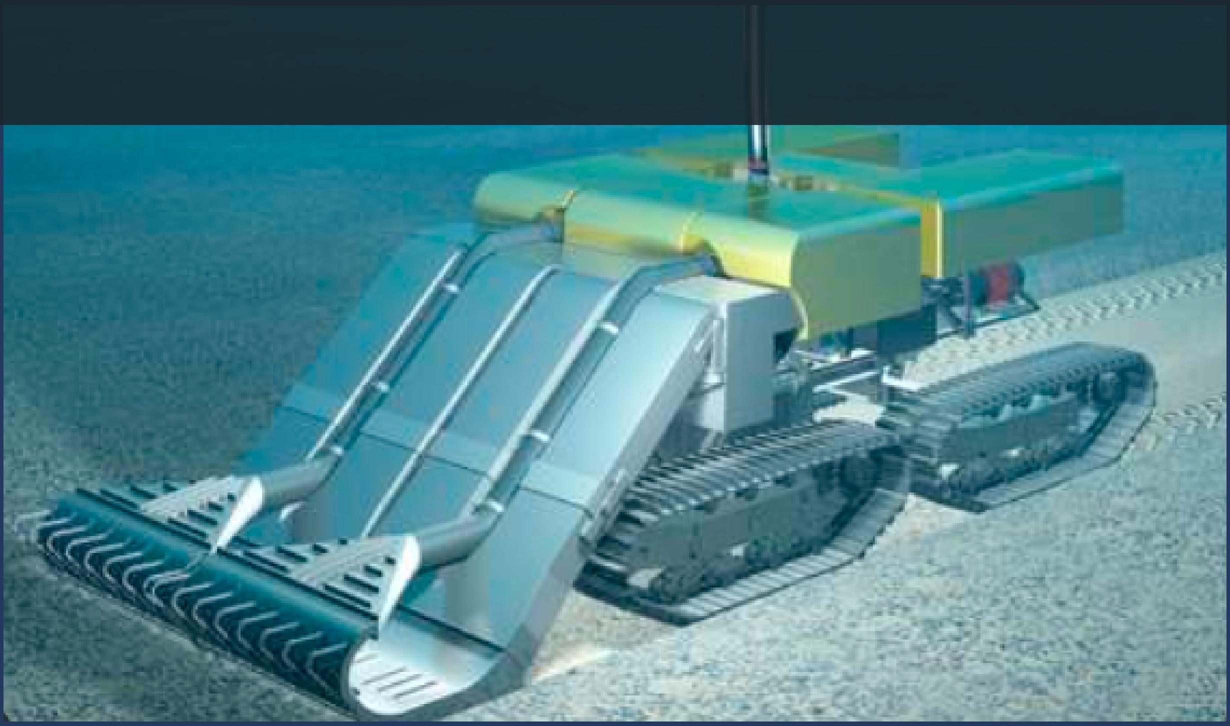
Impedance control strategies in soft sediment applications.

Using impedance control as a means to measure of penetration depth accuracy and force stability in deep sea mining applications.

Joseph Touzard



TU Delft

Thesis for the degree of MSc in Marine Technology in the specialization:
Marine Engineering and Maritime Operations and Management

Impedance control strategies in soft sediment applications.

By

Joseph Touzard

Performed at

TU Delft

This thesis (MT.24/25.033.M) is classified as confidential in accordance with the general conditions for projects performed by the TU Delft.

Date of Exam: 08/04/2025

Thesis exam committee
Chair/Responsible Professor: Andrea Coraddu
Daily Supervisor(s): Micah Prendergast, Rudy Helmons

Author Details
Studynumber:5693403

Contents

Abstract

This research is conducted in the framework of a Master's thesis within the Marine Engineering department of the Maritime Transport and Technology branch and in joint collaboration with the Robotics branch of the Mechanical Engineering Faculty of Tu Delft.

The precise control of robotic systems in granular underwater environments is essential for applications such as deep-sea mining, sediment sampling, and seabed infrastructure maintenance. In such environments, the interaction between the robot and deformable substrates like sand and clay plays a crucial role in operational efficiency and system stability. This research investigates how fine-tuning joint stiffness in an impedance controller influences penetration depth accuracy, horizontal force distribution, and force consistency along a trajectory mapped using 3D camera point cloud data. Understanding these relationships is critical for optimizing force control strategies in unstructured and dynamic underwater settings.

Experiments were conducted using a KUKA iiwa 7 robotic arm equipped with an impedance controller, following a mapped trajectory over a real sandbed in both dry and submerged conditions. The point cloud data from a 3D camera provided accurate environmental mapping, ensuring precise trajectory tracking. The results indicate a significant correlation between joint stiffness and penetration accuracy: higher stiffness improved depth accuracy and reduced external disturbances but compromised adaptability in cases where the robot encountered hard obstacles. Conversely, lower stiffness increased compliance, allowing for smoother interactions but at the cost of greater sensitivity to force fluctuations.

Fluid damping in submerged conditions was found to reduce penetration error variability, highlighting the stabilizing influence of water on force interactions. The study also revealed that current robotic systems for deep-sea applications differ significantly from the 7-degree-of-freedom (DOF) KUKA arm used in this research. In practical scenarios, deep-sea mining robots typically feature a single actuated DOF (pitch), with other degrees of freedom facilitated by passive flexibility rather than active control. These structural differences influence force distribution and overall system behavior, emphasizing the need for future studies tailored to real-world deep-sea mining configurations.

Future research should extend beyond sand to softer seabed sediments such as clay, which behaves more like a Bingham fluid and exhibits significantly lower shear strength—potentially by a factor of 5 to 10 compared to sand. Additionally, exploring adaptive stiffness strategies that dynamically adjust control parameters in real time could enhance the efficiency and robustness of underwater robotic systems. These advancements would contribute to optimizing force control for precise, adaptable interactions in unstructured marine environments.

Acknowledgment

I would like to extend my sincere gratitude to all the board members for their invaluable guidance and support throughout this research. Their expertise and insights have been instrumental in shaping the direction of this work.

A special thank you to Micah Prendergast, whose daily supervision and mentorship helped me navigate the steep learning curve of robotics. His patience and technical knowledge were crucial in refining my approach and deepening my understanding in this field.His perspective and strategic thinking have been essential in maintaining focus and direction.

I am also deeply grateful to Rudy Helmons, who not only helped me see the bigger picture but also provided the motivation and encouragement I needed during the most challenging moments of this journey.

Lastly, my heartfelt thanks to Andrea Coraddu for his guidance in ensuring that this research remains relevant to my Marine Technology track.

Their insights helped bridge the gap between theoretical developments and real-world applications, reinforcing the importance of this work in practical marine engineering. Their collective contributions have been instrumental in the completion of this research.

Introduction

Impedance control strategies are vital for enhancing the performance of underwater robotic systems, particularly in complex environments such as soft seabeds used in deep-sea mining and environmental monitoring. This research builds upon established methods to refine the interactions between robots and dynamic underwater terrains, focusing on the precision and adaptability required for effective operation. By examining existing methodologies and their results, this study identifies key areas for technological advancement and operational efficiency, with the aim of exploring solutions for deep sea robotics control strategies.

0.1. Background

0.1.1. Impedance Control

Impedance control is a widely researched method for managing robotic interactions with dynamic and unstructured environments. This technique focuses on regulating the relationship between force and motion by adjusting parameters such as stiffness, damping, and inertia. The literature highlights impedance control's adaptability in managing tasks where precise force regulation is critical, such as tool interaction, surface manipulation, and robotic grasping[1][2].

In underwater robotics, impedance control has been successfully applied to handle challenges arising from unpredictable hydrodynamic forces and varying surface properties. Studies emphasize the trade-off between compliance and stability: lower stiffness offers better compliance but can lead to inaccuracies, whereas higher stiffness improves stability but may increase sensitivity to disturbances [3] [4]. However, there is limited analysis of how fine-tuning stiffness affects penetration depth during soft-surface interactions, such as with soft sea beds.

Recent advancements in impedance control have explored adaptive strategies that dynamically adjust stiffness based on real-time feedback, enhancing the robot's ability to interact with unpredictable environments [5][6]. These studies provide valuable insights into optimizing impedance control for underwater robotics.

0.1.2. Sandbed Penetration Studies

Robotic penetration of granular materials like sand is a specialized application requiring careful force and position control. Often the focus is on the mechanical challenges posed by such environments, including their deformable and variable nature [7][8]. Accurate depth control is critical for applications like sediment sampling, resource extraction, and ecological monitoring.

Existing studies highlight the difficulty of maintaining consistent penetration depth due to factors such as uneven surface composition, compaction resistance, and granular flow behavior [9][10]. However, many works rely on simplified assumptions or static control strategies, leaving a gap in understanding how dynamic control techniques like impedance control perform in these scenarios. Additionally, there is limited exploration of penetration accuracy in underwater sandbeds, especially under varying environmental conditions (e.g., dry vs. submerged sand).

0.1.3. Trajectory Mapping with Point Clouds

Point cloud data has revolutionized trajectory planning in robotic systems by providing detailed 3D representations of environments. In terrestrial and controlled laboratory settings, depth cameras such as the Intel RealSense D455 are commonly used to generate point clouds. The RealSense D455 operates using active infrared stereo vision, capturing depth information through structured light projection and stereo image processing. This method is well-suited for environments with consistent lighting conditions and relatively short depth ranges. However, in underwater applications, depth perception and environmental mapping rely on significantly different technologies due to the attenuation of light and the challenges posed by murky water. Instead of infrared-based stereo vision, various types of sonar, including S-band and X-band systems, are employed to map the topology of the seabed with greater range and robustness to turbidity [11][12]. These sonar-based techniques enable large-scale seabed mapping, providing depth information that is critical for underwater navigation and robotic trajectory planning.

