**פרויקט גמר**

**Peg-in-Hole**

**מנחה:**

ישראל שלהיים

**סטודנטים:**

סער ברקן, ימית גרון

תוכן עניינים

[מבוא 3](#_Toc52365895)

[רקע 3](#_Toc52365896)

[מטרות 3](#_Toc52365897)

[דרישות 3](#_Toc52365898)

[לו"ז 4](#_Toc52365899)

[רקע תאורטי 5](#_Toc52365900)

[סיכום של מאמרים 5](#_Toc52365901)

[סימולציות בסיסות 10](#_Toc52365902)

[סימולציה מלאה – בקרת PID ובקרת עכבה 14](#_Toc52365903)

[אלגוריתם הבקרה 14](#_Toc52365904)

[תוצאות 15](#_Toc52365905)

[מסקנות 18](#_Toc52365906)

[סימולציה מלאה + שיפור ביצועים 18](#_Toc52365907)

[אלגוריתם חיפוש ספירלי 18](#_Toc52365908)

[תוצאות 19](#_Toc52365909)

[מסקנות 26](#_Toc52365910)

[מאמרים 27](#_Toc52365911)

# מבוא

## רקע

הכנסה של פין לתוך קדח(Peg-in-Hole) היא יכולת בסיסית ואינטואיטיבית עבור בני אדם, היא נלמדת באמצעות התנסות על ידי ילדים כבר בגיל צעיר, אך לא כך הדבר עבור רובוטים אשר פועלים בסביבה בה קיימת אי וודאות.

פרויקט זה נועד לסייע למתן מענה עבור צורך גובר בתעשייה לשילוב של רובוטים בתפקידים שכיום מבוצעים על ידי בני אדם בלבד, הוא חלק ממאגד "רובוטיקה בתעשייה" של רשות החדשנות ומבוצע במעבדת SMILE בטכניון.

אחד התפקדים הוא הרכבה של חלקים וגופים בהם יש להתאים פיני מיקום לתוך קדחים, האתגר בפתרון בעיה זו נובע מכך ששיטות הבקרה הקלאסיות (מיקום או כוח) אינן נותנות מענה שלם, לפיכך נבחן שימוש בשיטת בקרה שתופסת תאוצה בשנים האחרונות והיא בקרת עכבה (אימפדנס).

## מטרות

1. יצירת סימולציה של Peg-in-Hole.
2. ביצוע Peg-in-Hole ע"י זרוע רובוטית תוך שימוש בבקרת אימפדנס בתנאים אידיאליים.
3. ביצוע שיפור הביצועים ע"י חיפוש ספירלי

## דרישות

* הצגת היתרונות והחסרונות של שימוש בבקרת עכבה.
* ביצוע סימולציה של Peg-in-Hole באמצעות בקרת מיקום.
* ביצוע סימולציה של Peg-in-Hole באמצעות בקרת עכבה.
* ביצוע סימולציה משולבת חיפוש ספירלי והשוואה בין בקרת מיקום לבקרת עכבה

## לו"ז

|  |  |  |  |
| --- | --- | --- | --- |
| מס' | משימה | תיאור | משך (שבועות) |
| 1 | חומר תאורטי | קריאה וסיכום מאמרים | 3 |
| 2 | רקע תיאורטי | כתיבת רקע תיאורטי | 2 |
| 3 | סימולציה בסיסית בקרת מיקום | דרגת חופש אחת | 1 |
| 4 | סימולציה בקרת אימפדנס בסיסית | דרגת חופש אחת | 1 |
| 5 | חוג בקרה בסיסי |  | 1 |
| 6 | סימולציה 2 דרגות חופש |  | 2 |
| 7 | MuJoCo | לימוד סביבת עבודה | 2 |
| 8 | מערכת ניסוי | תכנון בנייה והתאמה לזרוע | 1 |
| 9 | סימולציה | בסביבת  MuJoCo | 2 |
| 10 | מודלים תיאורטיים | פיתוח משוואות תנועה ודינמיקה | 3 |
| 11 | חוג בקרה | יצירת חוג ואופן מימושו | 3 |
| 12 | סימולציה מלאה | בקרת אימפדנס | 4 |
| 13 | סקירת ספרות על שיפורים לביצועי המערכת | מציאת מאמרים | 3 |
| 14 | סימולציה מלאה | עם אלגוריתם החיפוש | 2 |
| 16 | סיכום ומסקנות, השלמת הדו"ח |  | 3 |

# רקע תאורטי

## סיכום של מאמרים

סיכום המאמר An introductory review of active compliant control:

נקודות עיקריות :

* סקירה של שיטות בקרת היענות אקטיביות המתמקדות במערכות קשיחות.
* עקב הופעה של יישומים חדשים רבים בתחום הרובוטיקה חל גידול משמעותי בכמות המאמרים בתחום, המאמר נועד לסקור את המאמרים ולסכמם באופן מסודר.
* המאמר מחלק ל - 4 שיטות עיקריות : אימפדנס ואדמיטנס , בקרת מיקום/כוח מקבילה או היברידית.
* המאמר מציע סכמת בחירת שיטת בקרה לפי אפליקציה ואילוצים.

גורמים להתפתחות הצורך בתחום:

* עלייה בכמות שיתוף פעולה בין רובוטים לאנשים.
* צרכים בתחום הרפואה למשל שלדים חיצוניים, פרוטזות ורובוטים לניתוחים.
* רובוטים הולכים.
* גידול בכמות הרובוטים התעשייתיים אשר מגיעים מזוודים עם חיישני כוח/מומנט.

בקרת היענות הינה תת תחום של בקרת כוח מתמשכת(משוב), היא מאפשרת ביצוע שינויים במאפייני ההיענות של מערכת הבקרה וכתוצאה מכך לשנות את התגובה הדינמית של המערכת.

תחום בקרת היענות התפתח מהגישה ההיברידית ומקבילית אל שיטת האדמיטנס והאימפדנס.

תכונת ההיענות ניתנת להשגה ושינויי בשתי דרכים

* פסיבי ע"י שינוי תכונות מובנות של המכניזם.
* אקטיבי ע"י בקרה.

בקרה ישירה –לולאת משוב הכוח נסגרת על ידי בקר כוח.

בקרה לא ישירה – שליטת הכוח מתממשת באמצעות בקרת תנועה. לולאת בקרת תנועה פנימית או חיצונית זו מאפשרת ליצור קשר רצוי בין תנועת המערכת לכוח.

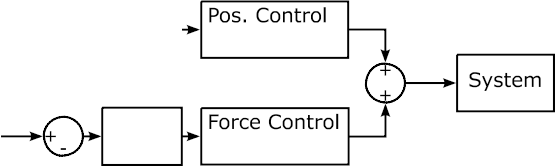
בקרת כוח מפורש ולא מפורש-בבקרת כוח מפורש השליטה מאופיינת על ידי משוב על הכוח בעזרת חיישן כוח, ואילו במקרה הלא מפורש המשוב מסופק על ידי בקרה בלולאה פתוחה ובאמצעות ההבדל בין המטרה לתזוזה המדודה.

בקרה לא מפורשת מתאימה רק למערכות בעלות Backdrivability וחיכוך זניח, כמו כן עקב מאפייני תגובת ההלם מומלצת למערכות איטיות ומשטחים רכים , חייבים מודל כדי להתגבר על החוסר נתונים מהחיישן.

**סוגי בקרת היענות:**

בקרת כוח/מיקום היברידית:

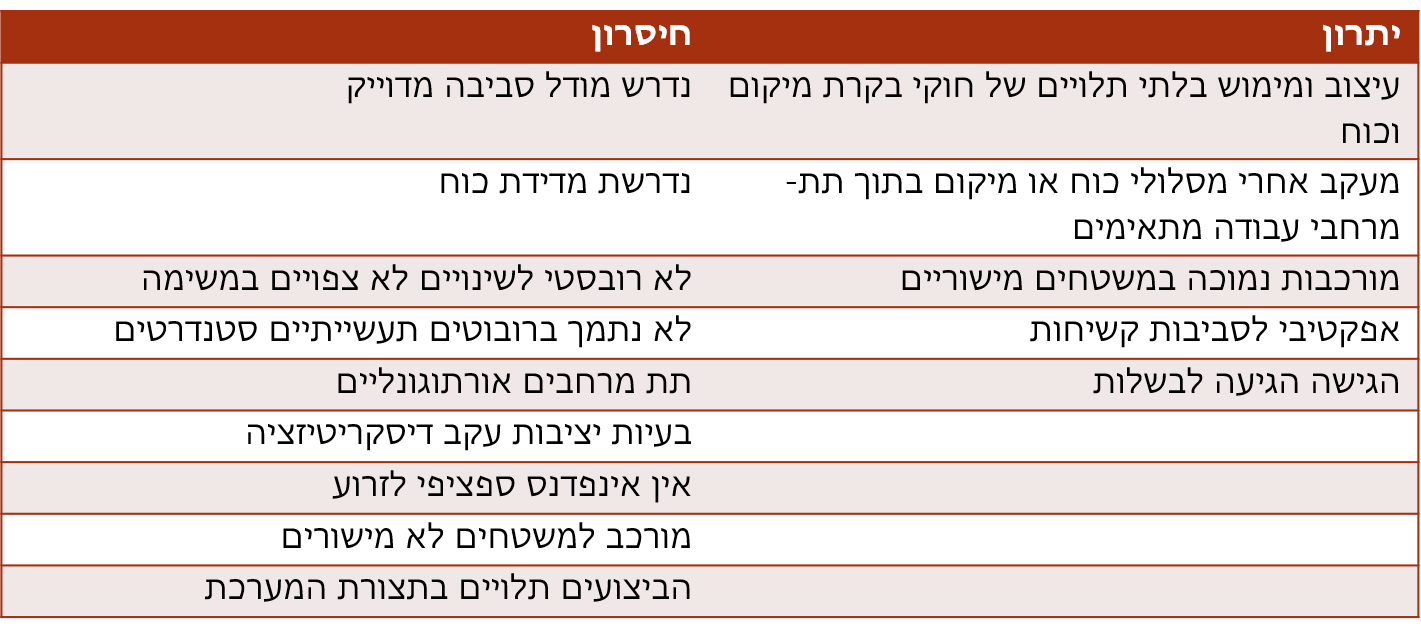
סכמת בקרה



חוק בקרה

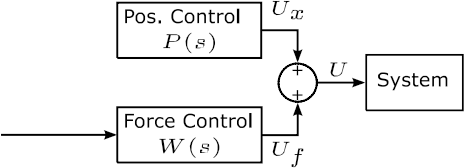
הסבר כללי

שיטה זו מחלקת את מרחב העבודה לתת-מרחבי עבודה אורתוגונליים משלימים, אשר בכל אחד מהם מתבצעת בקרת תנועה או בקרת כוח.



