**טיוטה לדו"ח פרויקט גמר**

**גל שכמונדס ונדב אמיר**

פרק 1: רקע

תת פרק 1.1: רקע תאורטי - בקרת עכבה

תת פרק 1.2: סימולציה מקדימה ב-Simulink

נספח 1: בחינת חלופות לכלי בקרה ותכנון תנועה של רובוטים

*פורמט לא סופי – לבדיקה ראשונית*

**1.1 -בקרת עכבה עם רובוט UR5**

רקע

עד שנת 1985 שיטת הבקרה העיקרית שהוטמעה על רובוטים תעשייתיים (למשל לצרכי צביעה או ריתוך) הייתה בקרת מיקום, ולעיתים נדירות יותר בוצע שימוש בבקרת כוח.

בשנים האחרונות חלה התפתחות רבה בעולם הרובוטיקה ועקב כך גם נרשמת עלייה מסיבית בהיקף השימוש בהם, בעיקר בתחומי התעשייה. על פי נוויל הוגן [1], האתגר המשמעותי הצפוי מהתפתחויות אלו הינו התמודדות עם רובוטיקה הכוללת מגע ושילוב רובוטים בכלל תחומי החיים, כתף אל כתף עם בני אדם.

בניגוד לבקרת מיקום קלאסית, בבקרת מגע נוצרת אינטראקציה דו כיוונית בין הרובוט (יחידת הקצה) לסביבה – קרי מתרחשת העברת אנרגיה. המסקנה העיקרית הנובעת מכך, הינה שקיים צימוד בין יחידת הקצה של הרובוט לבין הסביבה עימה הוא בא במגע. צימוד זה יוצר אילוצים פיזיקליים שניתן לתעל לטובת ביצוע המשימה הנדרשת.

כידוע, לא ניתן לבצע בקרה בו זמנית בלתי תלויה גם של מיקום וגם של כוח היות וקיים בניהם קשר ניוטוני. כמו כן, בקרה מדויקת של כוח **או** מיקום היא מאתגרת מכיוון שהיא דורשת מודל מדויק של האובייקט עימו הרובוט בא במגע (ובמודל מסוג זה לרוב קיימת אי ודאות).

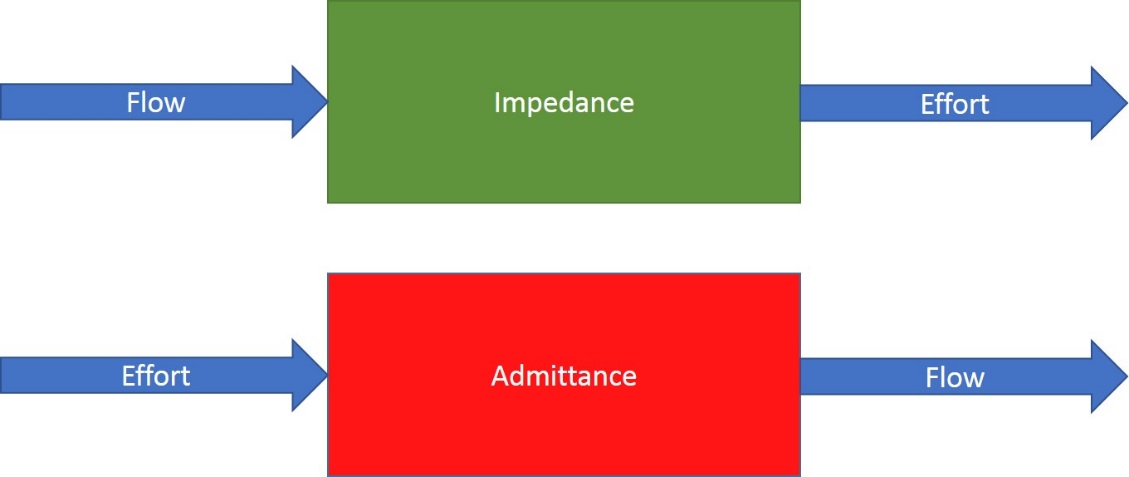
הפתרון שמציע נוויל הוגן הוא שימוש ב Port Behavior”" - ניסיון לשלוט בהתנהגות המערכת בנקודת המגע (Port) בין הרובוט לסביבה – בקרת מגע (או בקרת עכבה). כאשר עוסקים בבקרת מגע, חשוב להבין את העיקרון המנחה – עיקרון אימפדנס (Impedance) ואדמיטנס (Admittance).