The literature emphasizes the advantages of point cloud mapping for unstructured environments, allowing robots to navigate obstacles, adapt to varying surface contours, and optimize paths for task execution [13][14]. While most research focuses on navigation and obstacle avoidance, fewer studies have combined point cloud-based mapping with force-sensitive tasks like sandbed penetration. The integration of this technique with impedance control to ensure accurate force application along mapped trajectories represents a small niche to study [15][16].

0.1.4. Identified Gaps

While significant advancements have been made in impedance control, granular material interactions, and trajectory mapping, small gaps remain unaddressed:

- **Integration of Control and Mapping:** Limited studies explore the integration of impedance control with point cloud-based trajectory mapping for tasks requiring precise surface interaction [2][11].
- **Dynamic Environmental Conditions:** Research has largely overlooked how control strategies adapt to varying conditions like dry vs. submerged sand and flat vs. curved surfaces [10][8].
- **Quantitative Analysis of Stiffness:** The specific impact of joint stiffness tuning on penetration depth accuracy has not been systematically studied, particularly in tasks involving dynamic trajectories [3].
- **Comprehensive Testing:** Most experiments focus on simplified scenarios, with limited cross-condition analysis of control strategies under realistic operational conditions [4][16].

0.2. Research Objective

In the realm of underwater robotics, the ability to adapt to the dynamic and complex seabed terrain is crucial for operational success. As robotic technologies evolve, there is an increasing demand for systems that can effectively navigate and interact with these unpredictable environments. Impedance control, which adjusts the mechanical properties of a robot in response to environmental feedback, presents a significant opportunity to enhance this interaction. This study is motivated by the need to understand how variations in impedance controller settings, particularly stiffness, can optimize robotic performance in subsea operations.

The objective of this study is to evaluate the trade-offs between different impedance controller stiffness configurations in robotic interactions with soft seabed surfaces. Rather than assuming higher stiffness leads to better performance, this research examines whether lower stiffness can provide advantages in real-world seabed conditions by improving adaptability in uncertain environments.

This study aims to achieve the following:

- **Assess the Relationship Between Joint Stiffness and Environmental Interaction:** Investigate how varying stiffness levels impact penetration depth accuracy, force consistency, and overall interaction with the sandbed.
- **Identify the Trade-offs Between Stiffness, Force data, and Stability:** Analyze how different impedance settings affect the robot's ability to respond to unexpected seabed features, such as sudden changes in terrain or obstacles.
- **Evaluate Controller Performance in Simulated Underwater Conditions:** Determine whether a middle-ground stiffness configuration may be more suitable for real-world applications where accuracy, consistency, and adaptability must be balanced.

By leveraging point cloud-mapped trajectories, the study replicates real-world seabed navigation challenges to understand how impedance control strategies can be optimized for subsea robotic applications.

0.3. Scope and Limitations

0.3.1. Scope

This research explores the role of impedance control in underwater robotic systems, focusing specifically on how different stiffness configurations influence force distribution and trajectory tracking when interacting with deformable surfaces. The study utilizes a controlled test environment to isolate and quantify the effects of stiffness tuning on penetration accuracy and force response.

0.3.2. Limitations

Several factors constrain the applicability of the findings:

- **Mismatch Between Experimental and Real-World Robotic Systems:** The experiments are conducted using a **7-degree-of-freedom (DOF) KUKA iiwa 7 robot**, while industrial deep-sea mining (DSM) robots typically use **single-DOF actuators**. This difference introduces a discrepancy in force distribution and control complexity, affecting the generalizability of results.
- **Controlled Test Environment:** The study is conducted in a **2-meter laboratory water tank**, which lacks realistic environmental factors such as **currents, irregular sediment composition, and hard obstacles (e.g., rocks or debris)**. These real-world elements could significantly affect force feedback, penetration resistance, and trajectory deviations.
- **Restricted Speed Testing Due to Safety Features:** The **KUKA robot's internal safety limitations prevent testing at higher speeds**, which may limit the ability to analyze how increased velocity influences force variation and penetration accuracy.
- **Simplified Sandbed Model:** The **homogeneous sand layer** used in experiments does not fully replicate the stratified and highly variable nature of real seabeds, where differences in grain size, compaction, and moisture levels affect penetration behavior.
- **Fixed Control Parameters:** While this study adjusts **joint stiffness**, other impedance parameters such as **damping and inertia remain constant**. In real-world applications, dynamic tuning of multiple impedance parameters would be necessary to optimize interaction forces based on seabed variability.
- **Sensor Accuracy and Data Processing Limitations:** The **3D camera and point cloud mapping system** introduce potential errors due to **sensor noise, data processing delays, and depth estimation inaccuracies**. These factors may cause minor trajectory deviations that affect penetration depth accuracy measurements.

By acknowledging these limitations, the study aims to provide insights into impedance control trade-offs while recognizing the constraints of a laboratory setting compared to real-world seabed conditions.

0.4. Research question

What are the trade-offs associated with different impedance control settings in underwater conditions, and how do these settings affect the accuracy of sandbed penetration depth along a trajectory mapped using 3D camera point cloud data?

Methodology

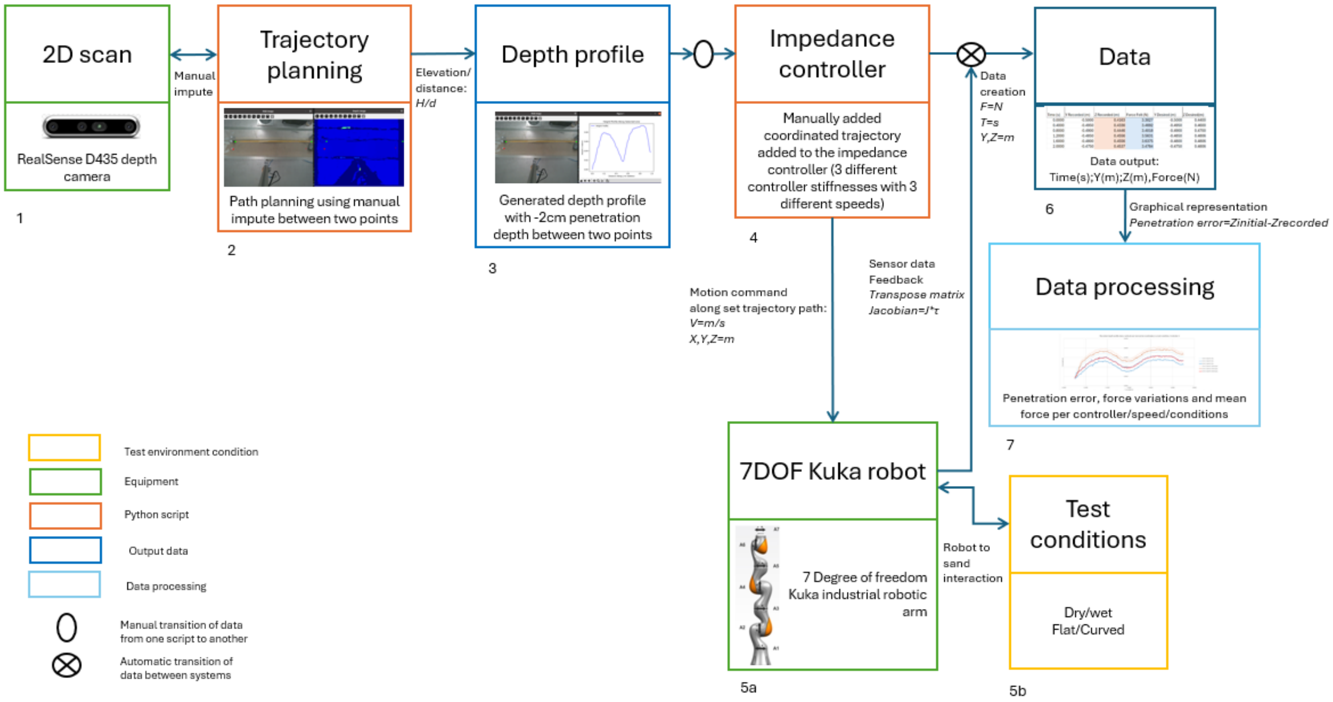

2D scan
RealSense D435 depth camera
1
Manual impute
Trajectory planning
Path planning using manual impute between two points
2
Elevation/ distance: H/d
Depth profile
Generated depth profile with -2cm penetration depth between two points
3
Impedance controller
Manually added coordinated trajectory added to the impedance controller (3 different controller stiffnesses with 3 different speeds)
4
Data creation
F=N
T=s
Y,Z=m
Data
Data output: Time(s);Y(m);Z(m),Force(N)
6
Graphical representation
Penetration error=Zinitial-Zrecorded
Data processing
Penetration error, force variations and mean force per controller/speed/conditions
7
Motion command along set trajectory path:
V=m/s
X,Y,Z=m
Sensor data Feedback
Transpose matrix Jacobian=J*τ
7DOF Kuka robot
7 Degree of freedom Kuka industrial robotic arm
5a
Robot to sand interaction
Test conditions
Dry/wet
Flat/Curved
5b
Test environment condition
Equipment
Python script
Output data
Data processing
Manual transition of data from one script to another
Automatic transition of data between systems

Figure 1: Experimental procedure block-diagram

0.5. Experimental Setup

The experimental setup consists of a KUKA iiwa 7 robotic arm equipped with an impedance controller, operating within a controlled test environment. The system includes a sandbed, a water tank, a RealSense D455 camera for trajectory mapping, and a force estimation framework using joint torque measurements. The following figure provides an overview of the experimental configuration, indicating the key components used in the study.