בקרת כוח/מיקום מקבילה:

סכמת בקרה



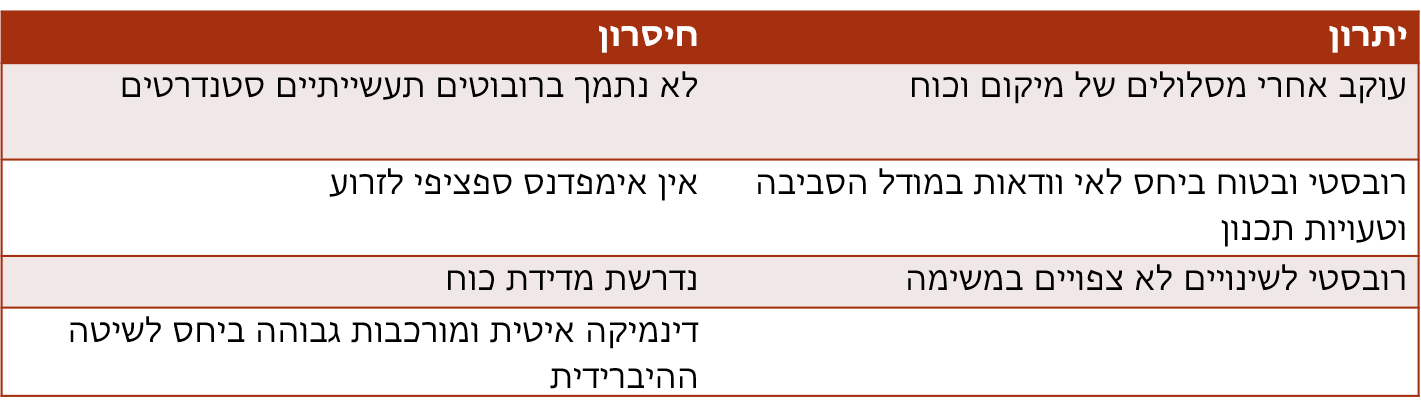
חוק בקרה

הסבר כללי

שיטה זו מנסה לעקוב אחרי מסלול תנועה נתון עבור כיוונים לא מאולצים ומנסה לשלוט בכוחות המגע אשר נובעים מכיוונים מאולצים.

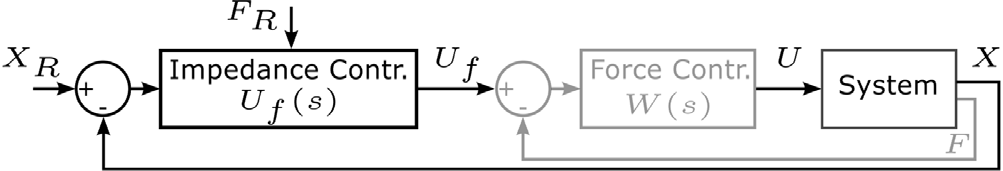
בניגוד לבקרה היברידית בבקרה מקבילה ניתן לבצע סופרפוזיציה בין היציאות של בקרי המיקום והכוח ובשל כך שניהם יכולים לפעול על אותו הכיוון

בשיטה זו קיימים שני בקרים במקביל כאשר ע"פ הספרות ממליצים על בקר PD עבור המיקום ו- PI עבור הכוח, בשל כך האינטגרטור בבקר הכוח גורם לכך ששגיאת הכוח תיסגר על חשבון שגיאת המיקום.



בקרת אימפדנס:

סכמת בקרה



\*בשחור הסכמה הבלתי-מפורשת, באפור ושחור הסכמה המפורשת.

חוק בקרה

האימפדנס מיוצג ע"י פולינום מסדר m.

הסבר כללי

שיטה זו מתמקדת במימוש יחס מטרה בין כוח לתנועה אך לא בהכרח עוקבת אחריהם במדויק.

כלומר יחס האימפדנס מגדיר פשרה בין דיוק במיקום לגודל הכוח המופעל.

השיטה לא דורשת מודל דינמי אך מודל כזה יכול לשפר את ביצועי המערכת.

במרחב החופשי מתנהגת כמו בקרת מיקום בעוד שבסביבה קשיחה מתנהגת בדומה לבקרת כוח מפורשת.

בתיאוריה הסדר של האימפדנס יכול להיות אך במאמרים שנסקרו לא הוזכרו מימושים פרקטיים מעל סדר זה.

סדר 0 –

בקר קשיחות, שקול לבקר מיקום P(במקרה הבלתי-מפורש).

יותר פשוט חישובית ויותר רגיש לאי וודאות אך יכול להקטין יציבות ודיוק במקרה המפורש.

המקרה הבלתי מפורש הינו פסיבי.

סדר 1 –

בקר ריסון, שקול לבקר מיקום PD(במקרה הבלתי-מפורש).

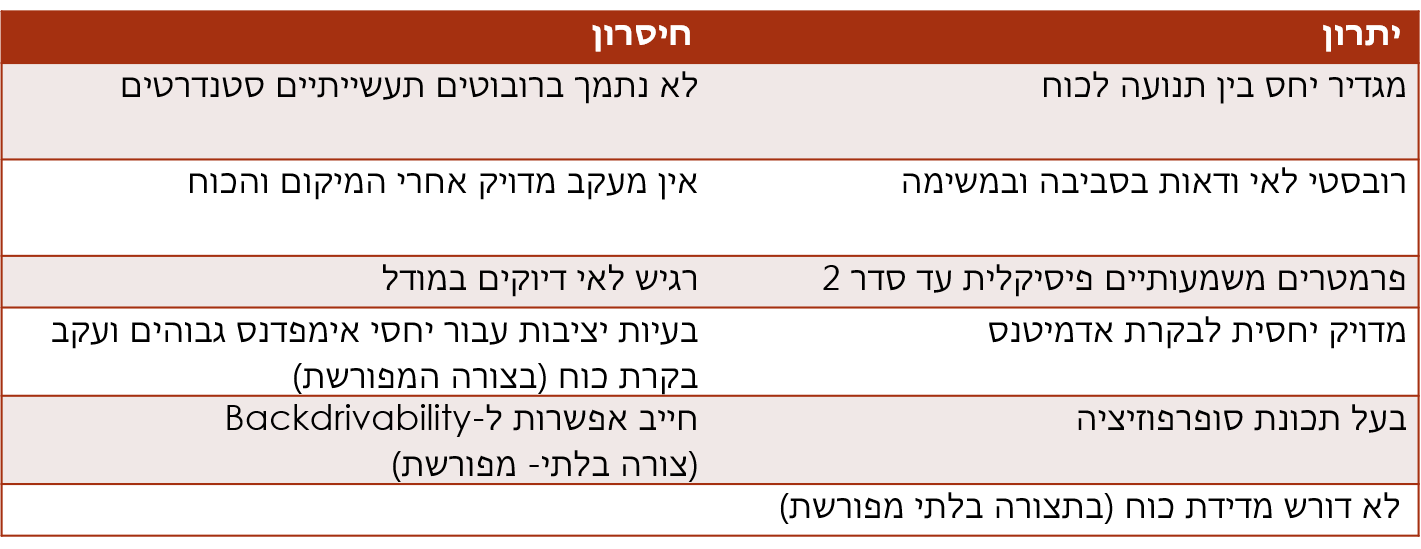
מקדם הריסון מפצה על ריסון אקטיבי טבעי.

סדר 2 –

ניתן לשנות את דינמיקת המסה(או האינרציה) של המערכת על מנת להשיג מסת מטרה .

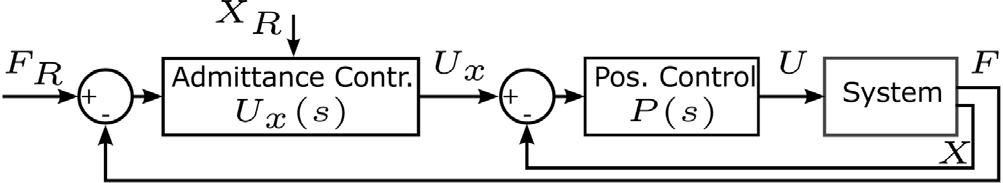
שקול לבקר מיקום PID(במקרה הבלתי-מפורש).

עד סדר זה יש משמעות לפרמטרים .



בקרת אדמיטנס:

