הטכניון - מכון טכנולוגי לישראל

הפקולטה להנדסת חשמל ע"ש אנדרו וארנה ויטרבי

המעבדה לבקרה, לרובוטיקה וללמידה חישובית

**ספר פרויקט**

**אלגוריתם זמן-אמת לניהול שדה תעופה**

מבצעים:

יואב כהן Yoav Cohen

הודיה כהן-אדיב Hodaya Cohen-Adiv

מנחה:

אייל טייטלר Eyal Taitler

סמסטר רישום: חורף תשפ"א

תאריך הגשה: אוקטובר, 2021

# תוכן עניינים

[רשימת איורים 3](#_Toc83229266)

[תקציר 3](#_Toc83229267)

[Abstract 3](#_Toc83229268)

[מבוא 3](#_Toc83229269)

[תיאור כללי 3](#_Toc83229270)

[תוצאות 3](#_Toc83229271)

[סיכום ומסקנות 3](#_Toc83229272)

[נספחים 3](#_Toc83229273)

[רשימת מקורות 4](#_Toc83229274)

# 

# רשימת איורים

# תקציר

בפרויקט זה נממש אלגוריתם online המסתמך על אלגוריתם offline קיים.

אלגוריתם ה-online מקבל תוכנית אופטימלית של המראות ונחיתות מטוסים ליום עבודה, ובמהלך היום יצטרך להתמודד עם תקלות ושינויים בזמן-אמת. זאת כאשר ההיבט המרכזי הוא התמודדות עם התקלות והשינויים בזמן-אמת תוך סטייה מינימלית מהתוכנית המקורית.

במקרה שהאלגוריתם לא ימצא פתרון בזמן-אמת, הוא ישלח את המצב הקיים עם האילוץ לאלגוריתם ה-offline שיבצע תכנון מחדש לתוכנית.

את האלגוריתם פיתחנו בסביבת python על סמך אלגוריתם ה-offline הקיים.  
בבסיסו אלגוריתם ה-online צריך לדעת לקבל את נתוני הפלט של אלגוריתם ה-offline אודות התוכנית האופטימלית של זמני ההמראה והנחיתה של המטוסים. כמו כן, עליו לתמוך בקבלת התקלות והשינויים בזמן-אמת. בנוסף, עליו לספק החלטה אודות המשך התוכנית בהתאם לתקלות ולשינויים.

# מבוא

* 1. **מוטיבציה**

# **כידוע, שדה תעופה מורכב מפעולות רבות במהלך היום – הוא מכיל מספר רב של מטוסים ומספר רב של מסלולים, ועליו לסנכרן בין כל הפעולות השונות ולפקח עליהן. שדה תעופה מכיל סוגים שונים של פעולות לדג': פעולות רבות המתרחשות בו-זמנית, פעולות אשר תלויות אחת בשנייה, פעולות אשר יוצרות הפרעה ועלולות לשבש את סדר היום שנקבע מראש.**

# **פיקוח ובקרה על שדה תעופה מתבצע בדרך כלל ע"י קביעת תוכנית לסדר-יום. תוכנית זו מכילה את לוח הזמנים עבור כל יום מראש. כאשר לוח הזמנים כולל את סדר ההמראות והנחיתות ואת זמנם, בנוסף הוא מציין את מספר המסלול עבור ההמראה והנחיתה של כל מטוס. כמו כן, הפיקוח והבקרה דורשים מעקב אחר התוכנית בזמן אמת, וטיפול בתקלות ובשינויים המתרחשים במהלך היום.**

* 1. **הגדרת הבעיה**

בניית תוכנית לסדר-יום עבור סדר ההמראות והנחיתות של המטוסים, פיקוח ובקרה אחריה בזמן-אמת וטיפול בבעיות המתרחשות במהלך היום, היא בעיה מורכבת.

ניתן לחלק את הבעיה לשתי תתי-בעיות:

1. בניית תוכנית יום – זה הוא אלגוריתם ה-offline[[1]](#footnote-1), אלגוריתם זה מתבסס על נתונים שמוזנים כקלט לבעיה.
2. פיקוח ובקרה אחר תוכנית היום בזמן אמת – זה הוא אלגוריתם ה-online שמימשנו במסגרת פרויקט זה. אלגוריתם זה מתבסס על תוכנית יום קיימת שמוזנת כקלט לבעיה.
   * 1. **אלגוריתם ה-Offline**

נרחיב מעט על אלגוריתם ה-offline, כיוון שאנו משתמשים ב-log המוצא שלו כקלט לאלגוריתם ה-online.

אלגוריתם ה-offline מבוסס על תורת הגרפים, על יוריסטיקות ומקבל boost מרשתות נוירונים.