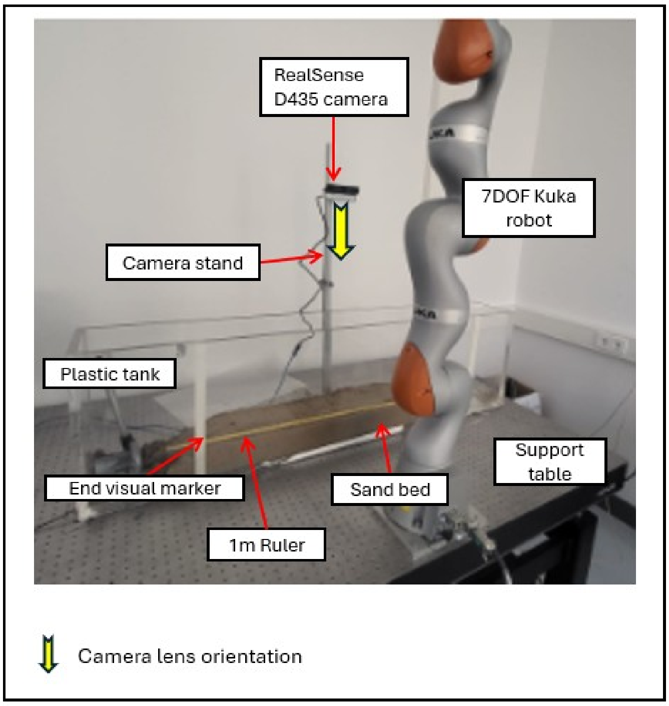


Figure 2: First step of experimental setup with key system components.

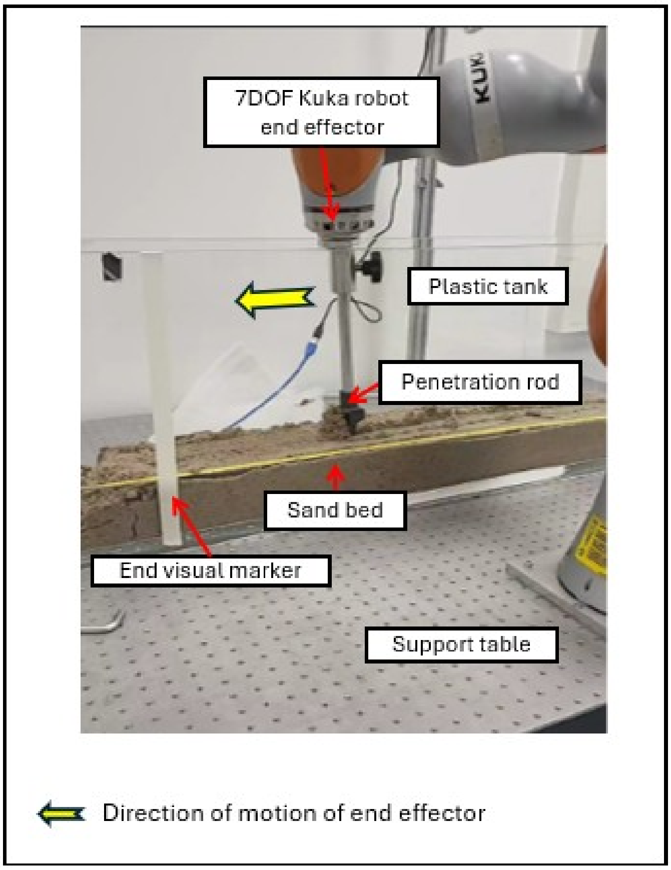


Figure 3: Second step of experimental setup with key system components.

Robotic System: The KUKA robotic arm follows a predefined trajectory while trying to maintain a consistent 2 cm penetration depth into the sandbed. A rod, 34 cm in length with a 4 cm width, is attached at a 60-degree angle to the end effector. The inclined angle minimizes sand buildup, reducing obstruction effects.

Plastic Tank: The test environment consists of a 200 cm long, 40 cm wide, and 100 cm tall plastic tank with 4.5 cm thick plexiglass walls. For submerged conditions, 20 cm of water was added to the tank, ensuring full submersion of the sand and the robotic tool while preventing excessive buoyancy effect that could influence force readings.

Sandbed: The sandbed consists of *Stonewish filling sand* (0–2 mm grain size). For flat conditions, the sand is uniformly leveled at a 5 cm depth. For curved conditions, a terrain profile was created with two peaks (8 cm and 9 cm height) and one valley (2 cm depth). After each test, a flat plate matching the tank's width was used to reshape the sand according to a predefined profile.

0.6. Camera System and Trajectory Mapping

Camera System: The RealSense D455 camera, mounted 60 cm above the sanded, captures RGB and depth data for trajectory mapping. The original camera intrinsics do not work in 2D mapping, therefore the camera is mounted parallel to the tank's bottom on a fixed rod and in the middle of the desired trajectory to maintain parallelism on each side of the trajectory from point y=0.

In submerged conditions, ensuring an undisturbed water surface was critical for accurate depth imaging. Reflections from surface ripples or turbidity in the water could interfere with infrared depth measurements. To minimize distortions:

- The water surface was allowed to settle for a sufficient time between runs.
- Low ambient lighting was maintained to prevent infrared interference.
- The water was clear but not perfectly transparent, allowing effective depth sensing.

Calibration: A one-meter ruler was placed in the tank to define a length scale in the camera's RGB feed. Green dots were marked at reference points to calibrate intrinsic parameters.

Trajectory Definition: The trajectory was mapped using two predefined red dots that are manualy picked by clicking in the RGB feed screen, marking the start and end positions. Two visual markings should be present on the screen to indicate to the used where to put the dots like the empty sand patch showed on 4. The motion path spans from $y = -0.5$ m to $y = +0.5$ m with a fixed horizontal position at $x = +0.45$ m. The Z depth coordinates are automatically subtracted -0.02m from the defined trajectory as seen4 before delivering the desired trajectory 5.

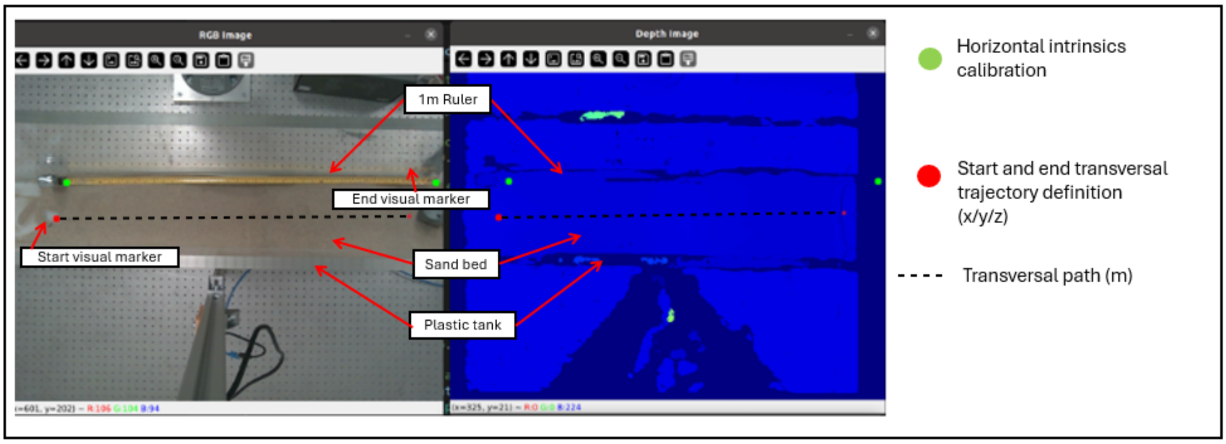


Figure 4: Trajectory definition example with axis labeled (x as horizontal and y as depth). Left photo is the RGB image and right photo is the Depth Infrared image.

Point Cloud Processing: The camera captures the elevation spectrum of the sandbed along the defined path. These depth coordinates are used to create the reference trajectory for the robot.

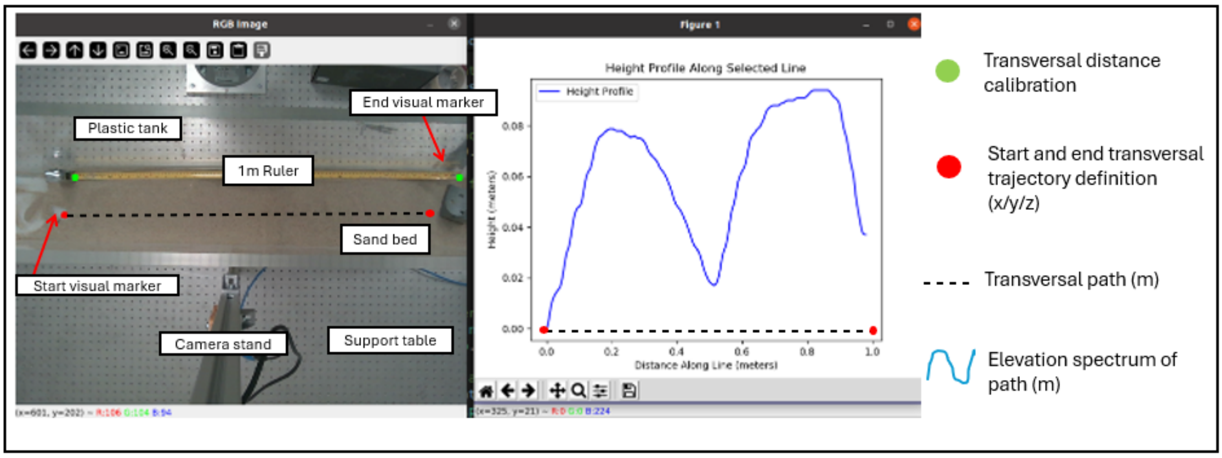

RGB Image
Plastic tank
End visual marker
1m Ruler
Sand bed
Start visual marker
Camera stand
Support table
(x=601, y=202) ~ R:106 G:104 B:94
Figure 1
Height Profile Along Selected Line
Height Profile
Height (meters)
0.00
0.02
0.04
0.06
0.08
0.0
0.2
0.4
0.6
0.8
1.0
Distance Along Line (meters)
(x=325, y=21) ~ R:0 G:0 B:224
Transversal distance calibration
Start and end transversal trajectory definition (x/y/z)
Transversal path (m)
Elevation spectrum of path (m)

Figure 5: Trajectory coordinate spectrum example (desired trajectory adjusted by -2 cm for penetration depth). Left picture is the RGB image, right picture is the generated elevation spectrum trajectory.

