* בניית ממשק בין האלגוריתם לסימולטור:

כדי שהאלגוריתם של האופטימיזציה יוכל לבצע את ייעודו צריך לבצע סימולציה, לכן צריך ליצור ממשק שיחבר בין האלגוריתם לסימולטור. ניתן לראות באיור 1 איך הממשק יעבוד: הממשק יקבל את מבנה הזרוע מהאלגוריתם, ישלח לסימולטור את מבנה הזרוע בפורמט המתאים ואת נקודות היעד שהזרוע צריכה להגיע אליה. הסימולטור יחזיר לממשק אם הזרוע הצליחה ואת זמן ההגעה לכל נקודה. הממשק יחזיר לאלגוריתם את זמן המחזור של פעולה.

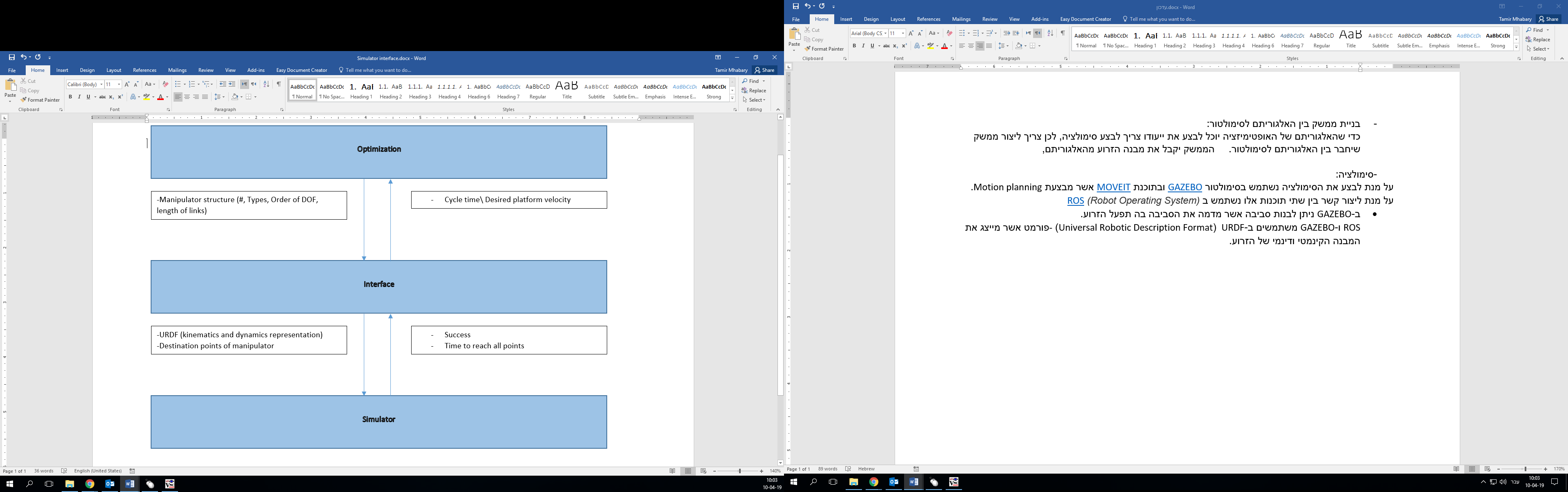


Figure 1- Simulator Interface

-סימולציה:

על מנת לבצע את הסימולציה נשתמש בסימולטור [GAZEBO](http://gazebosim.org/) ובתוכנת [MOVEIT](https://moveit.ros.org/) אשר מבצעת Motion planning. על מנת ליצור קשר בין שתי תוכנות אלו נשתמש ב [ROS](http://www.ros.org/about-ros/) (Robot Operating System)

* ב-GAZEBO ניתן לבנות סביבה אשר מדמה את הסביבה בה תפעל הזרוע.

ROS ו-GAZEBO משתמשים ב- URDF (Universal Robotic Description Format) -פורמט אשר מייצג את המבנה הקינמטי ודינמי של הזרוע.

* הנחות ליצירת זרוע

1. כל החוליות הם צילינדרים בקוטר הבא: בסיס, חוליות 1-3 ברדיוס 0.06 מטר, חוליה 4 ברדיוס 0.04 מטר, חוליה 5 ברדיוס 0.03 מטר וחוליה 6 ברדיוס 0.025 מטר
2. נבדוק בין 3-6 דרגות חופש
3. כל המפרקים יהיו מסוג: מסתובב או מתארך PARISMATIC / REVOLUTE
4. גבולות של זרוע: עבור מסתובב [-PI,PI] רדיאנים , עבור מתארכת עד פי 2 מאורך החוליה
5. Gazebo - - המערכת צירים נקבעת לפי החלק הקודם (PARENT) 🡨 אין סיבוב סביב ציר 'X'
6. אורכי הצירים יחולקו ל-4 מקטעים – (0.1 ,0.4 ,0.7 ,1 meter)
7. נבדוק בכל זרוע כאשר כל מערכת צירים תהיה בפוזיציה שונה (0 / -90 מעלות)
8. חוליה ראשונה בכל זרוע תהיה קבועה (0.1 מטר
9. שני מפרקים מתארכים אחד אחרי השני לא יהיו מקבילים (חייב סיבוב של -90 מעלות(
10. הציר של המפרק הראשון יהיה תמיד ניצב לבסיס ומסתובב (ציר (Z
11. ציר מתארך יהיה תמיד בציר Z assume 8
12. זרוע שכל המפרקים הינם מתארכים תיבדק רק עד 3 דרגות חופש assume 3
13. כאשר יש שני מפרקים מסתובבים – אז השני חייב להיות מאונך לראשון assume 6
14. שני צירים מסתובבים אחד אחרי השני לא יהיו שניהם Z (ללא סיבוב ב-90 מעלות) ?

* לפי הנחות 1-8 יש 1,108,377,600 אופציות שונות של זרועות. לאחר כל ההנחות יש 28,279,808אופציות שונות של זרועות. כאשר הנחה 9 מורידה 169,678,848 אופציות, הנחה 10 מורידה 226,238,464אופציות עבור ציר ראשון מסתובב ו554,188,800 אופציות עבור ציר ניצב, הנחה 11 מורידה 101,711,872 אופציות.
* את כל קבצי הקונפיגורציות השונות ( \*\*\* קונפיגורציות) של מבנה הזרוע ייצרתי מראש על מנת לחסוך זמן הרצה בזמן הרצת האלגוריתם. - לוקח כ
* כרגע הנקודה אליה הזרוע תזוז היא ע"י הזזת הזרוע במסך.
* מה יש לעשות עוד בסימולציה:
* קביעת נקודת היעד של הזרוע מהקוד ולא ידנית
* מידול של הצמח
* הוצאה מהסימולטור לממשק את זמן ההגעה של הזרוע לכל נקודה
* קביעת פרמטרים אחרונים למפרטים ביצירת URDF
* לשלב שני - כמה זמן יכולה להישאר בפוזיציה
* Path planning - האם משנה איזה אלגוריתם לבחור כדי לתכנן את מסלול הזרוע?