האלגוריתם מקבל כקלט קובץ קונפיגורציה המכיל את מספר המטוסים ואת מספר המסלולים בבעיה. בנוסף, עבור כל מטוס מכיל וקטור של נתונים התחלתיים. (תמונה של קובץ קונפיג + ההסברים המלאים של בר ותום?)

קובץ קונפיגורציה חוקי מכיל את הבאים (לפי הסדר משמאל לימין(:

1. מזהה המטוס.
2. זמן המינימלי (יחסית לתחילת הבעיה) בו המטוס חייב להתחיל להתיישר על מסלול.
3. הזמן המקסימלי (יחסית לתחילת הבעיה) בו המטוס חייב להתחיל להתיישר על מסלול.
4. הזמן שלוקח למטוס לבצע את משימתו באוויר.
5. הזמן המקסימלי בו המטוס יכול להיות באוויר.
6. הזמן המקסימלי (יחסית לתחילת הבעיה) בו המטוס חייב לנחות ולפנות את מסלול הנחיתה.
7. סטטוס המציין האם בתחילת הבעיה המטוס נמצא על הקרקע (1) או באוויר (5).

נציין כי כל הזמנים בתת סעיפים 1-7 מוגדרים בדקות. בנוסף, קובץ הקונפיגורציה מכיל את מספר המטוסים, מספר המסלולים וחסם עליון לזמן מציאת פתרון.

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטידוגמא לקובץ קונפיגורציה חוקי:

איור 1 – קובץ קונפיגורציה חוקי

אלגוריתם ה-offline מחשב את סדר היום האופטימלי בהינתן הנתונים שהוגדרו לו. האלגוריתם מוציא כפלט את תכנית סדר-היום אשר חישב.

הסבר על STN עכשיו? +מקרה מורחב

* **תכנון עם אילוצי זמנים**

בשונה מתכנון קלאסי, תכנון עם זמנים ואילוצים מתמקד בבעיה בה לכל פעולה יש התחלה וסוף, כלומר לכל פעולה קיים ממד נוסף והוא זמן הביצוע של הפעולה. בתכנון עם זמנים, יש צורך לקבוע את זמני ההתחלה והסיום של הפעולות וכן את האילוצים ואת התלויות בין הפעולות השונות. לצורף פשטות, נהוג לקבוע את כלל הזמנים בבעיה ביחס לזמן תחילת הבעיה אשר נקבע להיות 0.

דוגמא לאילוץ אפשרי: בבעיה בה כלל הזמנים מוגדרים ברזולוציה של דקות. כאשר מטוס ממריא במשך 20 דקות, בין זמן 0 לזמן 20 לא אפשרי שמטוס אחר ינחת באותו המסלול ובאותו חלון זמנים. ולכן, צריך להיות סדר בין הפעולות והאילוצים בין המטוסים.

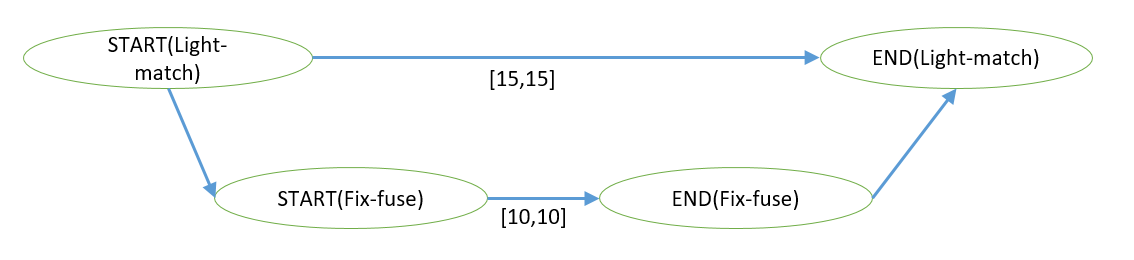
נציין כי בעיית הזמנים המאולצת היא בעיית אופטימיזציה עם אילוצים לכל דבר ועניין. לכן להגדרתה ולמציאת פתרון חוקי כנדרש נעשה שימוש בכלי אופטימיזציה מתאים.   
בנוסף נציין, כי האילוצים כשלעצמם בדרך-כלל פוסלים הרבה כיווני פתרון ובכך מקטינים משמעותית את מרחב המצבים האפשרי של הבעיה.