ניתן לתאר אינטראקציה פיזיקלית ע"י שני סוגי משתנים מצומדים:

* + **Effort** - למשל: כוח או מתח חשמלי.
  + **Flow** - למשל: מהירות או זרם חשמלי.

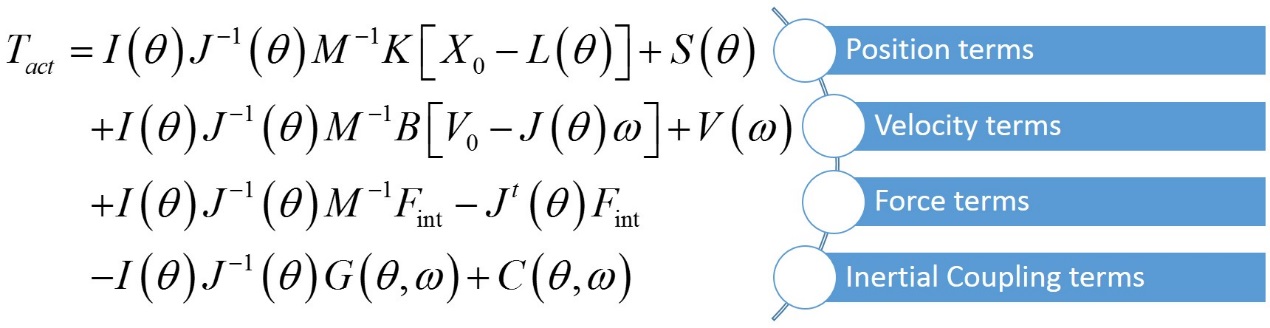
אין מקור בודד שיכול לייצר גם Effort וגם Flow בו זמנית.

ניתן לחלק את המערכות הפיזיקליות לשני סוגים: אימפדנס ואדמיטנס.



מטבע הדברים אובייקטים בסביבה מתנהגים כאדמיטנס (מקבלים כוח ומוציאים תזוזה) - לכן השאיפה היא **לתכנן בקרה שגורמת לרובוט ( Manipulator) להתנהג כאימפדנס.** שינוי האימפדנס = שינוי אופי האינטראקציה בין הרובוט לסביבה.

איור 1 – תיאור גרפי פיזיקלי של מערכות המתנהגות כ Impedance ומערכות המתנהגות כ Admittance

לפי הוגן, ביטוי אפשרי לתיאור התנהגות יחידת הקצה באופן הרצוי לנו, המוכתבת ע"י הבקר הוא:

איור 2 – חוק הבקרה המוצע על פי נוויל הוגן

*כאשר:*

*– טנזור האינרציה בקורדינטות האקטואטור, - יעקביאן, - טנזור האינרציה בקורדינטות יחידת הקצה. - קשר כוח/מיקום - קבוע קפיץ, - מיקום וירטואלי של יחידת הקצה, – משוואות קשר קינמטי. - כוח אינטרקציה, – מומנטים תלויי מיקום, – איברי צימוד תאוצתים, B – קשר כוח/מהירות - קבוע ריסון. - איברי צימוד אינרציאליים (קוריוליס וצנטרפוגלי). - מהירות וירטואלית של יחידת הקצה, – מהירות האקטואטור, – מומנטים תלויי מהירות.*

לאחר ציון חוק הבקרה שהוצע על ידי נוויל הוגן, ניתן להתייחס ליתרונותיו ולחסרונותיו:

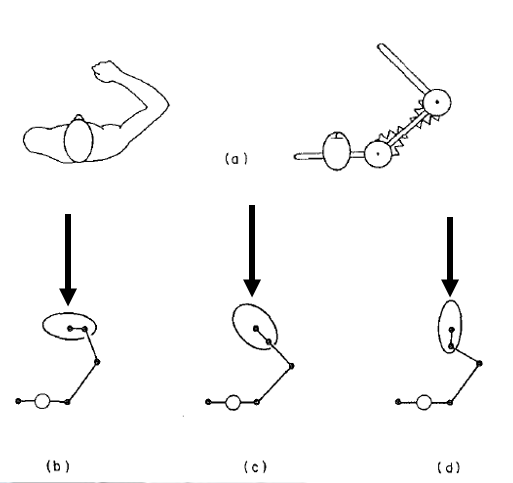
**יתרונות**:

* על ידי קביעת קבועי הקפיץ והריסון בחוק הבקרה (אימפדנס יחידת הקצה), ניתן לשנות את התנהגות יחידת הקצה כרצוננו ולהתאימה למשימה.
* אין צורך להשתמש בקינמטיקה הפוכה, אלא רק בקינמטיקה ישירה.