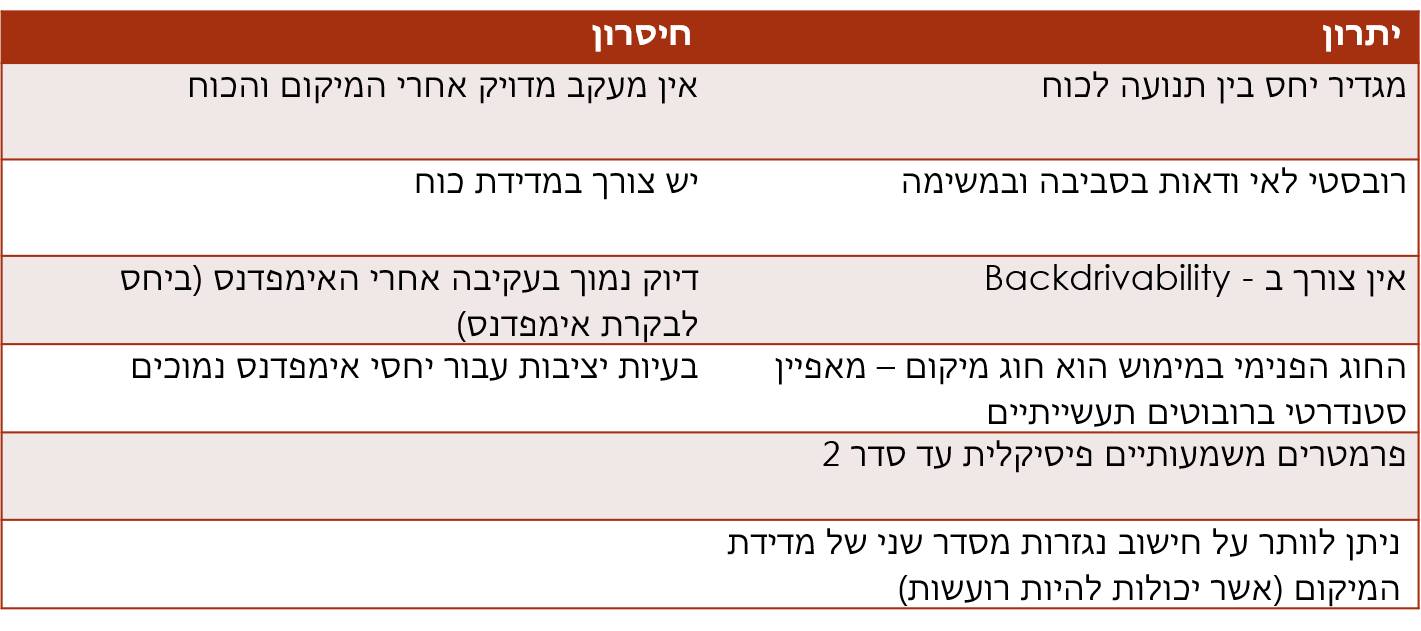
סכמת בקרה



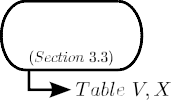
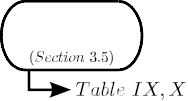
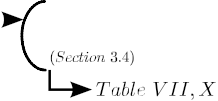
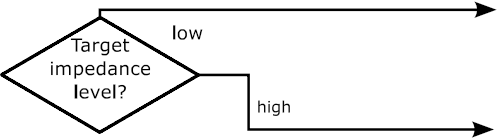
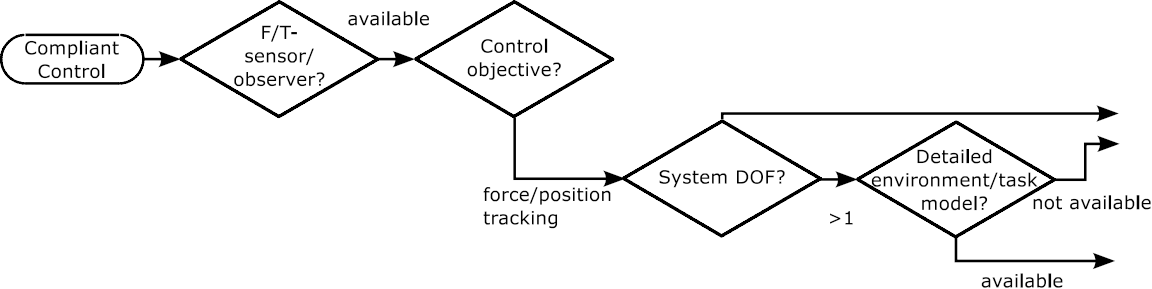
חוק בקרה

הסבר כללי

שיטה זו מממשת רעיון דומה לבקרת אימפדנס(לעיתים מכונה בקרת אימפדנס עם משוב כוח) אך שונה ממנה ביחסי המטרה בין הכוח לתנועה ובמאפייני מפתח במימוש בחוג הפנימי.



סכמת בחירה:



סיכום:

המאמר נותן לנו אפשרות נוחה לבחירת שיטת הבקרה לפי האפליקציה הרצויה.

מסכם ומתמצת את סוגי הבקרות ונותן הסברים נוספים על איך אפשר להתגבר על חלק מהחסרונות בשיטות ואיך ניתן לשלב ביניהם ליצירת מערכת אשר מתאימה לאפליקציות אחרות.

## סימולציות בסיסות

סימולציה פשוטה של רובוט 2 דרגות חופש ובקר PD + Inverse Dynamics:

בנינו תוכנית מטלאב אשר מקבלת 2 נקודות ובונה מסלול של קו ישר .

המרנו את המסלול שבנינו במערכת העולם בעזרת קינמטיקה הפוכה למסלול במערכת המפרקים.

לאחר מכאן הכנסנו את תנאי ההתחלה ואת הדינמיקה של הרובוט לפותרן נומרי (ode45) ביחד עם חוק הבקרה :



כיילנו את קבועי הבקרה עד לצמצום השגיאה במיקום וקבלת עקיבה טובה.

קבועי הבקרה אשר סיפקו ביצועים טובים אחרי כמה איטרצייות :



ניתן לראות את ביצועי הרובוט:

**מפרק 1**

תמונה שמכילה מפה, טקסט

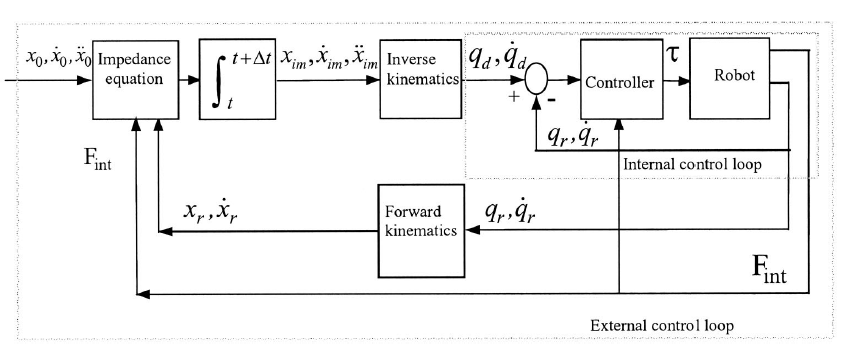
התיאור נוצר באופן אוטומטי

תמונה שמכילה טקסט, מפה

התיאור נוצר באופן אוטומטי**מפרק 2**

לאחר מכאן הוספנו חוג בקרה חיצוני אשר ממש בקרת אימפדנס , המכניס את הערכים המתקבלים על בקר המיקום אשר מימשנו .

בקרת האימפדנס בוצעה ע"פ הסכמה Instantaneous Model Impedance Control כפי שהוצע מאמר Accuracy/Robustness Dilemma in Impedance Control



חוק בקרה:



כאשר:



ביצענו מידול של קיר , הצבנו את יעד המטרה לאחר הקיר ובחנו את ביצועי הרובוט עם וללא הפרעת הקיר.

כיילנו את קבועי הבקרה עד לצמצום השגיאה במיקום וקבלת עקיבה טובה.

קבועי הבקרה אשר סיפקו ביצועים טובים אחרי כמה איטרצייות :



נבדק שני מצבים פעם אחד במצב חופשי ללא קיר:



מפרק 2

ומצב שני עם קיר:

מפרק 1

מפרק 2

# סימולציה מלאה – בקרת PID ובקרת עכבה

## אלגוריתם הבקרה

האלגוריתם שתיכננו פועל לפי דיאגרמת הפעולה הבאה:

קבלת נקודות מהמשתמש

בניית מסלול קרטזי

המרת המסלול למרחב המפרקים

עקיבה אחר המסלול באמצעות בקר נבחר

(PID , impedance)

הפקת גרפים

(כוחות , מומנטים , זוויות וכו')

**בניית מסלול קרטזי :**

בנינו פונקציה אשר מרכזת את תהליך בניית המסלול הקרטזי, הפונקציה נעזרת במספר פונקציות עזר.

execute\_time – מחשבת זמן ביצוע תזוזה בין שתי נקודות בהתאם למרחק ביניהן ולפרמטר המהירות הנקבע ע"י המשתמש.

motion\_plan – בונה מסלול קרטזי בין שני נקודות ע"י פולינום מסדר 5 כאשר המהירות והתאוצה מתאפסים בתחילת התנועה ובסופה (MINIMUM-JERK TRAJECTORY).

3. xyz\_acc , Xyz velocity - בונה את המהירות והתאוצה הנדרשים לביצוע המסלול

**המרת המסלול למרחב המפרקים:**

לאחר שקיים מסלול תנועה קרטזי נרצה להמירו לתנועה במרחב המפרקים, לשם כך יצרנו את הפונקציה build\_trajectory

פתרנו את הקינמטיקה הישירה וההפוכה על פי שיטת DH ובנינו פונקציות אשר ממשות את הפתרון

Inv\_kin – פונקצית קינמטיקה הפוכה, מקבלת מטריצת טרנספורמציה הומוגנית ומחזירה פתרון במרחב המפרקים. ניתן לבחור ידנית בין הפתרונות (סה"כ 8)

Forward – פונקצית קינמטיקה ישירה, מקבלת וקטור זוויות מפרקים ומחזירה את הטרנספורמציה בין הבסיס לתפסנית.

## תוצאות

**עקיבה אחר המסלול באמצעות בקר PID:**

* תחילה השתמשנו בנתוני הסימולטור למציאת פיצוי גרביטציה ומהירויות (מטריצות B,G)
* מימשנו בקר PID הסוגר חוג על זוויות ומהירויות המפרקים , אות הבקרה הינו המומנט הניתן למפרקים.
* תמונה שמכילה טקסט, מפה, מחשב, שולחן

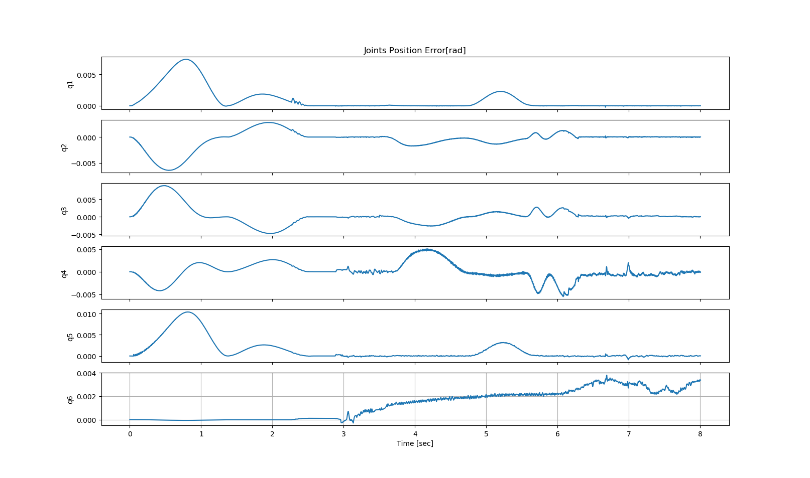
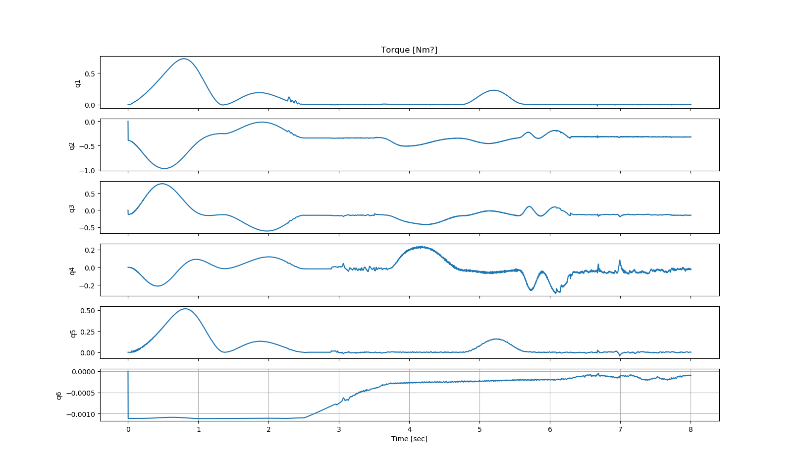
  התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט, מפה, מחשב, קבוצה