0.7. Experimental Procedure

Control Variations:

- Three impedance controllers were tested, each with different stiffness values: 300 N/m, 200 N/m, and 100 N/m.
- Three velocities were tested per controller: 0.0125 m/s, 0.025 m/s, and 0.05 m/s, resulting in nine total combinations.

The selected stiffness values represent a range from highly flexible to highly rigid controllers. This range was chosen based on preliminary studies in robotic impedance control for deformable surfaces and aligns with real-world considerations for seabed exploration.

Environmental Conditions:

- Tests were conducted in four conditions: dry-flat, dry-curved, submerged-flat, and submerged-curved.

Robot Movement: The robot starts at its initial position of z=-0.5m and follows the defined trajectory while trying to maintain a penetration depth of 2 cm. the rod attached to the end effector is reactive to the robot motion and tries to be set at a normal angle to the sand tank bottom and enters contact with the sand bed thus creating the penetration in the sand.

0.8. Data Collection and Processing

Force Estimation: Forces at the tip of the rod are computed using joint torque measurements mapped through the Jacobian transpose method. This approach involves converting the joint torques τ into end-effector forces **F** using the Jacobian matrix **J** of the robotic arm. The relationship between joint torques and end-effector forces can be expressed by the following equation:

$$\mathbf{F} = \mathbf{J}(\mathbf{q})^T \tau$$

Where:

- **F** is the vector of forces and moments at the end-effector.
- $\mathbf{J}(\mathbf{q})^T$ is the transpose of the Jacobian matrix, which depends on the current configuration **q** of the robot's joints.
- τ is the vector of joint torques.

The Jacobian matrix **J** is a function of the robot's joint angles and links the velocities and forces in the joint space with those in the Cartesian space. This matrix is critical for understanding how movements and forces translate from the robot's motors (at the joints) to the tip of the rod (the end-effector). The force at the tip of the rod is then used

Rod Position Recording: The z-coordinate of the rod is logged during motion, allowing for a direct comparison between the planned and executed trajectories.

Trajectory Comparison: The planned trajectory from point cloud data is compared with the recorded trajectory from the robot. The desired penetration depth is $Z_{extdesired} - 0.02$ m, and any deviation from this is computed as an error.

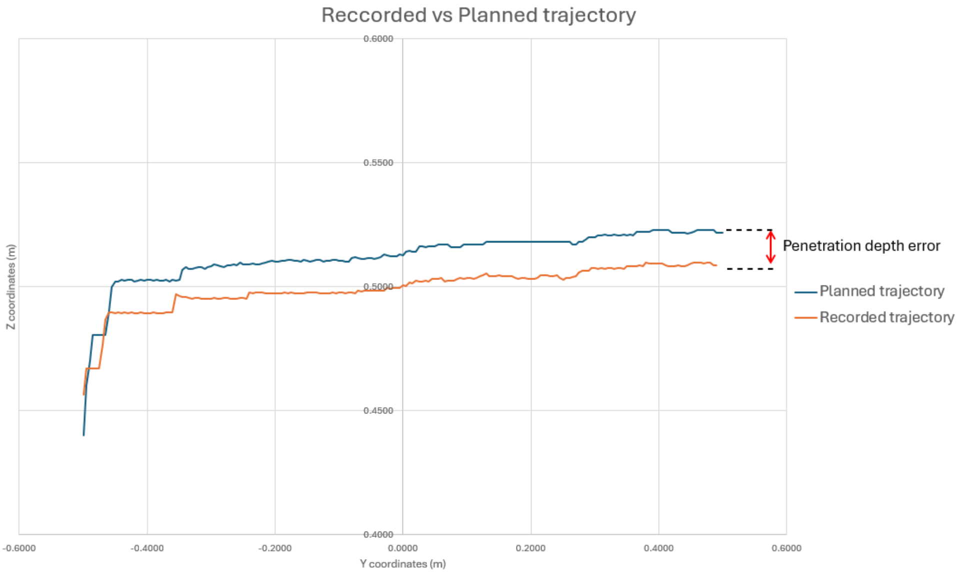
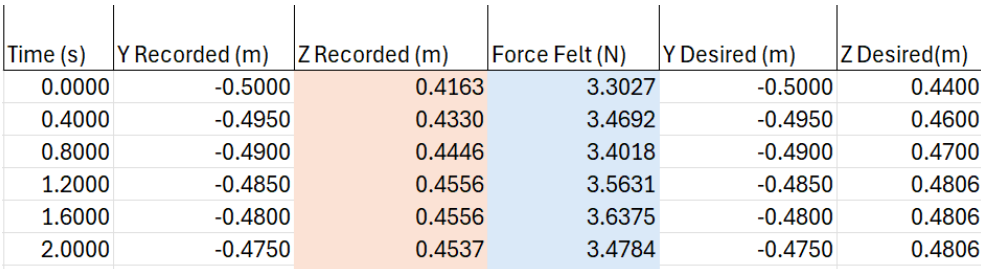


Figure 6: Example of recorded vs. planned trajectory.

Extracted data format: The extracted data format for each test comprised of [Time;Y recorded; Z recorded, Y desired, Z desired]



Time (s)	Y Recorded (m)	Z Recorded (m)	Force Felt (N)	Y Desired (m)	Z Desired(m)
0.0000	-0.5000	0.4163	3.3027	-0.5000	0.4400
0.4000	-0.4950	0.4330	3.4692	-0.4950	0.4600
0.8000	-0.4900	0.4446	3.4018	-0.4900	0.4700
1.2000	-0.4850	0.4556	3.5631	-0.4850	0.4806
1.6000	-0.4800	0.4556	3.6375	-0.4800	0.4806
2.0000	-0.4750	0.4537	3.4784	-0.4750	0.4806

Figure 7: Example of recorded data table.

Data Processing for Speed Normalization: Since force data was collected every 0.4 seconds (less time was considered between but would have lead to data files too large to be processed), lower-speed tests resulted in more data points than higher-speed tests. To standardize the dataset across all speeds for consistent comparison, a uniform resampling method was applied, ensuring equalized row counts across test cases, with 200 rows of data per test. This method consisted in using the test with the least data row's and adapting the tests with a greater number of data row's to match the same Y values, thus enabling the comparison of Z values at a same vertical distance.

0.9. Sand Surface Consistency

After each test, the sand surface was reset using a structured procedure:

- For flat tests, the sand was leveled to a uniform 5 cm depth.
- For curved tests, a shaping guide was placed on the sides of the tank to recreate two peaks (8 cm and 9 cm) and a valley (2 cm).
- A wide, flat plate was passed over the sand to ensure consistency between tests.

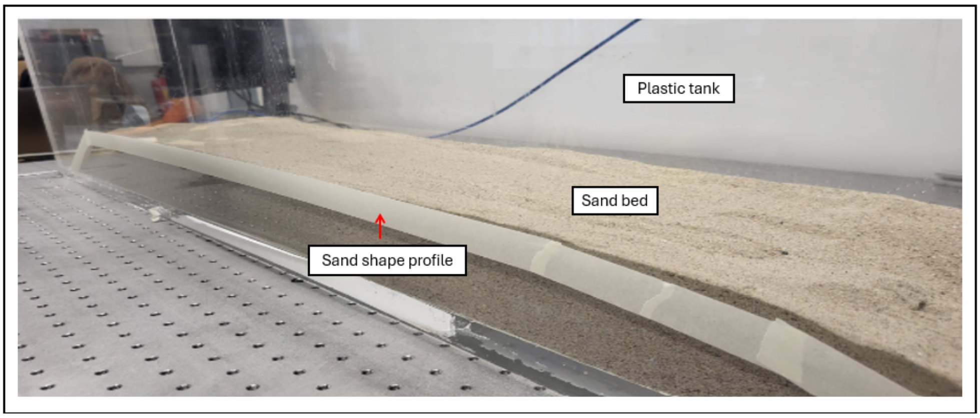


Figure 8: Example of contour guide profile used for sand profile shaping in flat conditions.

The controlled leveling of the sandbed ensured repeatability in experiments by maintaining uniform compaction and minimizing variations that could affect force measurements.

0.10. Data Analysis

Performance Metrics:

- **Penetration Accuracy:** Measures how closely the robot maintains the desired 2 cm penetration depth.
- **Force Consistency:** Evaluates force fluctuations and stability during movement.
- **Controller Effectiveness:** Determines which stiffness-speed combination provides optimal performance.

Visualization:

- Graphs comparing planned vs. executed trajectories.
- Force profiles along the y-axis to analyze stability.
- Computation of force standard deviation for consistency assessment.