**חסרונות**:

* חוק הבקרה הנ"ל טומן בחובו הנחה שמטריצת היעקביאן הינה הפיכה, מה שלא נכון בחלק מהמקרים.

עוד מציע נוויל הוגן יישום של בקרת עכבה ללא סגירת חוג, אלא על ידי יצירת יתירות של אקטואטורים או דרגות חופש. זאת, מפני שסגירת חוג מכניסה זמנים מתים (Time Delays) למערכת ויעילות בקרה בחוג סגור בנוכחות זמן מת גדול מוגבלת. לנושא זה עשויה להיות חשיבות רבה במטלות המערבות אינטראקציה דינמית. כך, ניתנת דוגמה מעולם החי אודות זמן המעבר של אות הנקלט על עור בזרוע אדם לאזור גבוה יותר במערכת העצבים - 70 ms. מעבר אות בחוג הביולוגי - מכניס Time Delay של 100-150 ms. חישוב ממוחשב והצגתו באופן וויזואלי - עוד 200-250 ms.



איור 3 – הפעלת כוח על יחידת קצה הנמצאת בקונפיגורציות שונות

בדוגמה המתוארת מעלה, **בתמונה b**הכוח גורם לתזוזה בעיקר בנקודה השמאלית ביותר. **בתמונה d** הכוח גורם לתזוזה של כל שלוש החוליות. כך, הלכה למעשה שונה אימפדנס יחידה הקצה, ללא קשר לחוג הבקרה.

בפרויקט זה, ביצענו שימוש בבקרת עכבה על רובוט מדגם UR5 לצורך ביצוע מטלה של חיווט חוט תקשורת בתפסניות ייעודיות ובתנאי אי וודאות. לאחר מכן, ערכנו השוואה בין התוצאות המתקבלות בבקרת עכבה לתוצאות המתקבלות על ידי בקרת מיקום קלאסית. כל זאת, כחלק ממאגד "רובוטיקה בתעשייה", במטרה לפתח טכנולוגיות ייצור רובוטיות גנריות, ליישום בתהליכי הרכבה ובמשימות לוגיסטיקה פנים מפעלית בתעשייה, המאפשרות בניית תהליכי ייצור רובוטיים הפועלים בגמישות ובתנאי אי-ודאות בתהליך הייצור של רכיבים "רכים" בסביבת בני אדם.

**1.2 – סימולציה מקדימה ב Simulink**

כחלק מתהליך הלימוד של נושא בקרת העכבה ולצורך הטמעת הידע בצורה מעשית, ביצענו מידול של מערכת ליניארית חד ממדית פשוטה, המורכבת ממסה, קפיץ ומרסן בSimulink. על מערכת זו מימשנו חוק בקרת עכבה Dynamic Based Impedance Control כפי שתואר במאמר של מרים זקסנהויז ותומר ולנסי [2].

את המערכת הנ"ל ניתן לתאר על ידי המשוואה:



  כאשר:

 ,  - קבוע הריסון,  - קבוע הקפיץ,  - המסה.