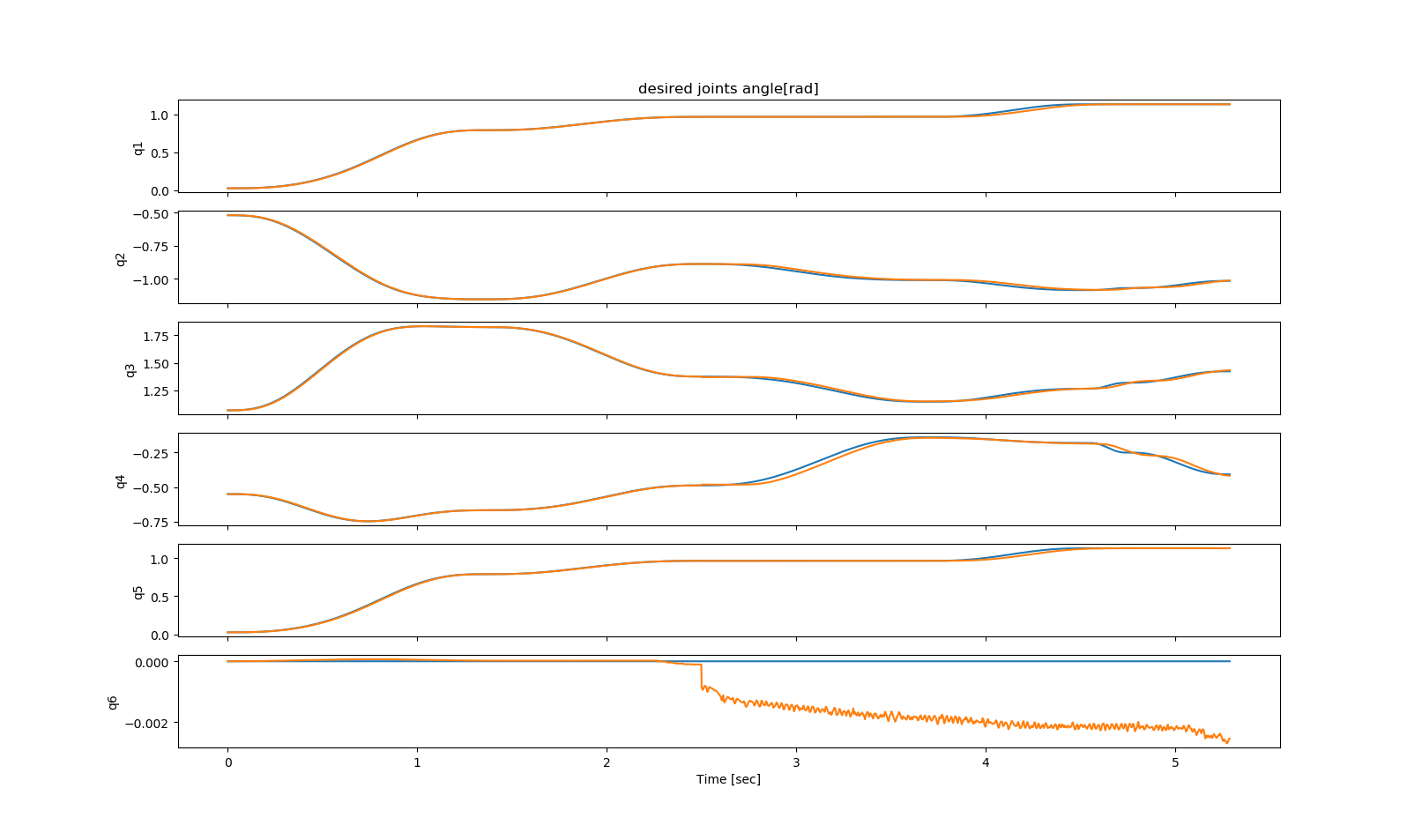
  התיאור נוצר באופן אוטומטיכיוונו עבור כל מפרק את קבועי הבקרה לקבלת עקיבה טובה לאורך המסלול.

תמונה שמכילה טקסט, מפה, ממולא, שולחן

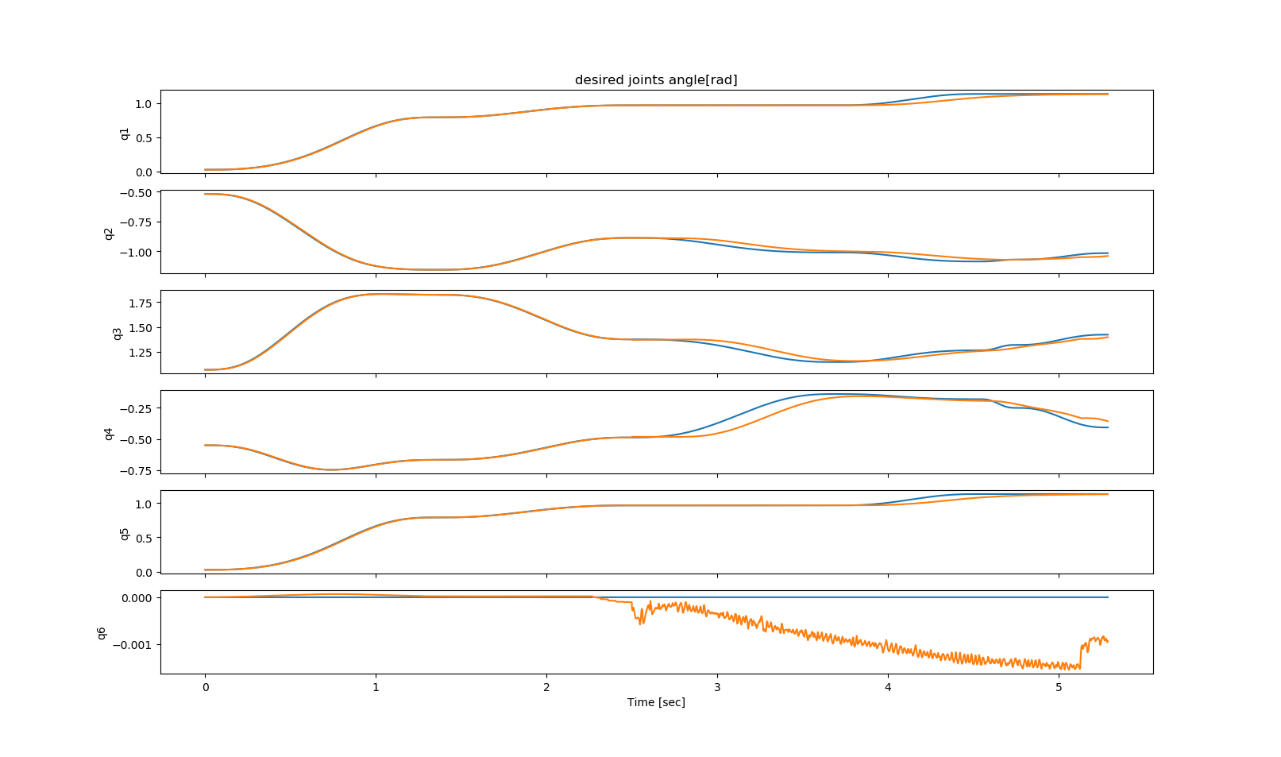
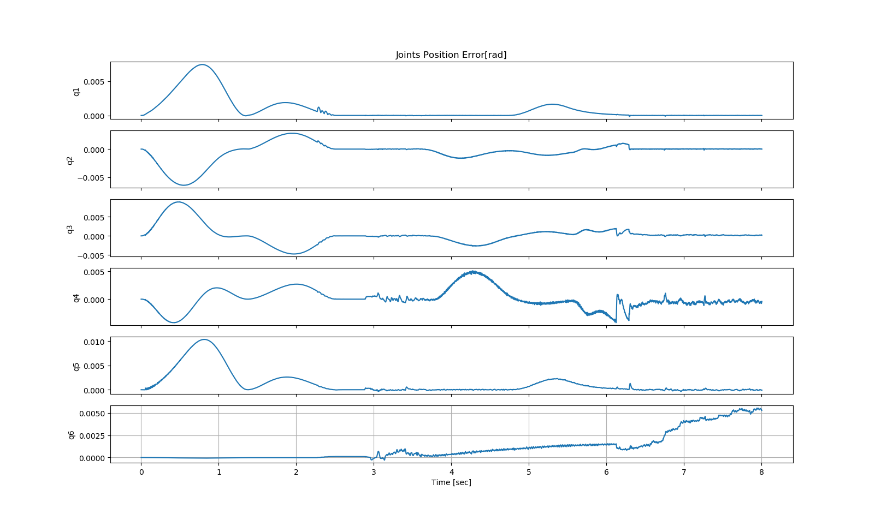
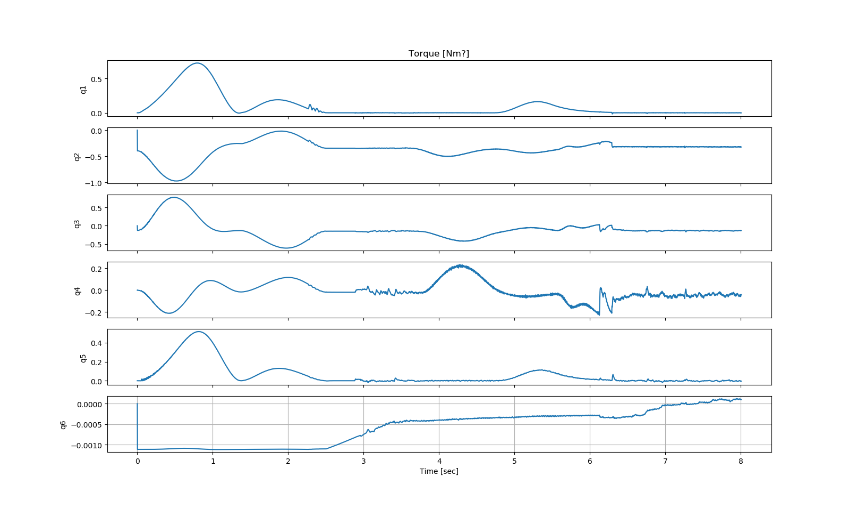
התיאור נוצר באופן אוטומטי

**עקיבה אחר המסלול באמצעות בקר impedance:**

* תחילה השתמשנו בנתוני הסימולטור למציאת פיצוי גרביטציה ומהירויות (מטריצות B,G)
* בחרנו לממש את האימפדנס לפי סכמת IM-IC
* ****יצרנו פונקציה המממשת את בקר האימפדנס הפונקציה imic המקבלת את הקריאה העדכנית של זוויות ומהירויות המפרקים , את המיקום ומהירות הקרטזים הרצויים ואת קריאת חיישן הכוח הממוקם על גליל המשחק.