This structured methodology ensures clarity, reproducibility, and relevance for future research.

Results Analysis

0.11. Mean Penetration Error by Controller Type

The mean penetration error varies significantly based on the stiffness of the controller. To illustrate the penetration behavior before averaging, Figure 9 and 10 presents the raw penetration depth data over the horizontal distance covered for Controller 1, showing how penetration fluctuates across all environmental conditions (three speeds, flat and curved surfaces, dry and submerged). This provides insight into real-time deviations before summarizing the data.

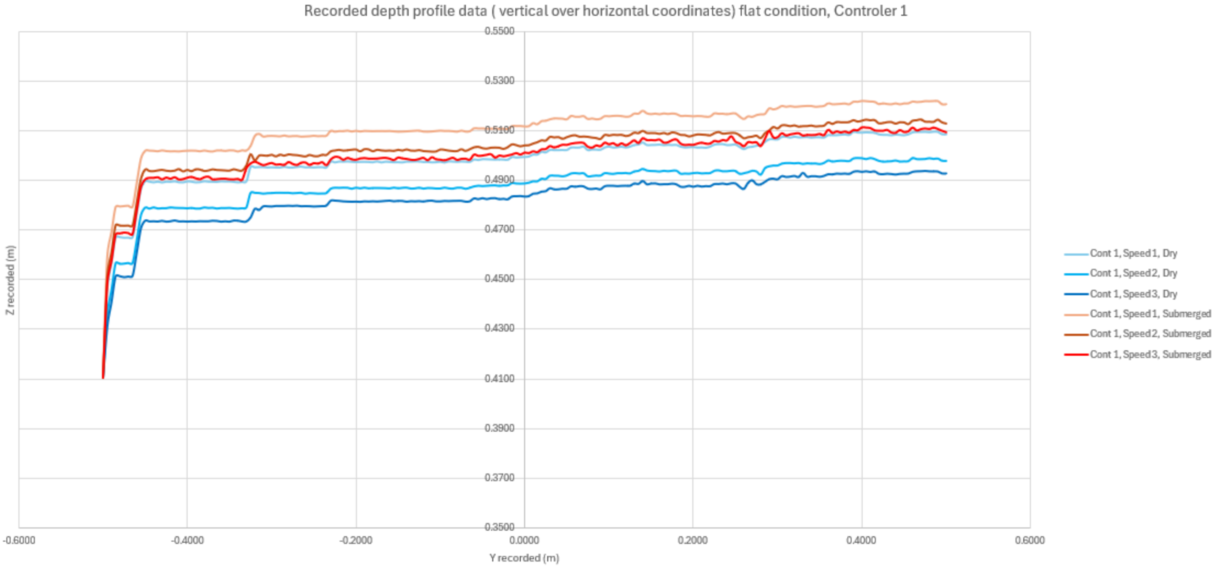


Figure 9: Raw data presentation of penetration depth Z over the vertical Y distance for Controller 1 across all speeds in with a flat surface in both dried and flat conditions.

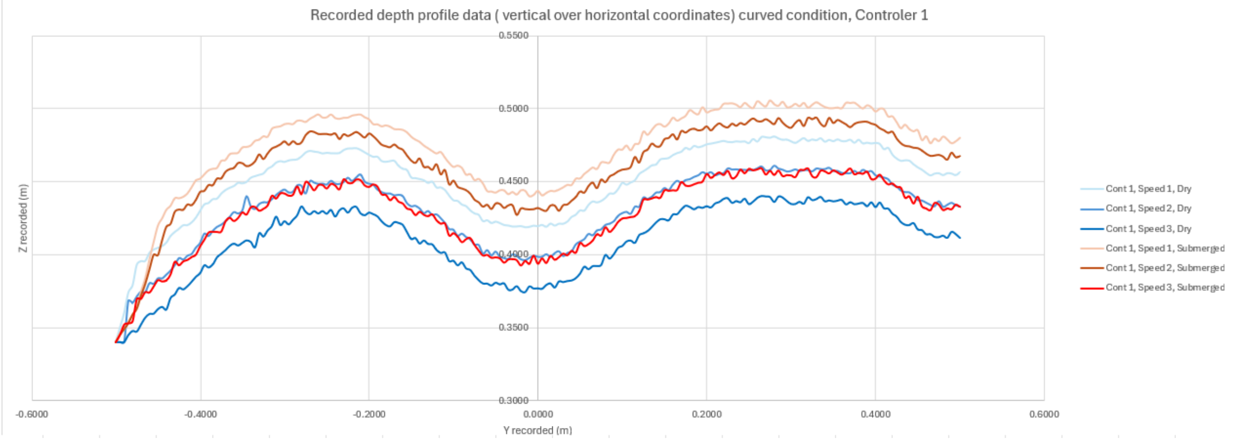


Figure 10: Raw data presentation of penetration depth over time for Controller 1 across all speeds in with a curved surface in both dried and flat conditions.

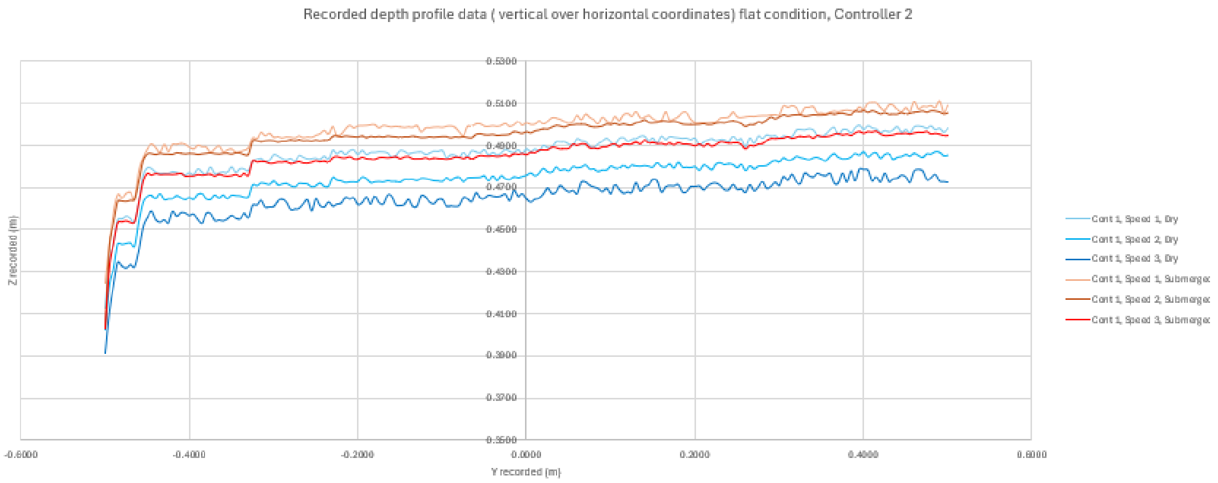


Figure 11: Raw data presentation of penetration depth Z over the vertical Y distance for Controller 2 across all speeds in with a flat surface in both dried and flat conditions.

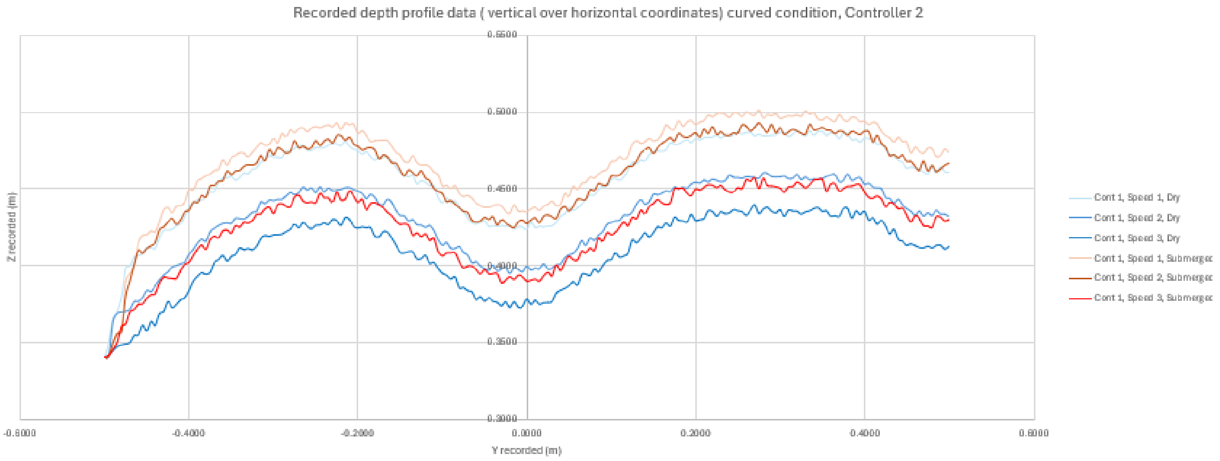


Figure 12: Raw data presentation of penetration depth over time for Controller 2 across all speeds in with a curved surface in both dried and flat conditions.