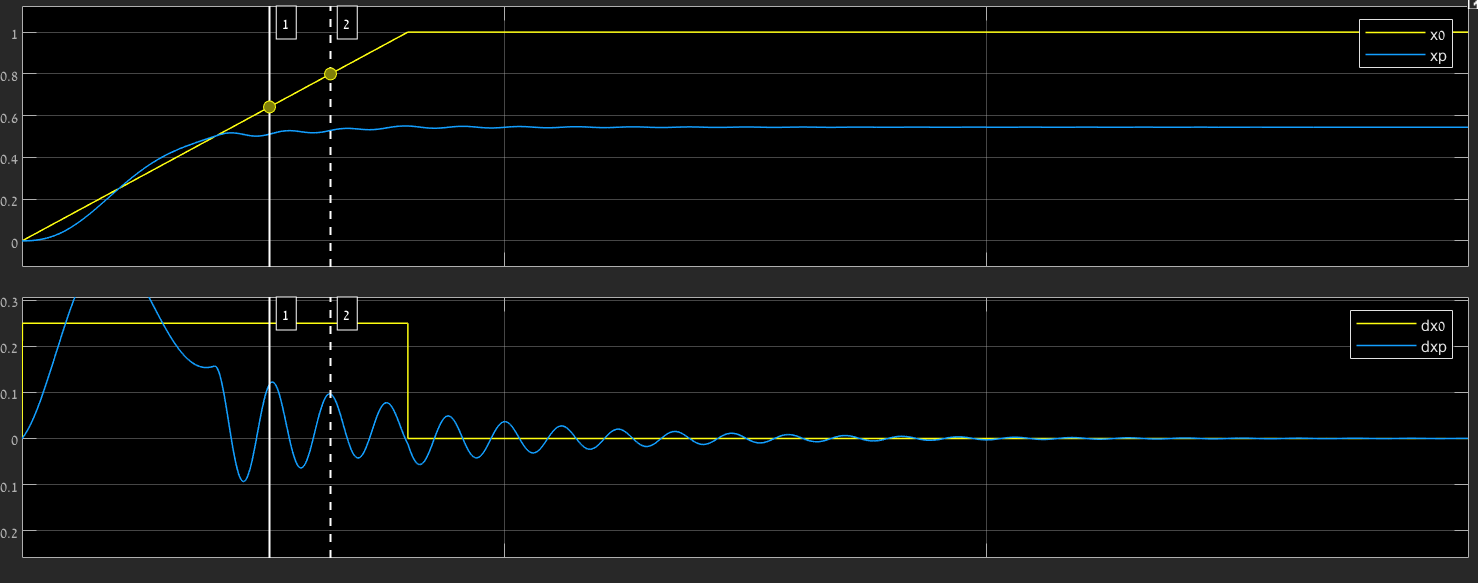


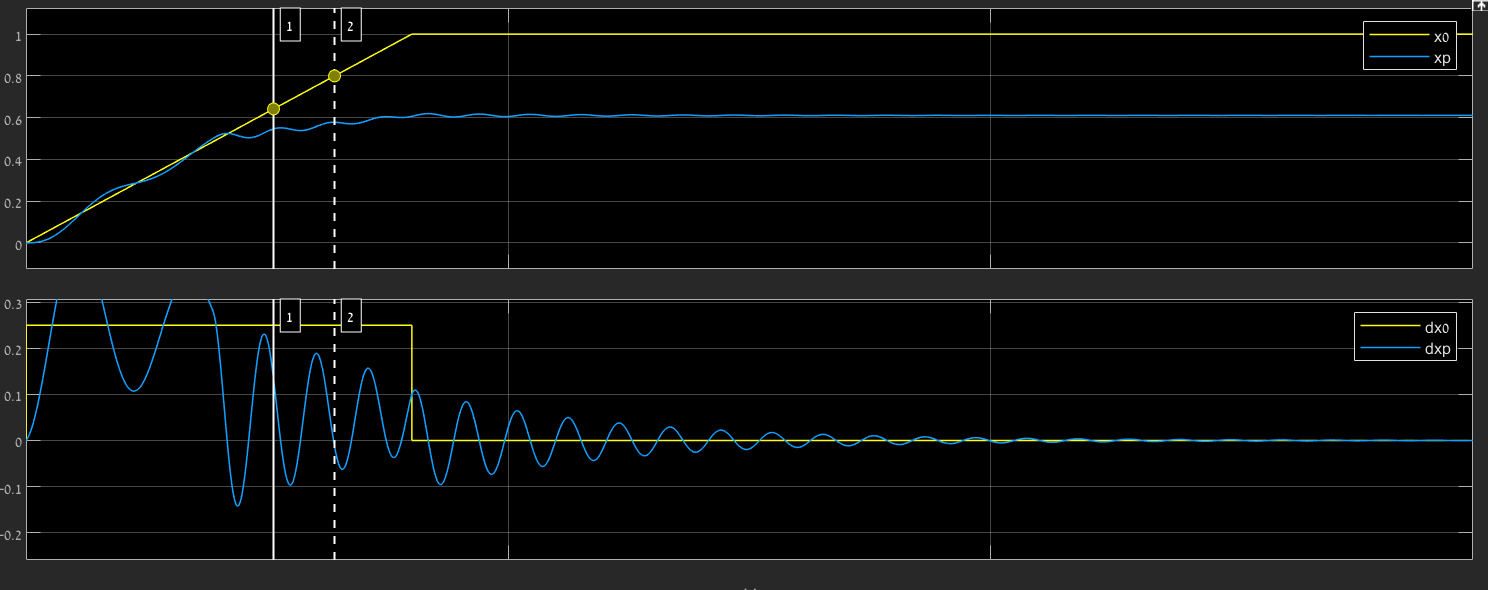
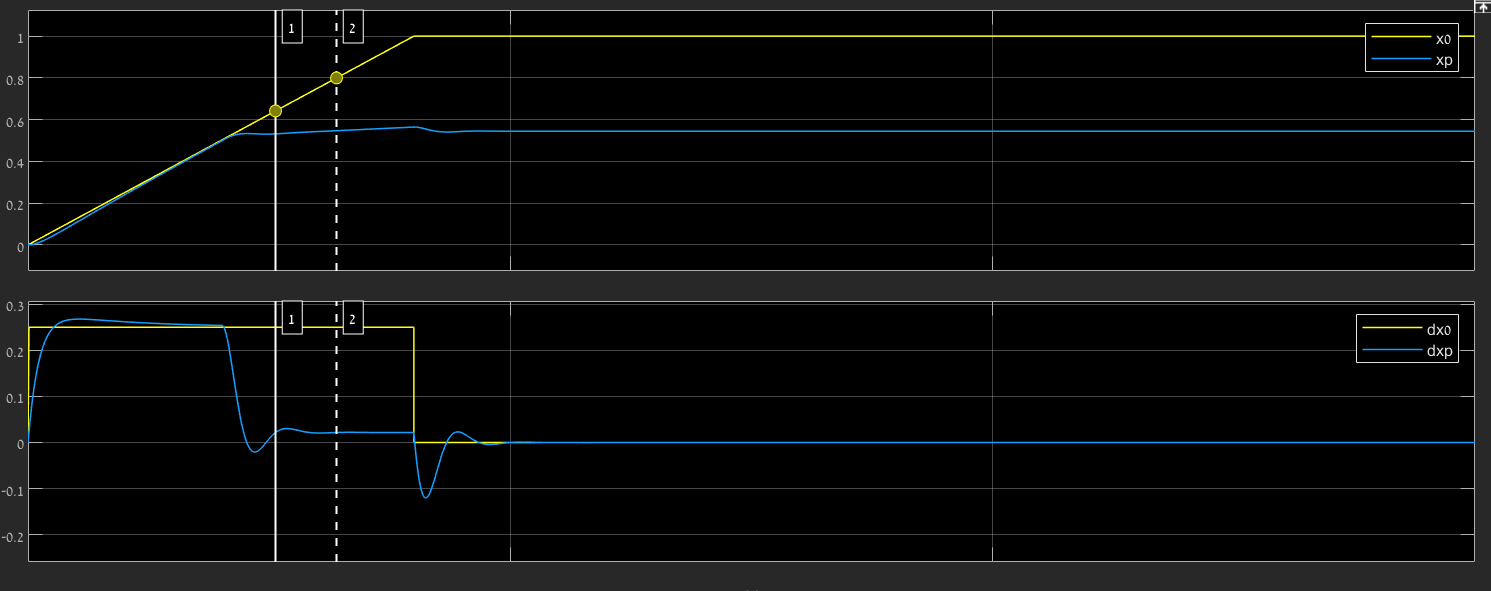
כאשר:

 - הפרמטרים המשוערים של המודל.

 - הפרמטרים של האימפדנס הרצוי.

בסימולציה הודגם חוג בקרה על מסה נקודתית במשקל 1 ק"ג, המחוברת לקפיץ בעל קבוע  ולמרסן בעל קבוע . במרחק חצי מטר מהמסה הוצב קיר שמודל ככוח מחזיר אקספוננציאלי. נראה שעל ידי שינוי ערכי קבועי הקפיץ והמרסן באופן "רך" (כלומר ללא שינוי פיזי של הרכיבים עצמם במערכת) אנו גורמים למערכת להתנהג כרצוננו.