****

בשביל לוודא כי בקרת העכבה שביצענו מתנהגת בצורה הגיונית , הגדלנו את מטריצת המסה בתוך בקרת האימפדנס ובחנו את התוצאות.

****

כמו שציפינו רואים בגרפים כי העקיבה פחות טובה אחרי הגדלת מטריצת המסה.

**תמונה שמכילה טקסט, מפה, שולחן

התיאור נוצר באופן אוטומטיההבדל בכוח בין השיטות:**

**אימפדנס**

**תמונה שמכילה טקסט, מפה, מביט, צילום

התיאור נוצר באופן אוטומטי**

**PID**

ניתן לראות ירידה בסדר גודל של הכוח בין השימוש בשיטת PID לבין שיטת האימפדנס .

## מסקנות

אחרי בחינת מגבלות של שתי השיטות, גילינו כי בקרת עכבה מתמודדת בצורה טובה יותר עם שגיאות במיקום לעומת PID, אך גם היא מצליחה להתמודד רק עם שגיאות מאוד קטנות עם השגיאה הליניארית גדולה מ0.001 מילימטר מהחור החלק לא יצליח להיכנס.

# סימולציה מלאה + שיפור ביצועים

## אלגוריתם חיפוש ספירלי

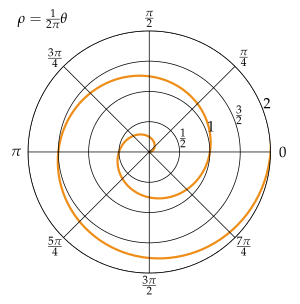
בחנו מספר מאמרים אשר סוקרים דרכים רבות לשיפור הצלחת משימת peg in the hole ושיפור הרובסטיות לתנאי התחלה שונים. [1][2][3][4]

לאחר הסקירה והבנה של אופציות לשיפור נתוני ההצלחה בחרנו לממש חיפוש בצורת ספירלה , אשר ישפר את נתוני הצלחת השיטה ויגדיל את טולרנס השגיאה איתו האלגוריתם יכול להתמודד.

אלגוריתם החיפוש שיצרנו מבוסס על משוואת הספירלה של ארכימדס:

בקורדינטות פולריות : 

שינוי a יגרום לסיבוב בספירלה והפרמטר b יגרום לשינוי במרחק בין הסיבובים.



ספירלת ארכימדס מוצגת על מערכת קואורדינטות קוטביות. קרן היוצאת ממוקד הספירלה פוגשת סיבובים עוקבים של הספירלה בנקודות הרחוקות זו מזו מרחק קבוע. ככל שהזווית גדלה כך גדל הרדיוס של בספירלה.

האלגוריתם מקבל את נקודת המגע בה חלק המשחק פוגע בקובייה, הנקודה מוגדרת כנקודת מרכז הספירלה a ו b הם נתונים מספריים קבועים לאורך כל מהלך החיפוש והזוויות משתנה בכל איטרצייה עד להכנסת החלק לחור אם הוא נמצא , במקרה והחלק לא מגיעה לחור או מפספס אותו אלגוריתם החיפוש ימשיך לפעול עד עצירת הסימולציה.

אלגוריתם החיפוש מבצע תנועה אך ורק במישור XY במישור Z ישנה לחיצה חלשה לכיוון הקובייה בכדאי לדעת מתי חלק המשחק נכנס לחור והמשימה מצליחה.

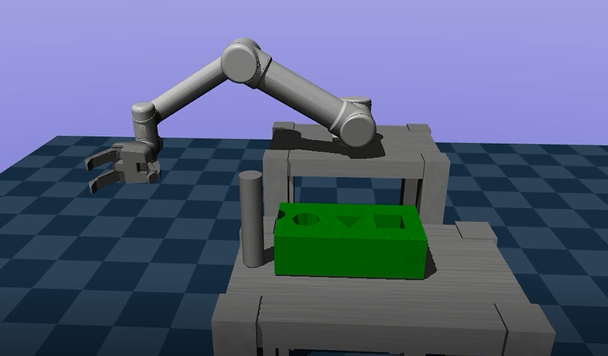
ניתן היה לראות בסימולציות כי חיפוש זה באמת משפר בצורה ניכרת את רובוסטיות הצלחת המשימה כאשר ישנם תנאי התחלה שונים ושגיאות ליניאריות גדולות.

מטרת הוספת החיפוש הספירלי הינו שיפור ההצלחות ושיפור מבחינת הצלחה במצבים בהם ישנה שגאה ליניארית .

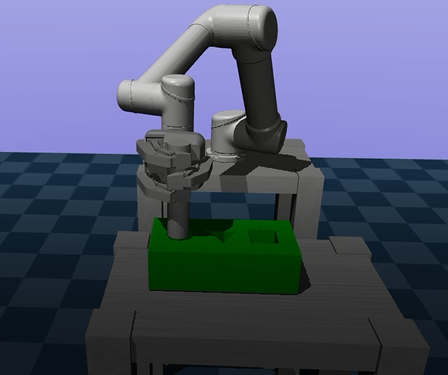
אנו מניחים כי החיפוש ישפר באופן ניכר את אחוז הצלחות המשימה.

## תוצאות

התחלת הסימולציה:

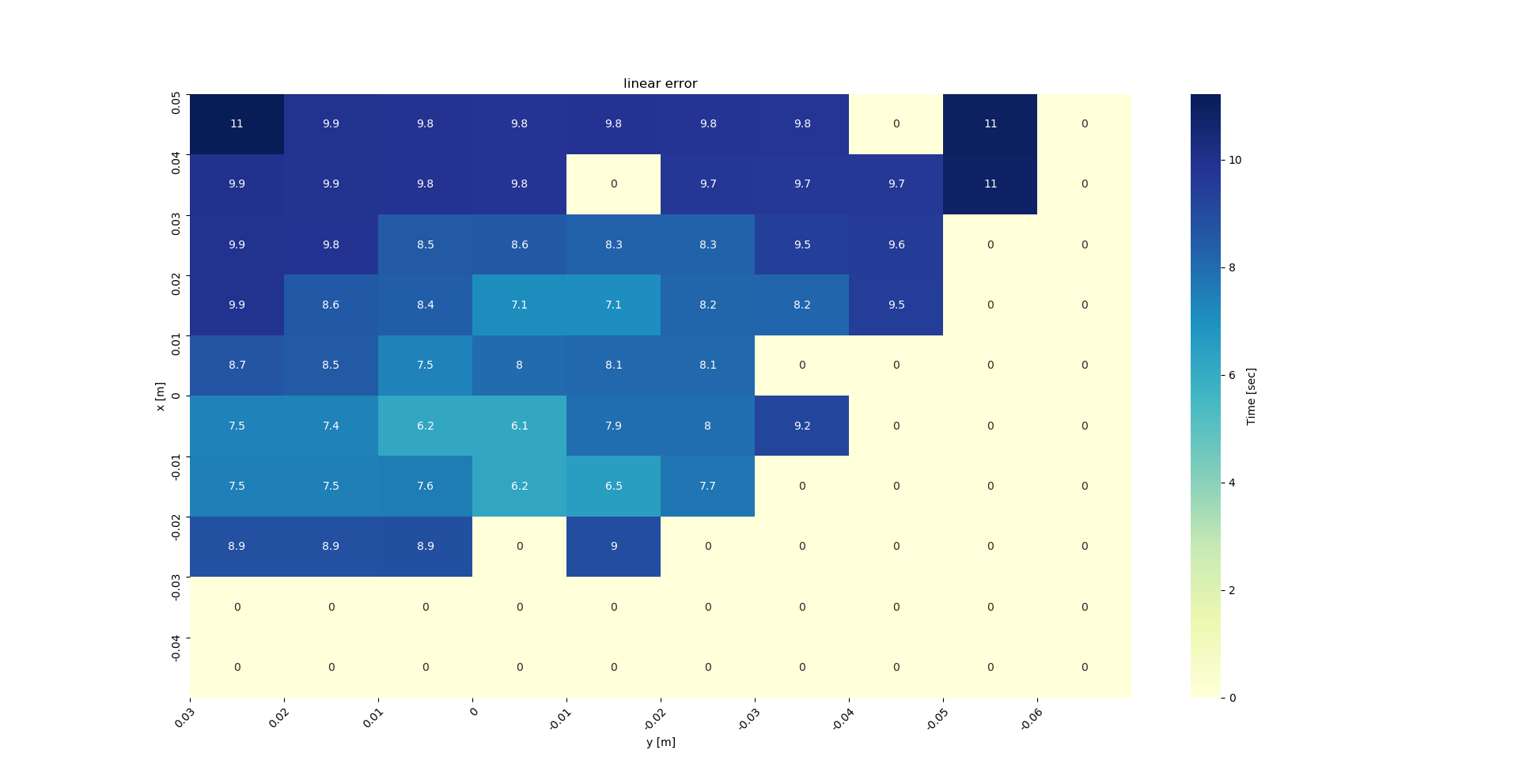


סיום הסימולציה במצב הצלחה:



זמן הניסוי נמדד מתחילת התנועה עד הכנסת החלק לחור או במקרה של אי הצלחה, זמן הניסוי מוגדר להיות 0.

בניסוי זה בחנו את השפעת השגיאה הליניארית בצירים XY על זמני הצלחת/אי-הצלחה של המשימה, כאשר השתמשנו בבקרת עכבה לצורך בקרת המערכת, ונתוני הספירלה קבועים והם a=b=0.0012.

הגרף הבא מציג את תוצאות הניסוי:

גרף 1 – הזמן שלוקח להכניס את החלק לחור כתלות בשגיאה הליניארית בציר X ו Y. 0 מיצג מצב של אי הצלחת המשימה.

מגרף זה ניתן לראות את זמני ההצלחה של הכנסת חלק המשחק לתוך החור כאשר טולרנסי השגיאה הליניארית משתנים.

