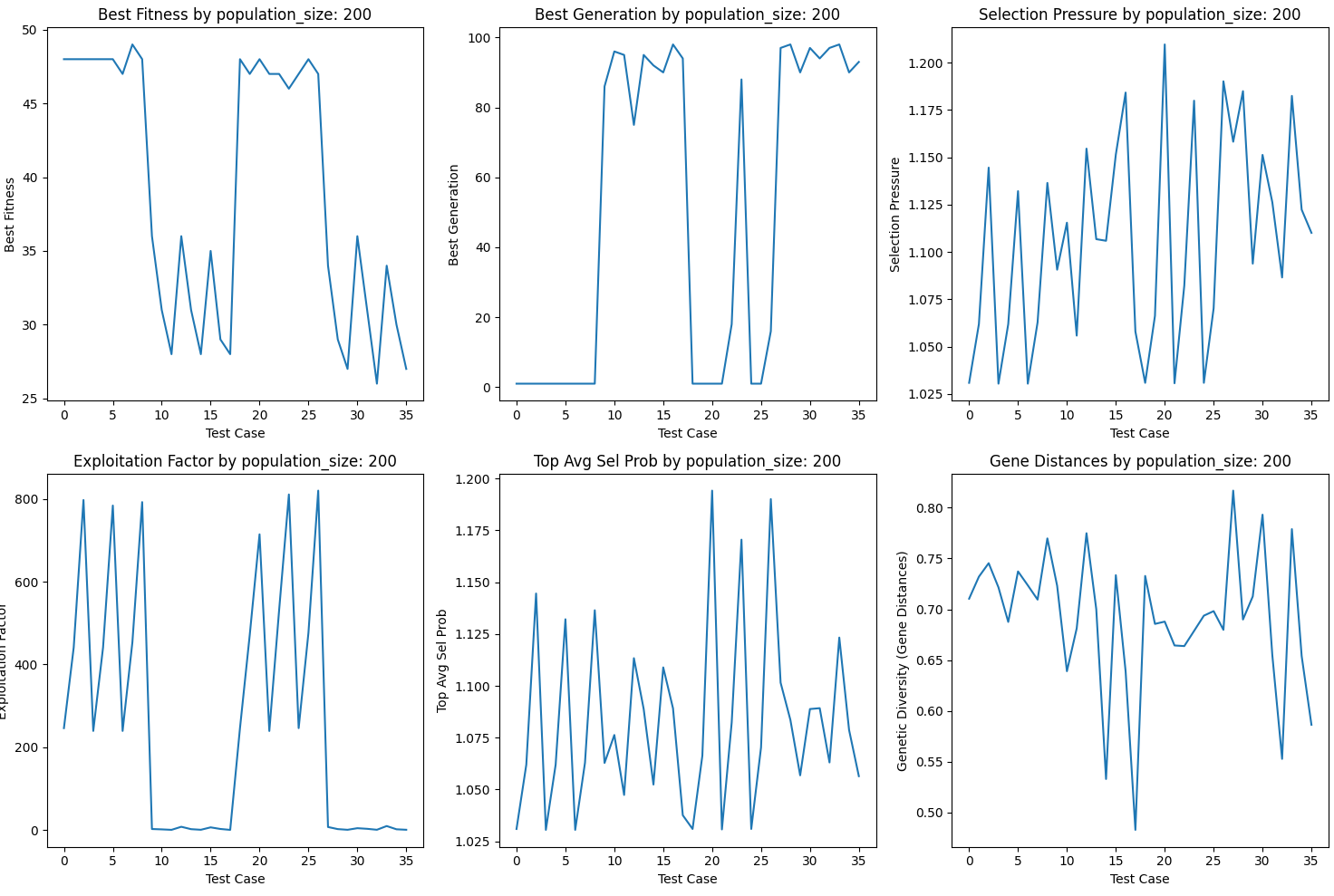


A graph of blue lines

Description automatically generated with medium confidence

A graph of a graph

Description automatically generated with medium confidence



מספר מדדים עוזרים לנו להעריך את הביצועים והמאפיינים והגדרות פרמטרים שונים. מדדים אלו מספקים תובנות לגבי התנהגות האלגוריתם ויעילותו, לדוגמה המטריצות והגדרתן:

**Selection pressure**: המטריצה הזאת מודדת יחסים של הפיטנס הגבוה ביותר עם הממוצע באוכלוסייה, לחץ בחירה גבוה מדי היה יכול לגרום לזמן התכנסות מהיר יותר אבל גם התכנסות מוקדמת ובכך לטעות.

**Exploitation factor:** המטריצה הזאת מודדת את ההתפלגות של הפיטנס באוכלוסייה, התפלגות גבוהה יותר מראה על יותר Exploitation ובכך האלגוריתם למעשה משפר תוצאות או מחדד אותם אם תרצה.

**Top average selection probability:** המטריצה בודקת את היחס בין הפיטנס הממוצע של אינדיבידואל לפיטנס הממוצע באוכלוסייה, זה נותן את כמה יותר טוב האינדיבידואל לעומת הממוצע באוכלוסייה.

**:Genetic diversity** המטריצה הזאת מעריכה את המגוון באוכלוסייה על בסיס של מרחק גנים, המגוון באוכלוסייה עוזר למנוע בעצם זמן התכנסות מהיר על ידי תחזוקת המגוון של הפתרונות.

ניתוח התוצאות:

**Selection pressure**: בניתוח זה למעשה הבטחנו שהלחץ לא גבוה מדי (מוביל להתכנסות מוקדמת) וגם לא נמוך מדי (מוביל להתכנסות איטית). מצאנו טווח אופטימלי כדי שנוכל לאזן בין exploration ו- exploitation.

**Exploitation factor:** השונות בערכי הפיטנס עזרו לנו להבין את מידת ה- exploitation. רצינו יותר exploitation מתון כדי שנוכל לחדד/לשפר פתרונות טובים מבלי לאבד את הגיוון.

**Top average selection probability:** היחס עזר לנו לזהות אם אינדיבידואלים היו טובים משמעותית מהממוצע, מה שימעיד על לחץ בחירה יעיל.

**:Genetic diversity** מגוון גנטי גבוה מועדף כדי למנוע התכנסות מוקדמת. חיפשנו הגדרות ופרמטרים שישמרו על מגוון גנטי לאורך הדורות.

בהתבסס על ניתוח זה, נבחרו הגדרות פרמטר אופטימליות. לדוגמה, העדיפו ספי הזדקנות ושיעורי אליטיזם ששמרו על מגוון גנטי ולחץ בחירה מתון.

נבחרו אסטרטגיות בחירה שיצרו exploitation and exploration מאוזנים.

על ידי ניתוח שיטתי, הצלחנו לזהות הגדרות פרמטרים השגנו איזון בין מהירות התכנסות לגיוון גנטי, מה שהוביל לאלגוריתם גנטי יעיל וחזק לבעיית bin packing, תוך כדי כך שנמנעו טעויות או מלכודות כמו התכנסות מוקדמת או חישוביות מוגזמת.