Τεχνολογίες Ευφυών Συστημάτων και Ρομποτική 2020-21
2η Ατομική Προαιρετική Εργασία

Μπουλαφέντης Χρήστος ΑΜ:1059612 Δ' Έτος

PID Controller:

Για την δημιουργία του PID Feedback Controller βασίστηκα στον PI Controller που μας δώθηκε. Η διαφόρα μεταξύ των δύο αλγορίθμων είναι πως στον PID Controller προστίθεται στην αναδρομικό εξίσωση που υπολογίζει την ταχύτητα με την οποία θα κινήθει ο end effector ένας επιπλέον όρος. Ο όρος αυτός υπολογίζει τον ρυθμό αλλαγής του Error και επομένως υπολογίζεται ως η παράγωγος του σφάλματος ως προς τον χρόνο dt. Επίσης πολλαπλασιάζομε τον όρο αυτόν με μία σταθερά Kd, επομένως έχουμε τον όρο:

$$Kd*(det)/(dt)$$

Η υπόλοιπη υλοποίηση είναι κοινή εκτός μικρών διαφορών λόγω της πρόσθεσης του προαναφέροντα όρου.
Τον όρο αυτόν στην Python τον υπολόγισα με τον εξής τρόπο:

```
paragwgos = (self._previous_error - error_in_world_frame) / self._dt

self._previous_error=error_in_world_frame

return self._Kp * error_in_world_frame + self._Ki * self._sum_error +
self._Kd * paragwgos
```

Όπου ως _previous_error έχω ορίσει το σφάλμα την προηγούμενη χρονική στιγμή dt.

Το αποτέλεσμα του PID Feedback Controller για τελική θέση του end effector:
[0, 0, 0, -1.5708, 0, 1.57079633, 0, 0.3, 0]

```
[ 1.37328359  0.50184047 -1.75525198 -1.54513368  0.49886741  1.46245734
-0.35019669 -0.2882892   0.05864182]
```

PI Controller:

Τον PI Controller τον τροποποίησα ώστε να δίνει και αυτό τυχαίες αρχικές θέσεις στο ρομπότ Franka και να έχει ώς στόχο τις ίδιες τελικές του end effector.

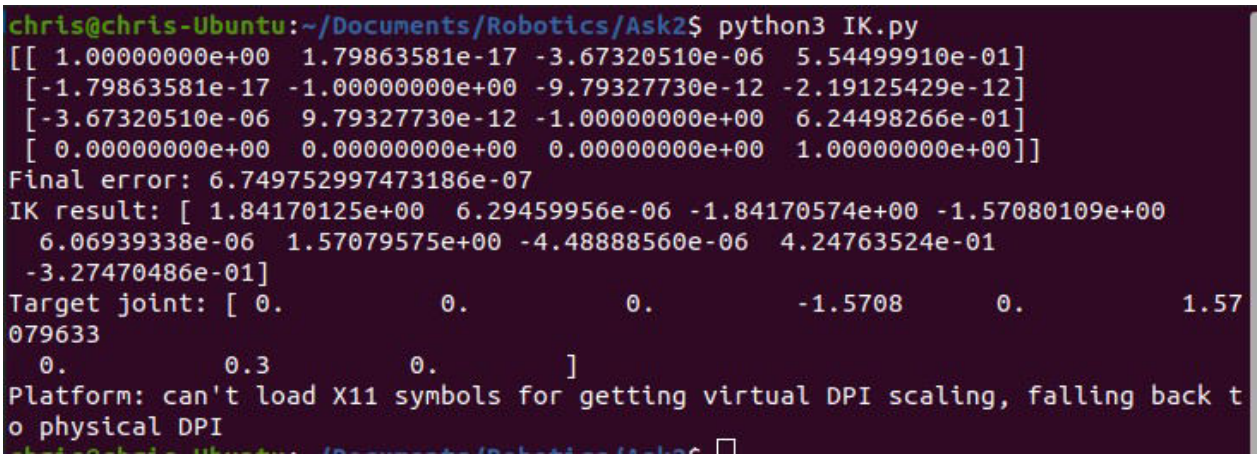
Το αποτέλεσμα του PI Controller για τελική θέση του end effector:
[0, 0, 0, -1.5708, 0, 1.57079633, 0, 0.3, 0]

```
[ 1.53091752e-03  8.20582422e-02  6.30480335e-04 -1.56889852e+00
 -1.58757712e-03  4.09762430e-01 -2.86239521e-01  7.93303298e-01
  7.03342944e-01]
```

Inverse Kinematics Controller:

Τέλος, στον Inverse Kinematics Controller, ομοίως με τον PI Controller, βασισμένος στον δωθέντα κώδικα έκανα αλλαγές ώστε να παίρνει και αυτός τυχαίες αρχικές θέσεις και να έχει την ίδια τελική.

Το αποτέλεσμα του IK Controller για τελική θέση του end effector:
[0, 0, 0, -1.5708, 0, 1.57079633, 0, 0.3, 0]



Σύγκριση Αλγορίθμων:

Βέλτιστος αλγόριθμος θεωρώ πως είναι ο PI Controller. Ο λόγος είναι πως ο Inverse Kinematics Controller δεν λαμβάνει υπόψην του τα αριθμητικά σφάλματα που δημιουργούνται μέσω ανακριβιών των αισθηρήρων, εξωτερικών παραγόντων και άλλων σε κάθε βήμα και επομένως το συσσωρευμένο τελικό σφάλμα μπορεί να είναι μεγάλο. Πρέπει να σημειωθεί πως και οι δύο αλγόριθμοι είναι πιθανό να “παγιδευτούν” σε μια λανθασμένη κατάσταση στην οποία να μην τους επιτρέπει να διαφύγουν τα φυσικά όρια του ρομπότ παρά την ύπαρξη λάθους.