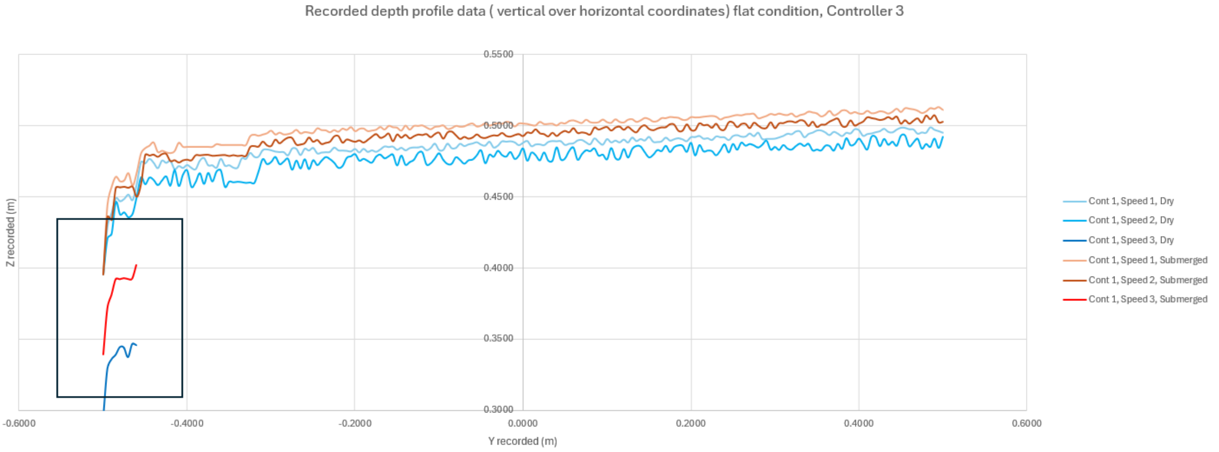


Figure 13: Raw data presentation of penetration depth Z over the vertical Y distance for Controller 3 across all speeds in with a flat surface in both dried and flat conditions.

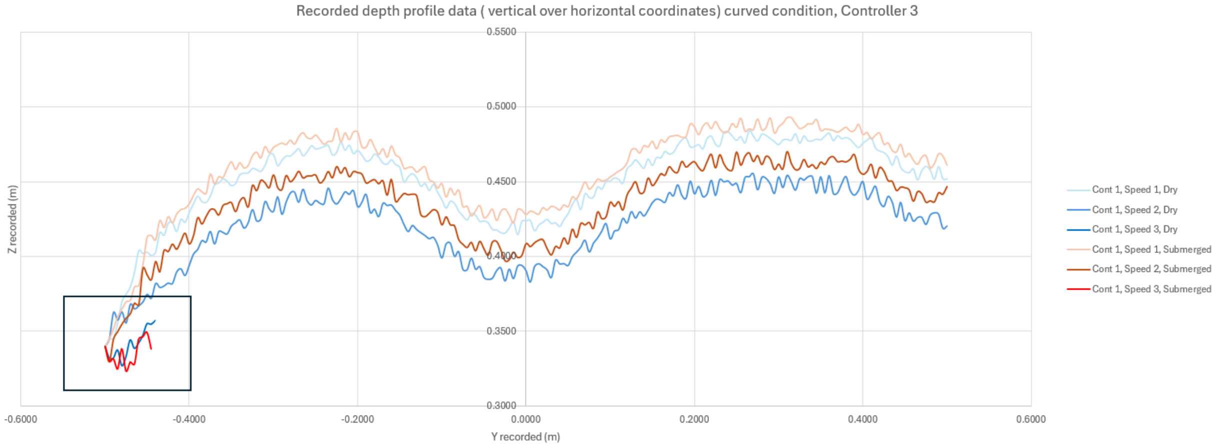


Figure 14: Raw data presentation of penetration depth over time for Controller 3 across all speeds in with a curved surface in both dried and flat conditions.

In both graphs of controller 3, tests done at the higer speed of 0.05m/s only show a short penetration path in the Y direction (black boxes). This is due to the saftey features of the KUKA iiwa 7 robot being triggered. Indeed, if the robot is requiring more force to keep up with the desired trajectory and speed, this can lead to the maximum allowable exerted force of the KUKA to be reached which leads to a full stop, thus failing all tests at minimum strifness (100 N/m) and maximum speed (0.05m/s).

To provide an overall view, Figure 15 presents the mean penetration error per controller, averaged over all test conditions. This includes variations in speed, surface type, and environmental conditions, ensuring a comprehensive representation of controller performance.

- Controller 1 (300 N/m): 0.013 m
- Controller 2 (200 N/m): 0.025 m
- Controller 3 (100 N/m): 0.03 m

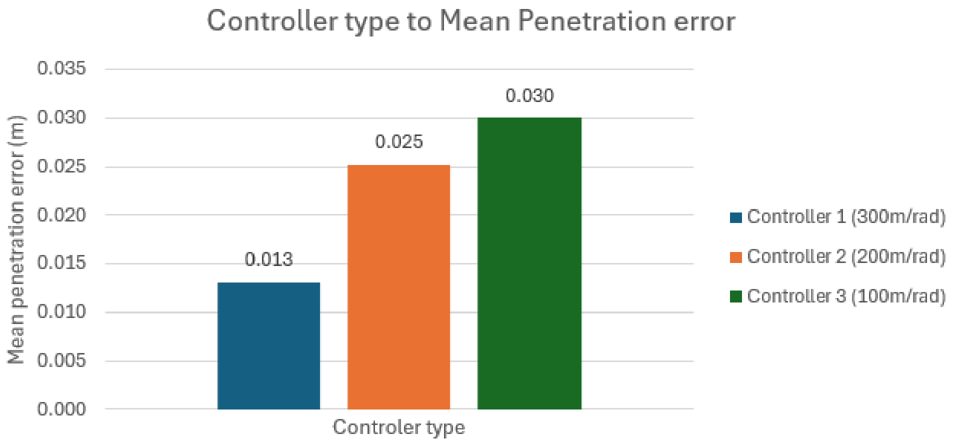


Figure 15: Penetration Error vs. Controller Type. This figure represents the **average penetration error per controller**, considering variations in speed (three levels), surface type (flat/curved), and environment (dry/submerged).

The results indicate that higher stiffness controllers exhibit lower penetration errors, following the desired trajectory more accurately. However, as controller stiffness decreases, penetration deviations increase, particularly under high-speed conditions and curved surface scenarios.

0.12. Effect of Speed and Environmental Conditions on Penetration Error

The relationship between speed and penetration error is analyzed for all three controllers, as shown in Figures 16, 17 and 18. Across all controllers, an increase in speed consistently results in greater penetration error, indicating reduced tracking accuracy at higher velocities. This trend is evident regardless of the controller's stiffness settings:

- At lower speeds, all controllers maintain relatively low penetration errors, with Controller 1 (300 N/m) performing the best overall. The errors are minimal, and the robotic system closely follows the desired trajectory.
- As speed increases, penetration errors become more pronounced. Controller 1 continues to exhibit the lowest errors, but the difference between controllers becomes more apparent. Controller 2 (200 N/m) shows moderate errors, while Controller 3 (100 N/m) experiences the highest errors, particularly at the highest speeds.
- At the highest speeds, the system's ability to maintain accurate penetration depth is significantly compromised. Controller 3 was unable to record data at the highest speed due to safety constraints, highlighting the challenges of high-speed operation.

Another observation is that submerged conditions mitigate the impact of increasing speed on penetration error. The presence of water provides fluid resistance, which stabilizes the robotic interaction and reduces errors compared to dry conditions. This effect is most likely due to dilatant strengthening of the sediment, meaning that at higher deformation rates, the sediment becomes stiffer. The restriction of water movement within the low-permeability sandbed increases resistance to penetration, effectively strengthening the substrate and reducing excessive deviations.

Additionally, the type of surface (flat vs. curved) influences how speed affects penetration error. Curved surfaces generally result in higher errors at increased speeds, as the uneven terrain introduces additional challenges for the controller to maintain a steady penetration depth.

Effect of Speed on f Mean Penetration Error per Controller for Flat and Dry Conditions

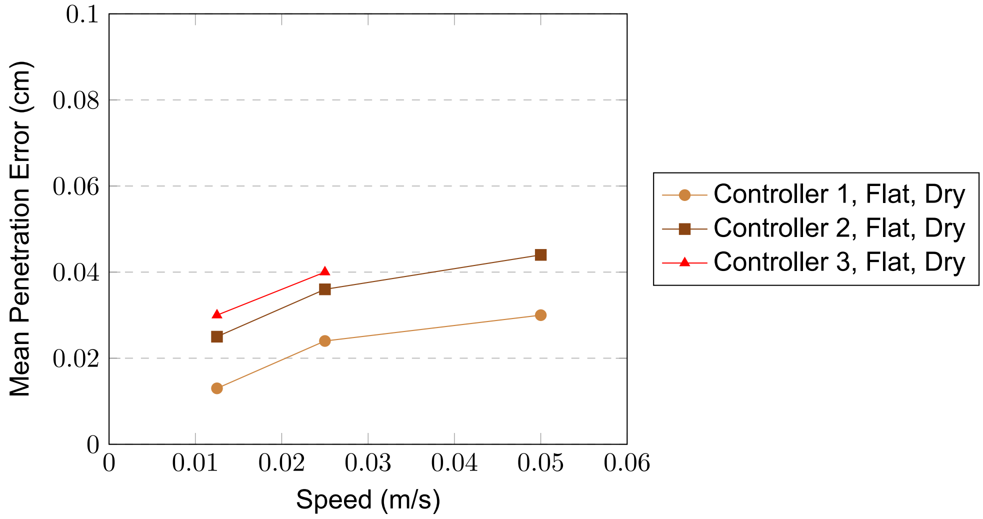


Figure 16: Graphical representation of Mean Penetration Error as a function of speed for different controllers and conditions.