ניתן להבחין כי באמת כאשר אין שגיאה החלק ניכנס ישירות לתוך מקומו (זמן הכניסה המינימלי הכולל את כל התנועה הינו 6.1 שניות) , ככל שמתרחקים והשגיאה הליניארית גדלה כך לוקח יותר זמן להגיעה ולמצוא את המקום המתאים וזמן החיפוש מתחיל להיות משמעותי.

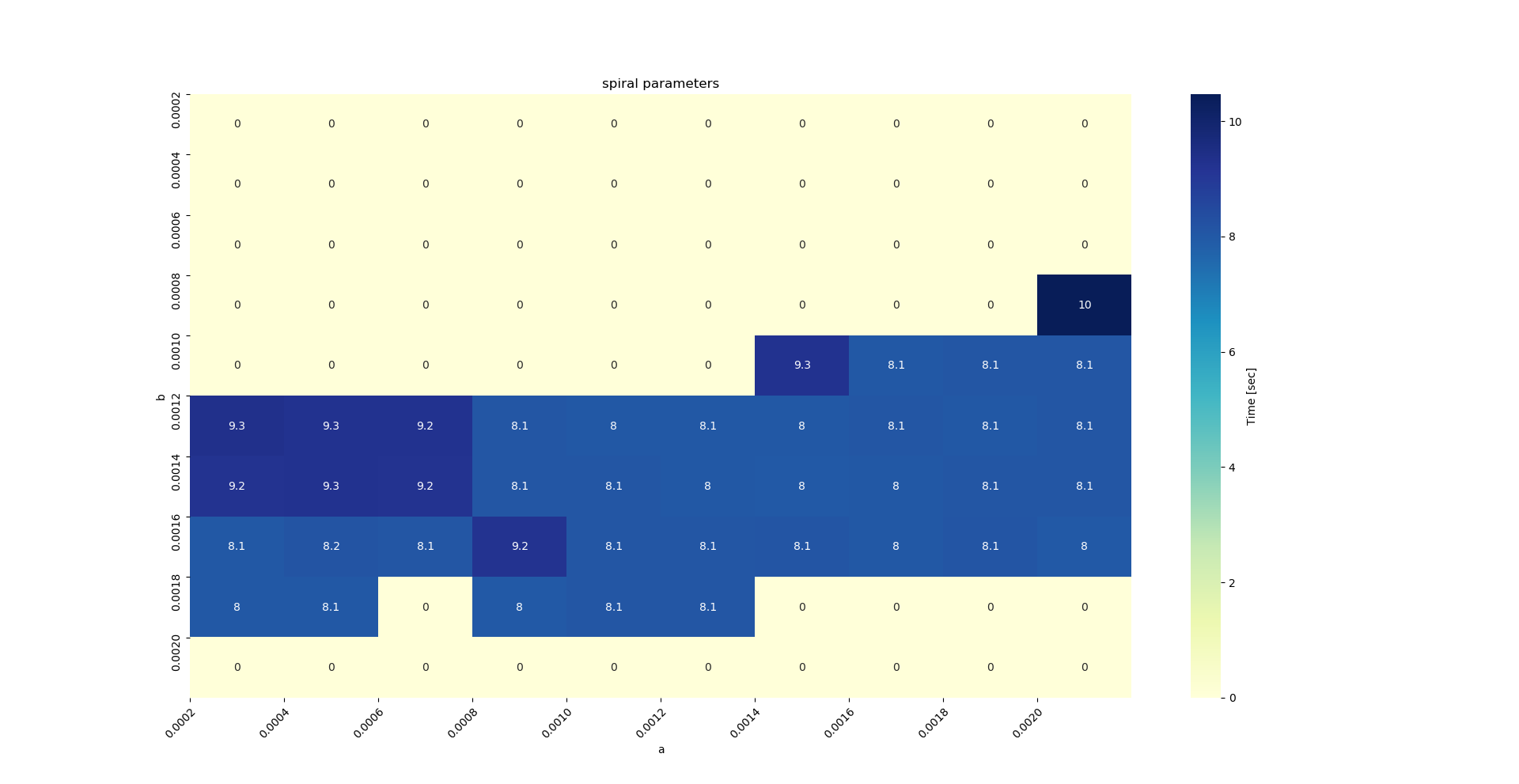
**הטווח בו נראה כי היה הצלחה טובה של המשימה הוא:**

שגיאה ליניארית בציר x בין -0.03 ל- 0.05מילימטר, שגיאה ליניארית בציר y בין -0.03 ל-0.05 מילימטר.

כאשר זמן ההצלחה הקצר ביותר בתחום זה הינו 6.1 שניות והזמן הכי ארוך הינו 11 שניות בהוצאת המקרים בהם המשימה לא הצליחה.

בניסוי הבא בדקנו את השפעת שינוי פרמטרי הספירלה a,b על משך זמן הצלחת המשימה, כאשר השתמשנו בבקרת עכבה לצורך בקרת המערכת והשגיאה הליניארית קבוע.

הגרף הבא מציג את תוצאות הניסוי:



גרף 2 - הזמן שלוקח להכניס את החלק לחור כתלות בפרמטרי הספירלה a ו-b , 0 מיצג מצב של אי הצלחת המשימה.

ניתן לראות כי הפרמטרים משפיעים מאוד וישנו תחום גדול של פרמטרים אשר החלק בכלל לא מצליח להיכנס לחור, לכן צריך לבחור בקפידה את הפרמטרים ואף לשקול לבחור את הפרמטרים כל פעם לפי טווח השגיאה הצפויה, אם ניתן לדעת אותה מראש.

**הטווח בו נראה כי היה הצלחה טובה של המשימה הוא:**

ערך הפרמטר a בין 0.0002 ל-0.002, ערך הפרמטר b בין 0.0012 ל-0.0018.

כאשר זמן ההצלחה הקצר ביותר בתחום זה הינו 8.1 שניות והזמן הכי ארוך הינו 9.2 שניות בהוצאת המקרים בהם המשימה לא הצליחה.

בגרפים הבאים נציג את מהלך התנועה המלאה בציר הזמן, עבור פרמטרים אופטימליים. כאשר הבקרה שאנו משתמשים בא הינה בקרת עכבה.

קבועי בקר PID בבקרת מקום ובקרת עכבה:

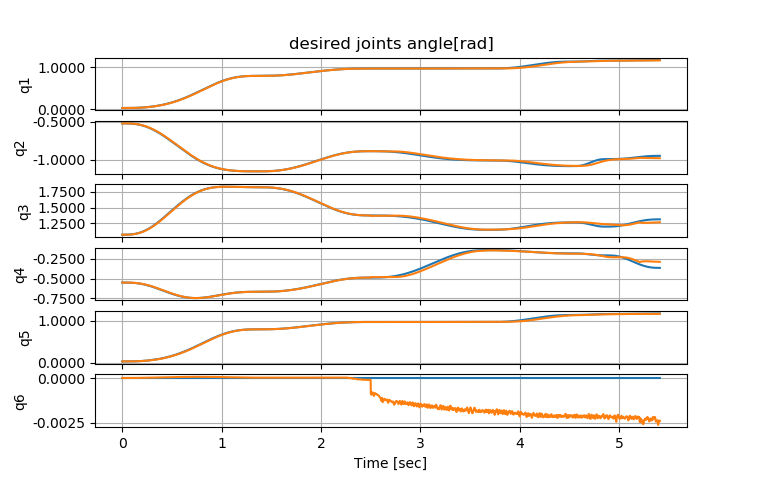
|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| אימפדנס | 0 | diag(100,100,100,50,50,0.1) | diag(0,0,0,0,0,0) |
| מיקום | 0 | diag(100,100,100,50,50,0.1) | diag(0,0,0,0,0,0) |

מטריצות האימפדנס:

|  |  |  |
| --- | --- | --- |
| M | B | K |
| eye(3)\*0.2 | eye(3)\*0 | eye(3)\*4000 |

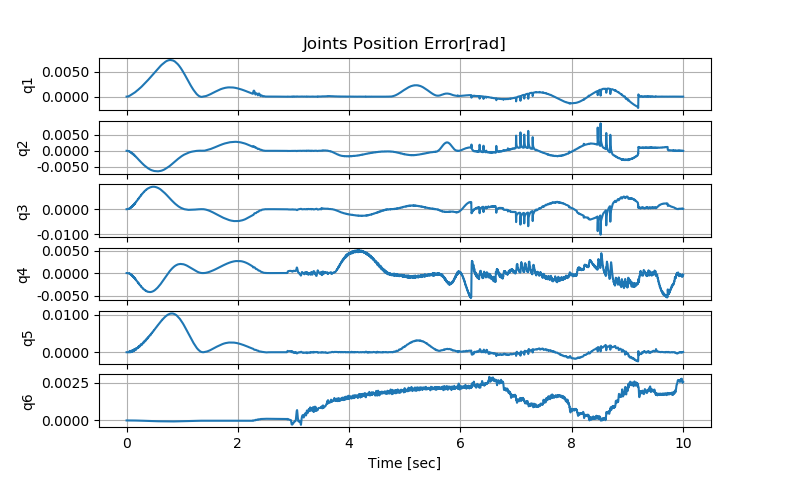
נתוני הספירלה :

|  |  |  |  |
| --- | --- | --- | --- |
| a | b | שגיאה ליניארית בx | שגיאה ליניארית בy |
| 0.0012 | 0.0012 | 0 | 0.002 |



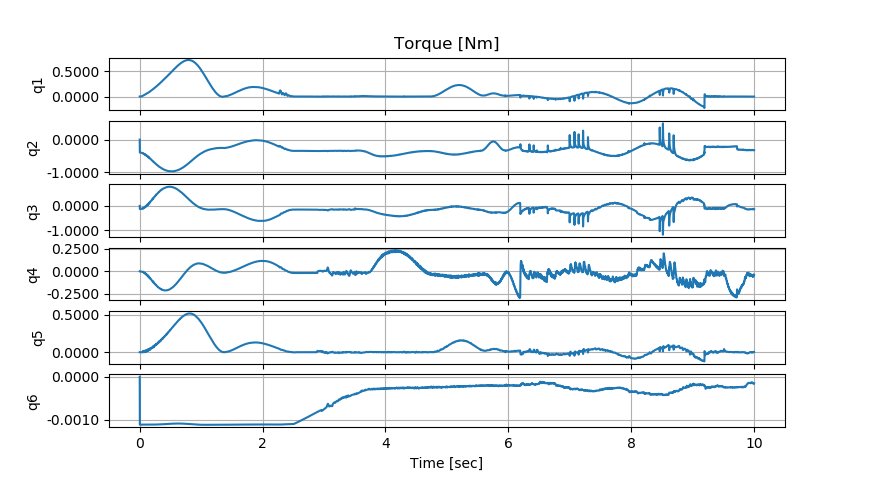
גרף 1- זוויות מפרקים מתוכננות(כחול) לעומת הזוויות בפועל(כתום)

ניתן לראות כי המסלול המתוכנן והמסלול בפועל מאוד דומים אחד לשני והעקיבה טובה.