****תוצאות: (בדו"ח הסופי נשים פה גרפים שיישמרו ישירות ממטלאב ויהיו יותר אינפורמטיביים)



**בצהוב**: אות הרפרנס, **בכחול**: התנועה בפועל.

**עבור הגרף העליון:** נתוני האימפדנס כמו הנתונים ההתחלתיים של המערכת.

**עבור הגרף האמצעי:** קבוע הקפיץ גדול פי 3.

**עבור הגרף התחתון:** קבוע הריסון גדול פי 10.

**דיון בתוצאות:**

כפי שניתן לראות, על ידי שינוי פרמטרי האימפדנס באופן רך, ניתן לגרום ליחידת הקצה להתנהג כרצוננו. הערה לישראל: נשמח לשוחח עם עמך איך להרחיב את הפרק הזה. האם לבצע עוד סימולציות. כיצד יש לדון בתוצאות.

**נספח 1 - חלופות לכלי בקרה ותכנון תנועה של רובוטים**

**טבלה מסכמת:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **האם דרוש מחקר נוסף** | **דירוג** | **הערות** | **GitHub** | **מחיר** | **שפות תכנות** | **ייבוא UR5** | **ייעוד** |  |
| - | 4.5 | הטוב והמוכר | V | 9000$ - חשבון מחקרי.  רשיון רגיל לסטודנט - חינם | PYTHON,C++,MATLAB | V | סימולציות של רובוטים וביומכניקה | **Mujoco** |
| - | 4 | רק לינוקס | V | קוד פתוח | PYTHON,C++ | V | סימולטור רובוטיקה | **MoveIt** |
| - | 4 | נראה מעולה | V - קהילה גדולה | קוד פתוח | C,C++,PYTHON, MATLAB, JAVA, ROS | V | סימולטור רובוטיקה מרחבי | **Webots** |
| חסר מידע - דרוש מחקר רב נוסף | 3 |  | V | קוד פתוח | ? | VX | סימולטור רובוטיקה | **Gazebo** |
| ייבוא UR5 | 4 |  | V | רשיון חינם לסטודנטים | C,C++,PYTHON,JAVA,LUA, MATLAB, OCTAVE | ? | רובוטיקה | **Coppelia robotics** |
| ייבוא UR5 | 3.5 | נראה טוב | V- קהילה גדולה | קוד פתוח | Python | ? | סימולציות רובוטיקה | **Bullet** |
| תכנות בקר אימפדנס | 2-4 |  | V | 195 דולר שנתי לסטודנים | כלי ייעודי, PYTHON, C, C#, MATLAB, VISUAL BASIC | V | סימולציות לרובוטים תעשייתיים | **RoboDk** |
| דרוש מחקר רב – לבדוק אם וכיצד אפשר להכניס לשם רובוטים | 3 |  | V - קהילה קטנה | קוד פתוח | ? | ? | מנוע פיזיקלי רב עוצמה | **PHYSX** |
| דרוש מחקר נוסף. איך ניתן לבצע בקרה של רובוט | 2 |  | V | קוד פתוח | C#,C++,JAVA | - | סביבה לתכנון משחקי מחשב | **Unity** |

**פירוט החלופות:**

**Mujoco:**

**כתובת**: <http://www.mujoco.org/book/index.html>

מנוע פיסיקלי אשר מיועד לביצוע סימולציות ברובוטיקה, ביומכניקה ועוד. בעל יכולות מתקדמות במידול מגע בין גופים. ניתן לממש בסביבת העבודה בקרים לשליטה ברובוטים שתוכננו על ידי המשתמש ולבחון אותם לפני בדיקה בפועל על הרובוט האמיתי. ניתן לבצע אינטגרציה של סימולציה באמצעות מנגנון הRendering של-unity. ניתן לתכנות ב C++, Python ו- Matlab.

**MOVEIT:**

**כתובת:** <https://moveit.ros.org/>

סביבת סימולציה לרובוטים, כולל סימולציית מגע, התנגשויות, קינמטיקה (ישירה והפוכה), בקרה, תכנון מסלול. עבודה רק בלינוקס, **מחייב התקנה של ROS לפני ההתקנה.** קיים מודל של רובוט UR5. נראה שניתן להעביר קוד לרובוט דרך סביבה זו.