Effect of Speed on Mean Penetration Error per Controller for Curved and Dry Conditions

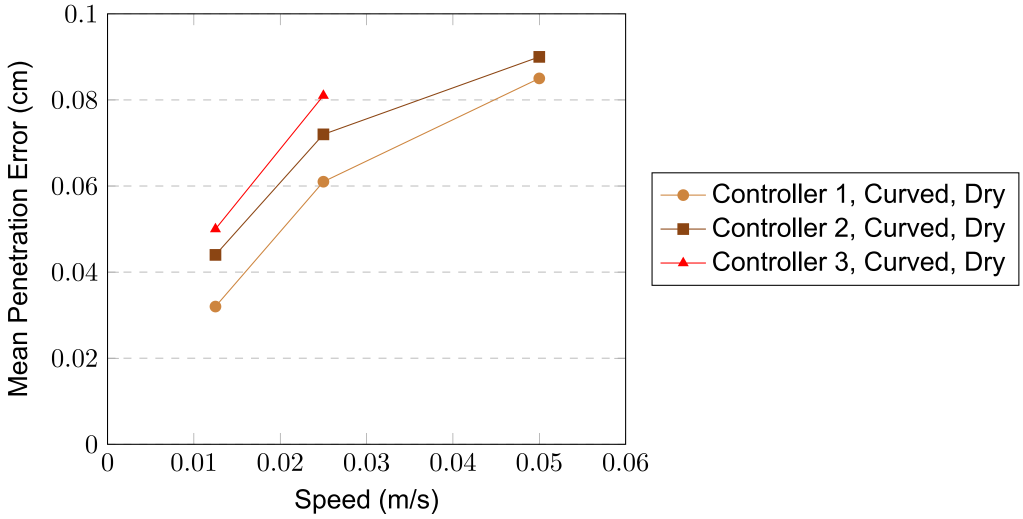


Figure 17: Graphical representation of Mean Penetration Error as a function of speed for different controllers and conditions.

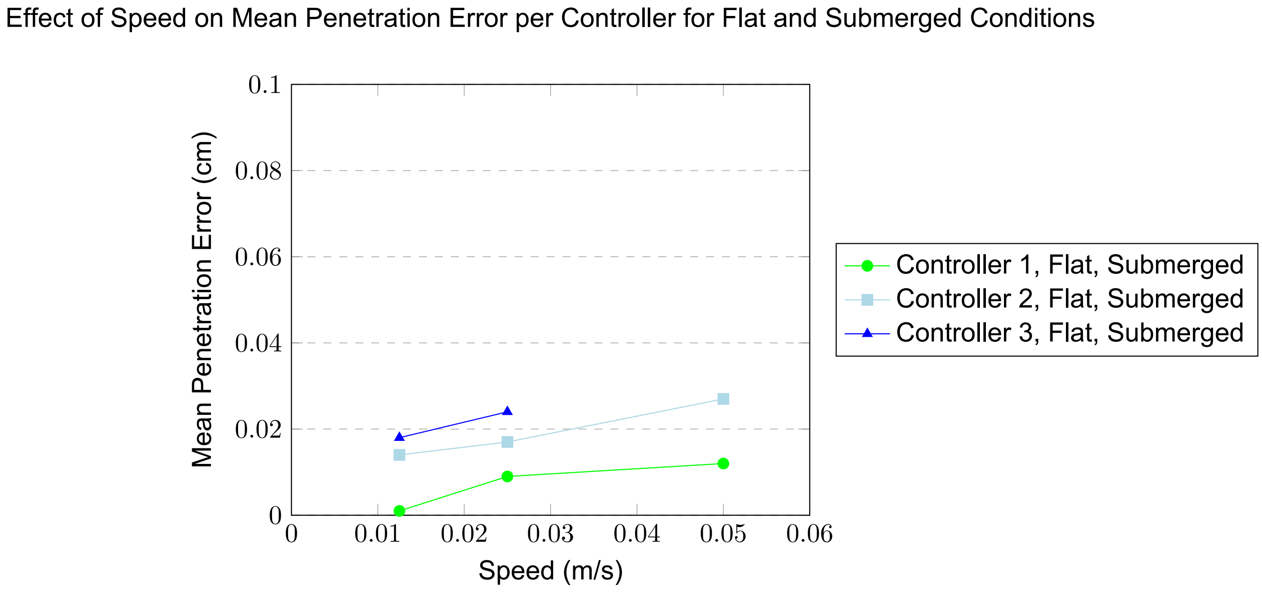


Figure 18: Graphical representation f Mean Penetration Error as a function of speed for different controllers and conditions.

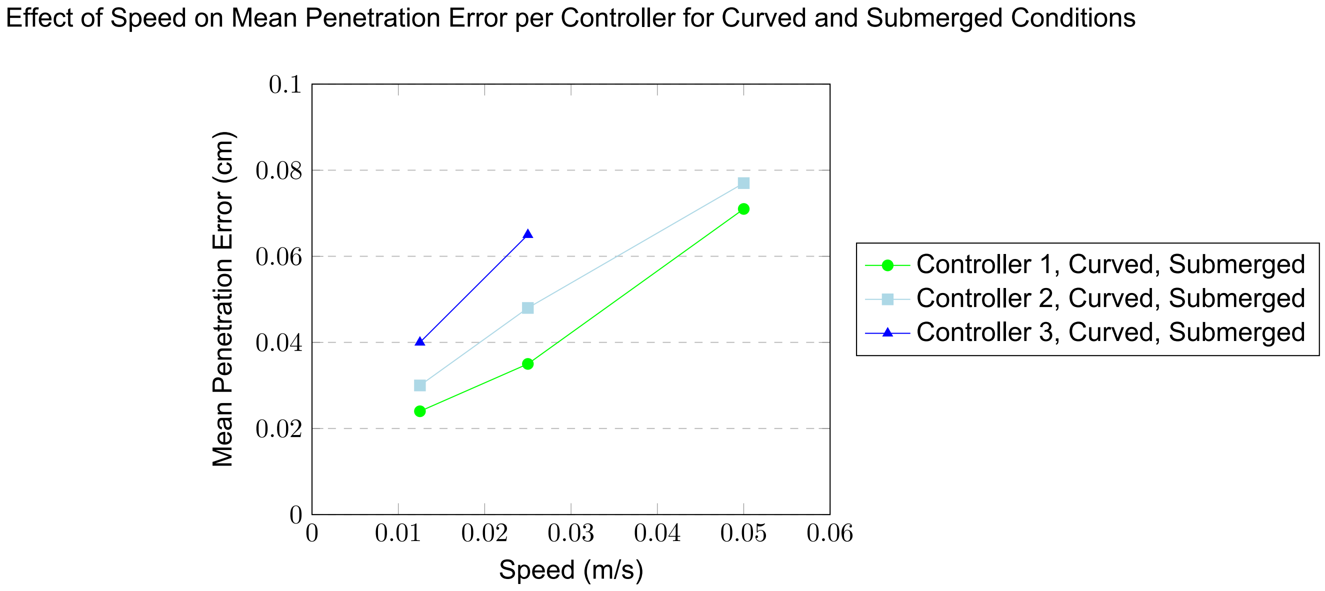


Figure 19: Graphical representation f Mean Penetration Error as a function of speed for different controllers and conditions.

0.13. Effect of Speed and Environmental Conditions on Mean Force

The variation of mean force under different speed, surface, and environmental conditions is analyzed for all three controllers, as illustrated in Figures 21, 22 and 23.

A general trend observed across all controllers is that:

- Increasing speed leads to a rise in mean force, suggesting that at higher velocities, the robot exerts greater interaction forces on the sandbed.
- Surface type influences force application: Curved surfaces generally lead to higher mean forces than flat surfaces, particularly in dry conditions.

- Submerged environments result in lower mean forces compared to dry conditions, likely due to fluid damping effects reducing resistance.

Breaking down the results per controller:

- Controller 1 (300 N/m) maintains relatively stable mean force levels across conditions, with a gradual increase at higher speeds, especially in curved, dry scenarios.
- Controller 2 (200 N/m) shows greater sensitivity to speed changes, with a sharper increase in mean force at higher speeds.
- Controller 3 (100 N/m) exhibits the highest mean force values across all conditions, particularly in dry environments. This suggests that lower stiffness results in greater force oscillations and fluctuations during interaction.

To further analyze the impact of terrain, Figure 20 presents the penetration error. The penetration error is computed by abstracting the desired penetration (2cm) by the recorded penetration (Z desired- Z recorded). This allows for a clearer assessment of whether deviations are due to hills, valleys, or both. In both flat and curved cases, the submerged conditions have a lower average error and present less deviations overall. One noticeable aspect of the figure is that the concentration of error deviations from the average error are not uniform over the y axis. Indeed for speed 1, Curved and submerged case (light blue line), there is an accumulation of error deviation towards the beginning and the middle of the horizontal path. These are two abrupt (first going up, second going down) slopes on the curved trajectory path. This suggests that terrain variation may have affected the penetration error. The green and light blue lines start later on the graph as some skewed values had to be taken out in order to make the graph more readable.