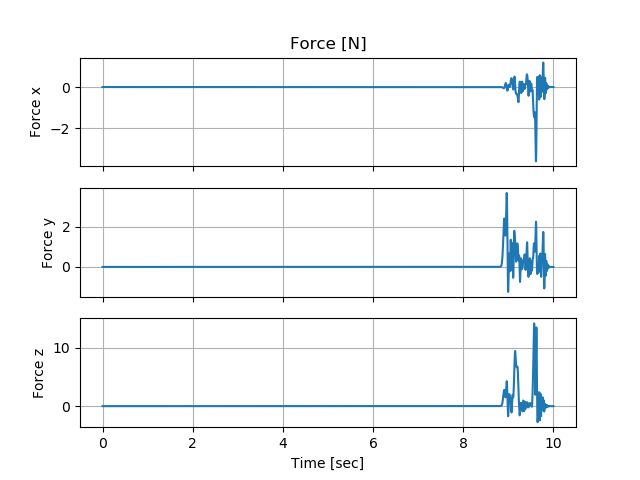
גרף 2 – שגיאת הזווית במפרקים כתלות בזמן

שגיאות המפרקים במהלך התנועה זניחות כמו שהיה ניתן להסיק מהגרף הקודם

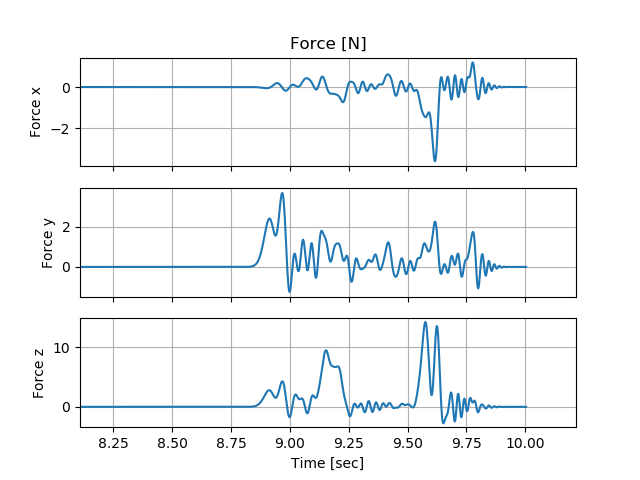


גרף 3 – המומנט המתפתח במפרקים כתלות בזמן

הכוחות המתפתחים במהלך התנועה על הגליל הינם:



גרף 4 – כוחות בצירים השונים כתלות בזמן

נתמקד בחלק הרלוונטי

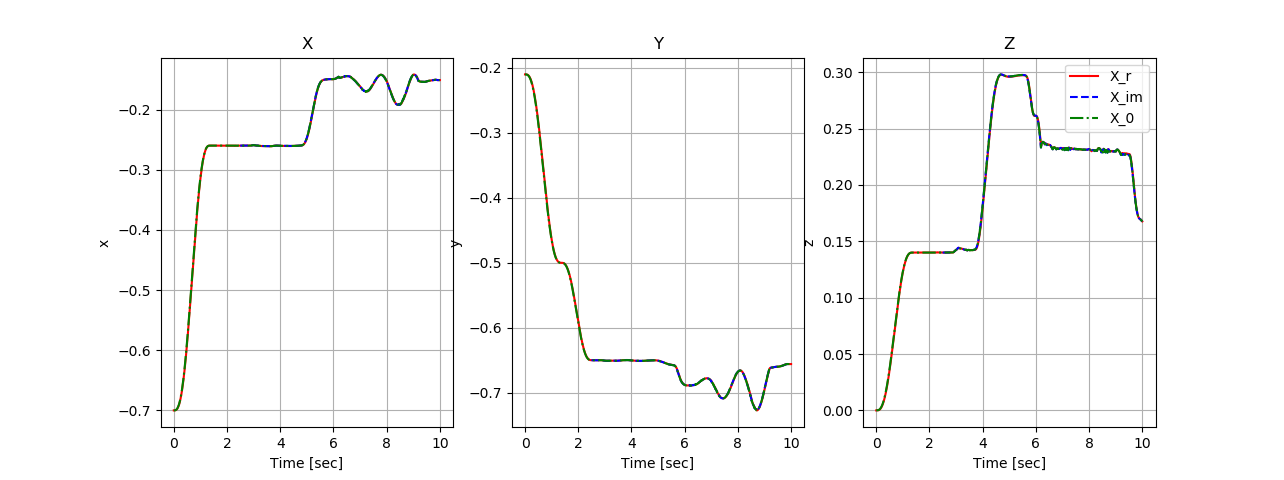
גרף 5 - כוחות בצירים השונים כתלות בזמן בהגדלה

נוכל לראות כי אכן יש קפיצה בכוח, כאשר החלק מתנגש בפעם הראשונה בקופסא ולאחר מכאן נגרר עליה בצורת חיפוש הספירלה, לכן יש קפיצות בכוח.

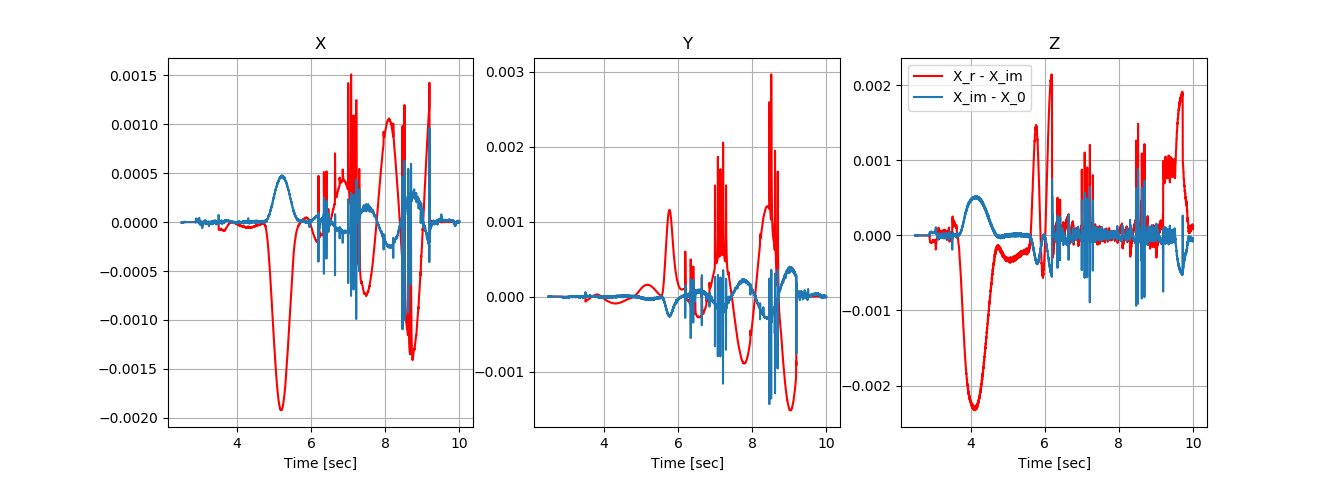
לבסוף החלק מגיע אל החור ונכנס ,ולכן לא פועל שם כוח בכיוון Z בזמן זה.

הכוח המקסימלי המתקבלת בציר Z במהלך התנועה הינו 14.5 N.

הגרף הבא מציג את התנועה המלאה של התפסנית במשך כל המשימה בקורדינאטות קרטזיות, לפי הצירים השונים:

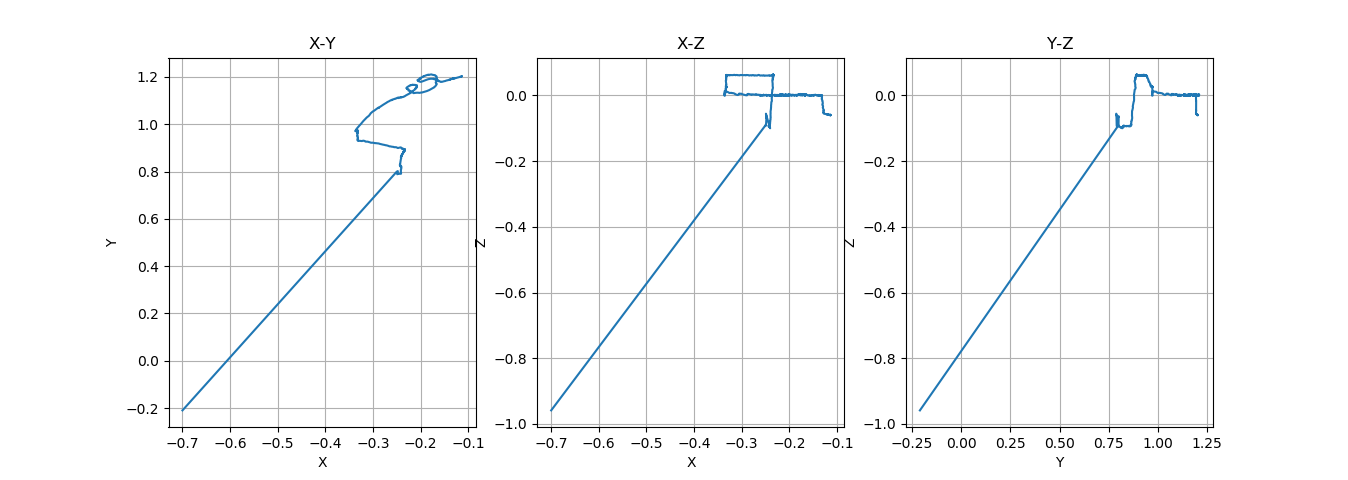
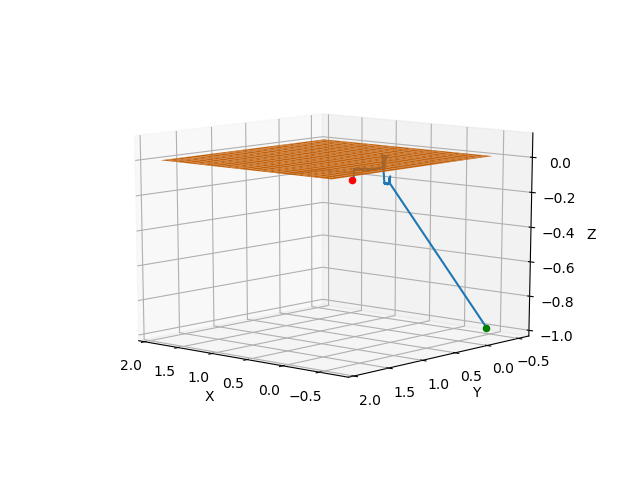


גרף 6 – תזוזת התפסנית כתלות בזמן



גרף 7 – שגיאת המיקום

הגרף הבא מציג את מסלול החלק שאותו אנו מנסים להכניס לחור במערכת צירים קרטזית ב3D ולפי הצירים השונים: (המשטח הכתום בגרף 3D מייצג את מישור החלק העליון של הקובייה , המישור בוא מתבצע חיפוש הספירלה.)