<https://github.com/ros-planning/moveit>

**Webots**

**כתובת:** <https://cyberbotics.com/#webots>

סביבת עבודה מיועדת לסימלציות מרחביות של רובוטים: Free Open Source, מאפשר ייבוא קבצי CAD. קיימות ספריות רבות מהן ניתן לייבא רובוטים, חיישנים, אקטואטורים ואובייקטים – כולל UR5. ניתן לעבוד על ווינדוס/לינוקס. אפשר לעבוד עם ROS. שפות תכנות מחוץ ל WEBOTS - C, C++, Python, ROS, JAVA, Matlab.

**GAZEBO**

**כתובת:** <http://gazebosim.org/>

סביבת סימולציה קוד פתוח. מיועדת לרובוטים. דומה לwebots-. ניתן לעבוד בענן.

<https://github.com/ros-simulation/gazebo_ros_pkgs>

**Coppelia robotics** (לשעבר (Vrep

כתובת: <http://www.coppeliarobotics.com/>

סביבה למידול סימולציות של רובוטים מפיתוח של חברת Coppelia Robotics. בקרי הרובוטים ניתנים לתכנות בשפות בשפות: C/C++, Phyton, Java, Lua, Matlab או Octave. הסביבה מכילה תכונות רבות כגון: מנגנון לסימולציות פיזיקליות, קינמטיקה של רובוטים (ישירה והפוכה), תכנון תנועה, זיהוי התנגשויות ועוד. כמו כן, ניתן להוסיף פיצ'רים מתכנון אישי. **רישיון סטודנטיאלי ניתן בחינם**. נראית תוכנה מקצועית. מבדיקה שטחית נראה שיש קהילה קטנה מאוד באינטרנט והרבה לקוחות מהשורה הראשונה ( Fanuc, Kuka, Google, Amazon ועוד).

**Bullet**

**כתובת:** <https://pybullet.org/wordpress/>

סביבה קוד פתוח המאפשרת מידול סימולציות פיזיקליות בעלת מיקוד בעולם הרובוטיקה. נראה שהיישומים הניתנים לביצוע ידי סביבה זו מקבילים ליישומים הניתנים לביצוע על ידי Mujoco. ניתן למדל לתוך סביבת העבודה מודלים של אובייקטים ורובוטים מבחוץ בדומה ל Mujoco וליישם חוקי בקרה שנכתבו במיוחד.

**RoboDk**

כתובת: <https://robodk.com/>

סביבת עבודה לתכנות רובוטים תעשייתיים. נראה שמיועדת לתכנון מסלולי תנועה לרובוטים הנושאים כלי עבודה. בתוכנה קיים מאגר של מאות רובוטים שניתן לייבא אליה כולל UR5. ניתן לייבא מהתוכנה קוד ישר לרובוט. עם זאת, נראה שלא ניתן לתכנן חוג בקרה מתוכנן אישית אלא רק בעזרת כלי ייעודי בתוכנה – מומלץ לבצע בדיקת המשך. קיימת יכולת מסוימת לעבודה עם פיתון. רישיון סטודנטיאלי לשנה עולה 195 דולר.

**PHYSX**

**כתובת:** <https://www.nvidia.com/en-us/drivers/physx/9_18_0907/physx-9-18-0907-driver/>

מנוע פיזיקלי מבית NVIDIA, תוכנן במקור עבור תמיכה במשחקי מחשב מרובי דינמיקה. סביבה בקוד פתוח נגישה למפתחי משחקי מחשב. סביבה תומכת בסימולציות של מגע בין גופים קשיחים ובין גופים רכים ברמה גבוהה מאוד. לא ברור כיצד לממש בקרה רובוטית.

**Unity**

**כתובת:** <https://unity.com/>

סביבה המיועדת למידול תלת מימדי של משחקי מחשב עם יכולות Rendering טובות מאוד ועם מנוע פיזיקלי. נראה שניתן לממש סימולציות בקרה רובוטיות אבל באיכות נמוכה. מצד שני, נראה שניתן להכניס לתוכו Plugin של Mujoco. קיימת סביבה בשם Unity simulations לביצוע סימולציות פיזיקליות כרגע בגרסת בטא. מחיר: בחינם למשתמשים פרטיים.

**סביבות נוספות שנסקרו ונמצאו לא מתאימות**:

**Visual components**

**כתובת:** <https://www.visualcomponents.com/>

סביבה המאפשרת תכנון וייעול של קווי ייצור במפעלים.