* **Simple Temporal Network (STN)**

**את פתרון בעיית האופטימיזציה שתוארה בתחילת תת פרק 2.4 נהוג להציג בגרף, הנקרא גרף** STN**. זהו למעשה גרף המייצג את הממד הזמני של הפעולות שנבצע בפתרון הבעיה, כאשר כל פעולה מיוצגת על-ידי זמן ההתחלה היחסי שלה (כפי שהוגדר בתחילת תת פרק 2.3) והזמן הכולל הדרוש לביצועה.   
~~חשוב לציין כי מעצם היותו פתרון בעיית אופטימיזציה, הפתרון המוצג ב-~~**~~STN~~ **~~הינו פתרון אופטימלי עבור מסלול הפתרון עבורו נפתרה בעיית האופטימיזציה, אם כי אינו המסלול האופטימלי מבין כל המסלולים האפשריים המובילים לפתרון חוקי של הבעיה.~~**

נדגים בעזרת בעיית האופטימיזציה הבאה:  
תיקון פיוז לוקח בדיוק 10 דקות. התיקון חייב להתבצע כאשר הוא מלווה באור הבוקע מנר אשר יכול להיות דלוק למשך 15 דקות לכל היותר, זהו האילוץ.  
יש למצוא פתרון אופטימלי הממזער את הזמן היחסי בו תיקון הפיוז הושלם.

איור 5 -פתרון בעיית האופטימיזציה הנ"ל מוצג בגרף stn.

**פתרון הבעיה הנ"ל מתואר בגרף ה-STN באיור 4 מטה:

**ניתן לראות כי לפי הפתרון שהתקבל בגרף ה-**STN **באיור 4 תיקון הפיוז החל אפסילון זמן לאחר הדלקת הנר ונמשך 10 דקות.**

* + 1. **אלגוריתם ה-Online**

אלגוריתם ה-online מקבל כקלט את אותו קובץ קונפיגורציה שהוכנס לאלגוריתם ה-offline, ואת גרף ה-STN המיוצג כטבלה (פלט ה-offline).

פלט האלגוריתם הוא סדר הפעולות שקרו במהלך היום, כולל ההפרעות והאילוצים שהתרחשו.

אלגוריתם ה-online משמש כ-"מגדל פיקוח", אשר מפקח ומבקר על תוכנית סדר-היום שהתקבלה. האלגוריתם שולח את הפעולות לסביבת ההרצה ומבצע החלטות על בסיס מצב העולם וההפרעות המתרחשות.

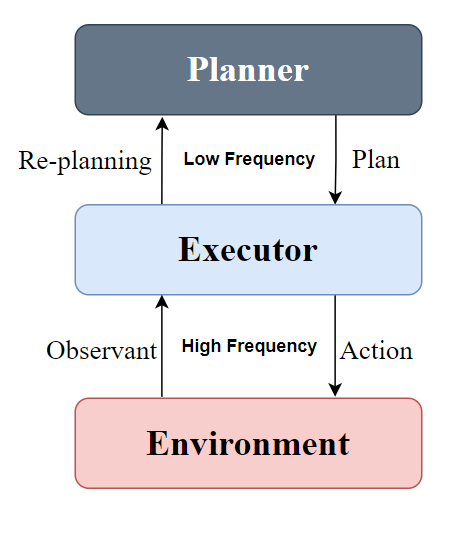
# מטרת הפרויקט

מטרת הפרויקט היא בניית אלגוריתם online לניהול המראות ונחיתות של מטוסים. האלגוריתם מנהל את שדה התעופה בהתאם לתוכנית סדר-היום שהתקבלה מאלגוריתם ה-offline. זאת תוך מעקב והתמודדות עם שינויים ואילוצים המתרחשים בזמן-אמת.

כאשר מתרחשת הפרעה אלגוריתם ה-online צריך לקבוע בזמן-אמת בין האפשרויות הבאות:

1. הוא יכול לרוץ עם ההפרעה והאילוצים שהתרחשו.
2. יש לבצע תיכנון מחדש ביחס למצב העולם בו הוא נמצא.
3. להחליט שהתוכנית לא אפשרית ולשלוח שגיאה.

# תיאור כללי של הבעיה

זוהי סכמה כללית של הבעיה המלאה:

כאשר:

* ה-Planner הוא אלגוריתם ה-offline שמומש בפרויקט הקודם.
* ה-Executor הוא אלגוריתם ה-online.
* ה-Environment הוא הסביבה אשר מדמה את שדה התעופה כולל ההפרעות והאילוצים בזמן-אמת. בסביבה זו, בדקנו את אלגוריתם ה-online.