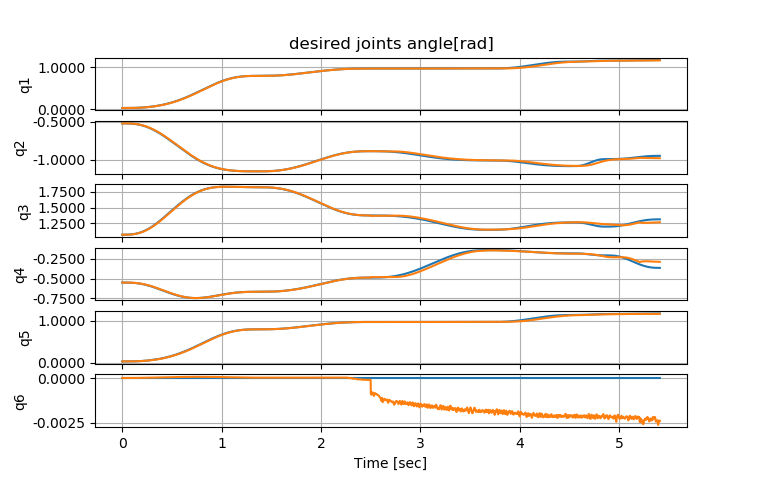
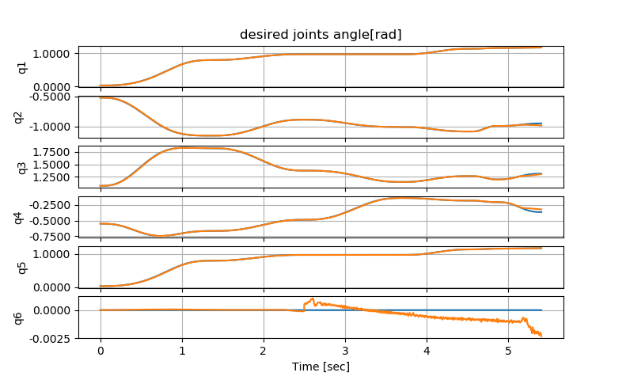
גרף 8 - מיקום הגליל במרחב

ניתן לראות כי באמת צורת התנועה המתקבלת של הגליל בציר XY דומה לספירלה עם שגיאה קטנה הנוצרת כנראה מחיכוך החלק במשטח ומזה שהבקרה הינה על תנועת התפסנית ולא על החלק עצמו.

בניסוי האחרון ביצענו השוואה בין התוצאות שקיבלנו בניסוי הקודם עם בקרת עכבה לעומת שימוש בבקרת PID

גרף 10 - בקרת אימפדנס

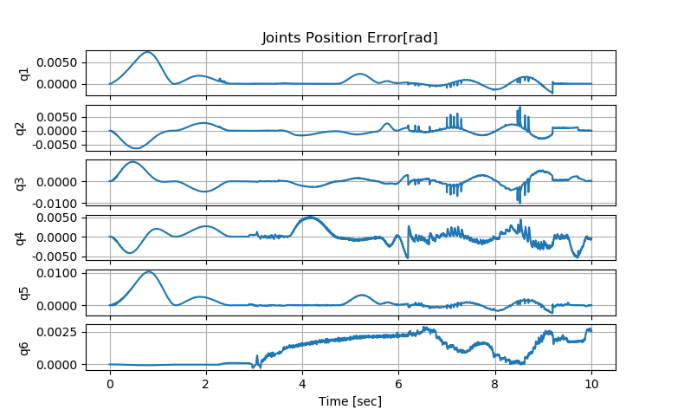
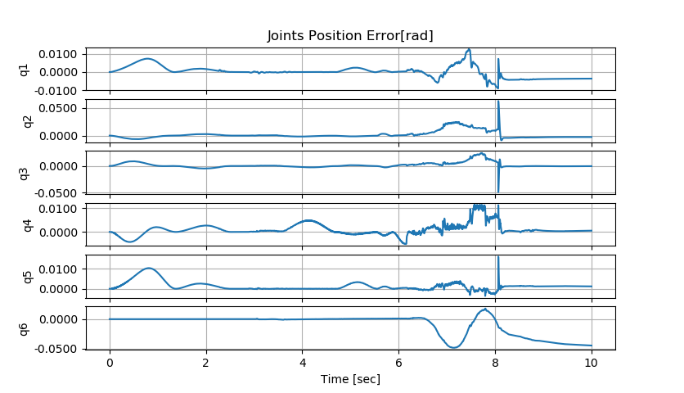
גרף 10 - בקרת PID



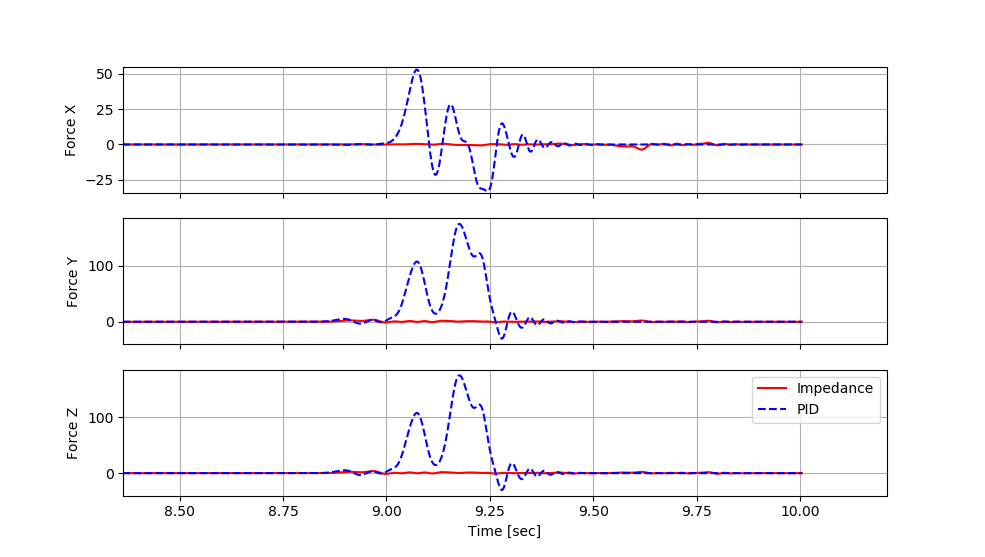
ניתן לראות כי בקרת PID עוקבת בצורה מדוייקת יותר אחרי תכנון התנועה לעומת בקרת העכבה אשר עושה "ויתורים" בדרך.

גרף 12 - בקרת PID

גרף 12 - בקרת אימפדנס



ניתן לראות כי השגיאה במפרקים באותה סדר גודל בין שני השיטות.



גרף 13 – הכוח המתפתח במהלך הפעולה בצירים השונים

לעומת זאת, ניתן לראות כי הכוח המופעל בבקרת PID הינו גדול בסדר גודל מהכוח המתפתח בבקרת אימפדנס, זהו דבר מאוד משמעותי כי ככל שהכוח היה קטן יותר כל התנועה תהיה פחות אגרסיבית ויותר מדוייקת.

ניתן לראות כי גם תנועת הגליל לא מבצעת את הספירלה הרצויה בצורה חלקה, זאת משום שישנו חיכוך גדול מאוד מה שמקשה על ביצוע התנועה, לעומת מה שראינו כאשר השתמשנו בבקרת עכבה (התנועה הייתה הרבה יותר חלקה).

## מסקנות

מהתוצאות ניתן לראות כי בקרת אימפדנס מתאימה יותר לביצוע המשימה של peg in the hole וגם לביצוע של החיפוש הספירלי, היא יותר רכה ופחות אגרסיבית לעומת בקרת PID. בשימוש בבקרת PID ניתן להגיע לתוצאה הרצויה אך עם הרבה שגיאות, מאחר שמתפתחים כוחות נורמליים גדולים וכתוצאה מכך כוחות חיכוך המפריעים לתנועה. לעומת זאת, בשימוש בבקרת אימפדנס אשר מתפשרת על המיקום ונותנת לנו את היכולת להוריד את כוחות אלו וליצור תנועה חלקה ומדוייקת יותר.

מצאנו כי הטווח האידיאלי להצלחת המשימה מבחינת השגיאות הליניאריות ופרמטרי הספירלה

אשר משפיעים בצורה משמעותית על הצלחת המשימה הינם:

שגיאה ליניארית בציר x בין -0.03 ל- 0.05מילימטר, שגיאה ליניארית בציר y בין -0.03 ל-0.05 מילימטר.

ערך הפרמטר a בין 0.0002 ל-0.002, ערל הפרמטר b בין 0.0012 ל-0.0018.

וכי זמני הכנסת החלק לחור נעים בין 6.1-10 שניות כאשר זמן החיפוש נע בין 0-3.9 שניות.

ראינו כי אם חורגים מטווחים אלו אחוזי השגיאה מתחילים לגדול . ניתן לשפר את החיפוש כך שיאפשר הצלחה בטווחים גדולים יותר , ניתן לנסות להוסיף אופציה של שינוי פרמטרי הספירלה לפי השגיאה המתקבלת.

# מאמרים

1. An Approach for Peg-in-Hole Assembling Based on Force Feedback Control

Peng Zou, Qiuguo Zhu, Jun Wu, Jianxiang Jin

1. Intelligent and Environment-Independent Peg-In-Hole Search Strategies

Kamal Sharma, Varsha Shirwalkar, Prabir K. Pal Division of Remote Handling & Robotics

Bhabha Atomic Research Centre Mumbai, India

1. Multi-Robot Assembly Strategies and Metrics

JEREMY A. MARVEL, ROGER BOSTELMAN, and JOE FALCO, U.S. National Institute

of Standards and Technology

1. Compliant Peg-in-Hole Assembly Using Partial

Spiral Force Trajectory with Tilted Peg Posture

Hyeonjun Park1;2, Jaeheung Park1;3, Dong-Hyuk Lee2, Jae-Han Park2, and Ji-Hun Bae2