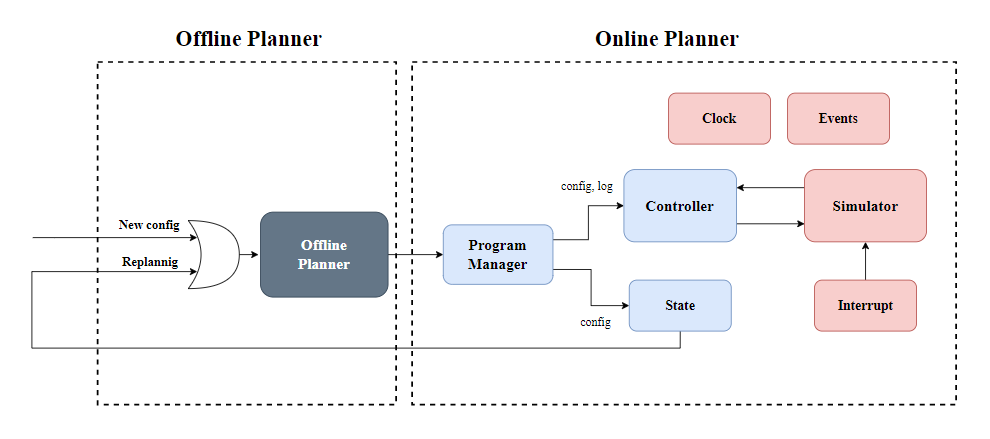
בנוסף כפי שצוין בסכמה – התקשורת בין ה-Planner ל-Executor היא בתדר נמוך, התקשורת מתבצעת מעט פעמים במהלך ריצת אלגוריתם ה-online. לעומת זאת, התקשורת בין ה-Executor ל- Environmentהיא בתדר גבוה, יש תקשורת רבה בין מגדל הפיקוח לשדה התעופה, בדומה לעולם האמיתי.

# הנחות שבוצעו במסגרת הפרויקט

במסגרת הפרויקט הנחנו מספר הנחות:

* לא מתבצעות המראות ונחיתות בו-זמנית. כלומר, במרחב האווירי יש מטוס יחיד בכל רגע נתון.
* זמן התיכנון מחדש זניח ביחס לזמניי הבעיה. לכן, כאשר נדרש לבצע תיכנון מחדש, הקפאנו את מצב העולם ביחס למצב בו הוא נמצא.
* ההפרעה היחידה שביצענו היא עיכוב/דחייה של פעולה (להסביר פה מה זה אומר עיכוב ומה זה דחייה?). כלומר, מתיחה של גרף ה-STN.

# תיאור כללי

להלן הסכמה הכללית של הפרויקט:

תקציר תפקידו של כל בלוק בסכמה:

1. Offline Planner – זה הוא, אלגוריתם ה-Offline. הוא מקבל כקלט קובץ קונפיגורציה היכול להיות אחד משני מקרים:

- קובץ קונפיגורציה חדש המסמל על יום חדש.

- קובץ קונפיגורציה שנבנה מתכנון מחדש (משוב).

הוא מוציא כפלט את תוכנית סדר-היום (גרף ה-STN).

1. Program Manager – מקבל כקלט את תוכנית סדר-היום. הוא מסדר את התוכנית בצורה שתהיה נוחה למודולים השונים. בנוסף, הוא מייצר את המודולים השונים ואת הקשרים ביניהם.
2. Controller – מקבל כקלט את קובץ הקונפיגורציה שהוכנס ל- Offline Planner ואת תוכנית סדר-היום. תפקידו לשמש כמגדל פיקוח ובקרה.
3. State – מייצג את מצב העולם. הוא מציג בכל רגע נתון את מצב המטוסים, המסלולים, המרחב האווירי – מה עוד?
4. Simulator – משמש כמודל לעולם, אשר מריץ את הפעולות. בנוסף, הוא נותן אינדיקציה לתחילה ולסיום של כל פעולה.
5. Interrupt – אחראי על ההפרעות המתרחשות במהלך היום. המודול בוחר באופן רנדומלי איזו פעולה תקבל הפרעה, ומה יהיה אורך ההפרעה.
6. Events – מודול גלובלי המשמש כאמצעי תקשורת בין המודולים השונים – מעין BUS תקשורת.
7. Clock – מודל גלובלי אשר סופר זמן באופן דיסקרטי ומשמש כשעון בסביבה שיצרנו.

בנוסף, ניתן לראות כי צבעי המודולים באיור 3 הינם בהתאמה לאיור 2.

# תיאור מפורט

# תיאור כללי

<סכמת בלוקים כללית, ותיאור בקצרה של כל בלוק>

תיאור מפורט

<פירוט מפורט על כל בלוק, תפקידו, איך בנינו אותו ולמה>

# תוצאות

# סיכום ומסקנות

# נספחים

# רשימת מקורות

1. אלגוריתם זה מומש בפרויקט קודם ע"י הסטודנטים בר מימרן ותום שפירא, בהנחיית איל טייטלר. **אנחנו מתבססים על פרויקט זה/ הפרויקט הנוכחי מתבסס על פרויקט זה.** [↑](#footnote-ref-1)