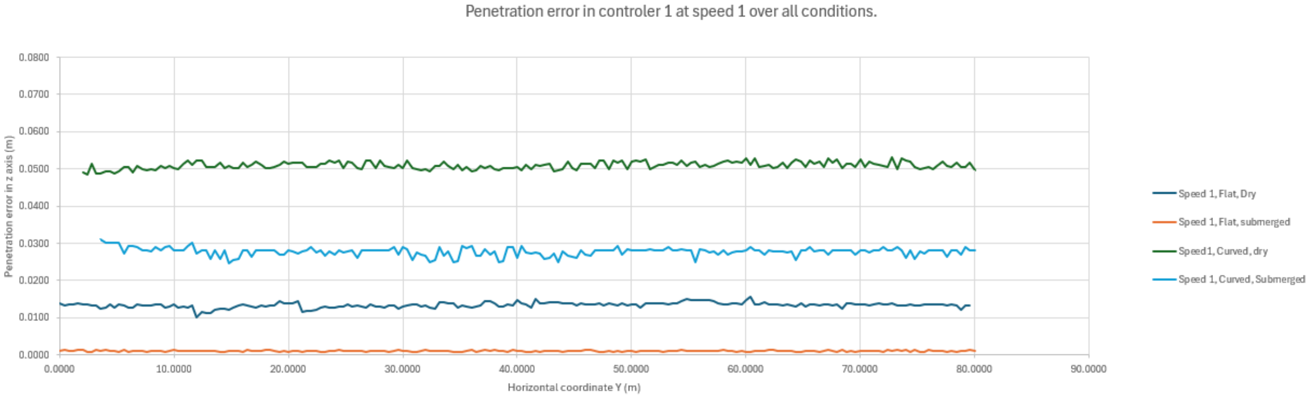


Figure 20: Penetration error over time, showing the impact of terrain variations and fluid medium on error variation.

In addition, if Figures 21, 22, 23 are analyzed in the context of sand cutting theory assuming steady-state models. In submerged conditions, force is expected to increase linearly with velocity due to the uniform resistance provided by the water column. This trend is visible in the submerged test cases, where the force measurements show a proportional increase with speed, indicating that the fluid environment stabilizes the interaction forces and prevents abrupt changes in resistance.

Conversely, in dry conditions, force scales with velocity squared, as increased velocity leads to greater inertia effects and higher resistance from the granular medium. This effect is particularly evident in Figures 21 and 22, where force measurements in dry conditions exhibit a more pronounced increase at higher speeds. The squared scaling behavior suggests that at greater velocities, the tool is experiencing additional resistive forces from the compacting sand, which aligns with expectations from granular flow models.

Furthermore, the increased force fluctuations at higher speeds in dry conditions highlight the instability introduced by rapid deformation of the sandbed. The lack of fluid support in dry conditions leads to greater variance in the measured forces, as seen in Controller 2 and particularly Controller 3, where the lowest stiffness setting results in larger force deviations. This is in contrast to the submerged conditions, where force values remain more stable due to the presence of water, which dampens sudden fluctuations.

The results indicate that while a stiffer controller generally exerts lower mean forces, submerged conditions help mitigate force fluctuations across all configurations. This damping effect is particularly evident for Controllers 2 and 3, where force values are noticeably reduced in the submerged setting. The observed behavior suggests that underwater environments provide natural stabilization, reducing the need for excessive stiffness in impedance control strategies when operating in fluidic conditions.

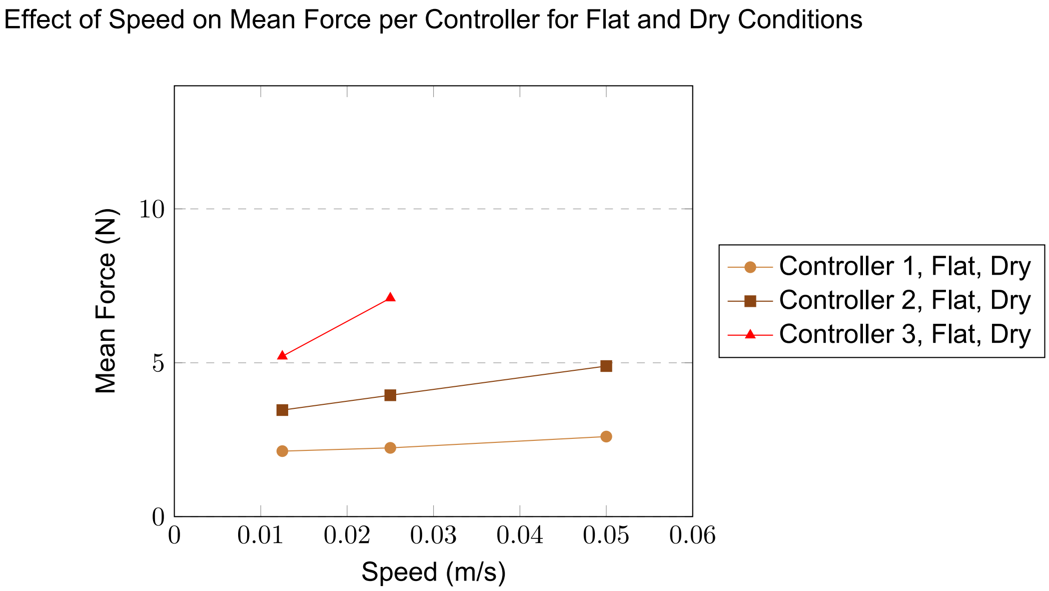

Effect of Speed on Mean Force per Controller for Flat and Dry Conditions

Figure 21: Graphical representation of Mean Force as a function of speed for different controllers and conditions.

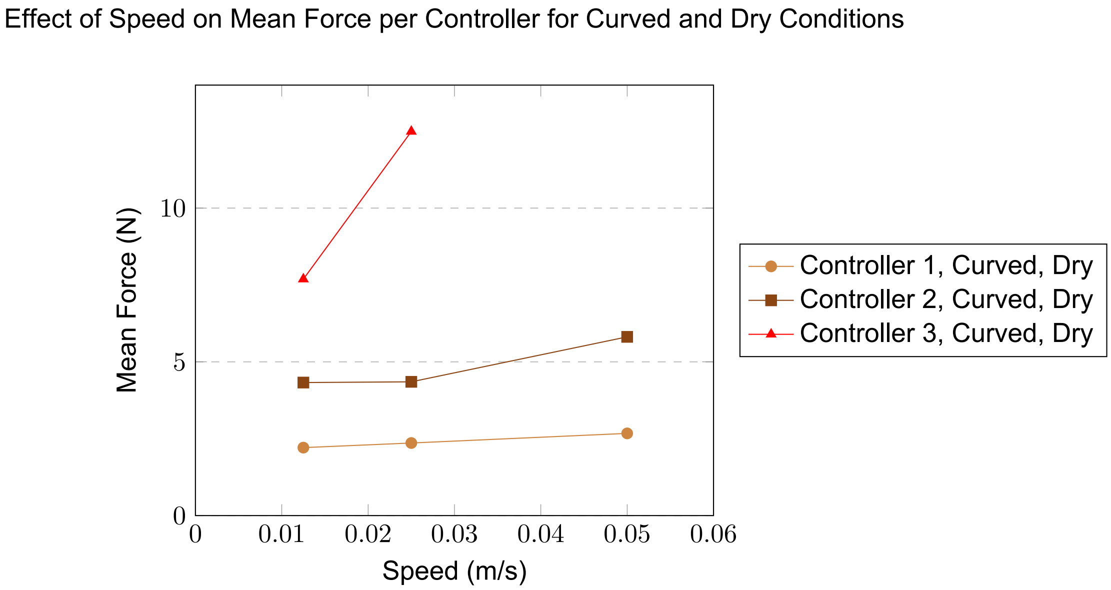

Effect of Speed on Mean Force per Controller for Curved and Dry Conditions

Figure 22: Graphical representation of Mean Force as a function of speed for different controllers and conditions.

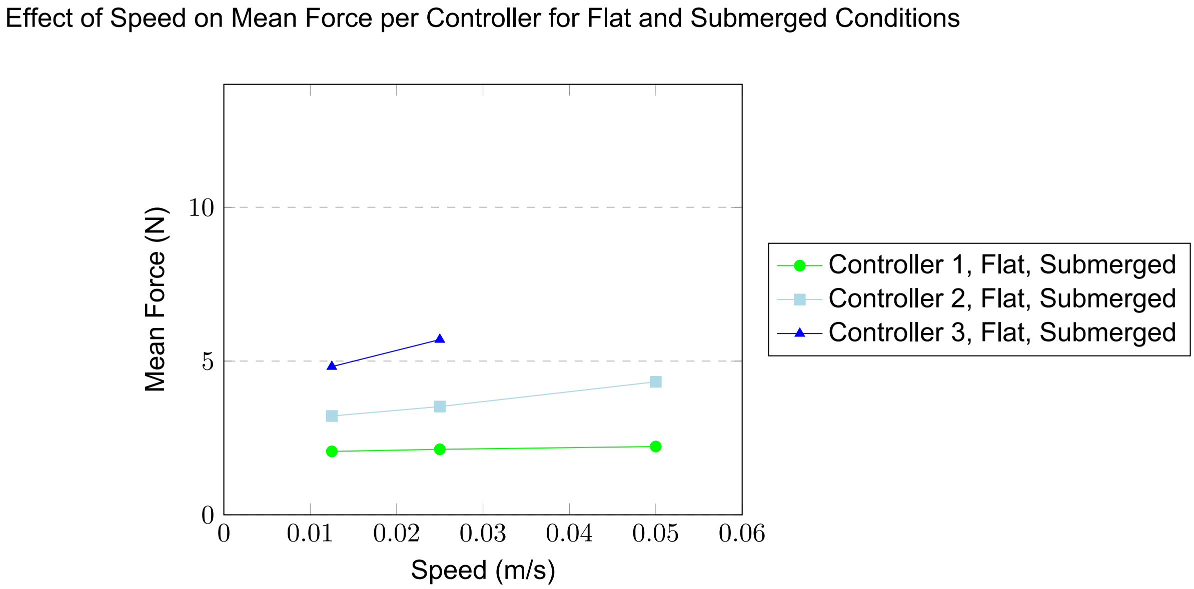


Figure 23: Graphical representation Mean Force as a function of speed for different controllers and conditions.

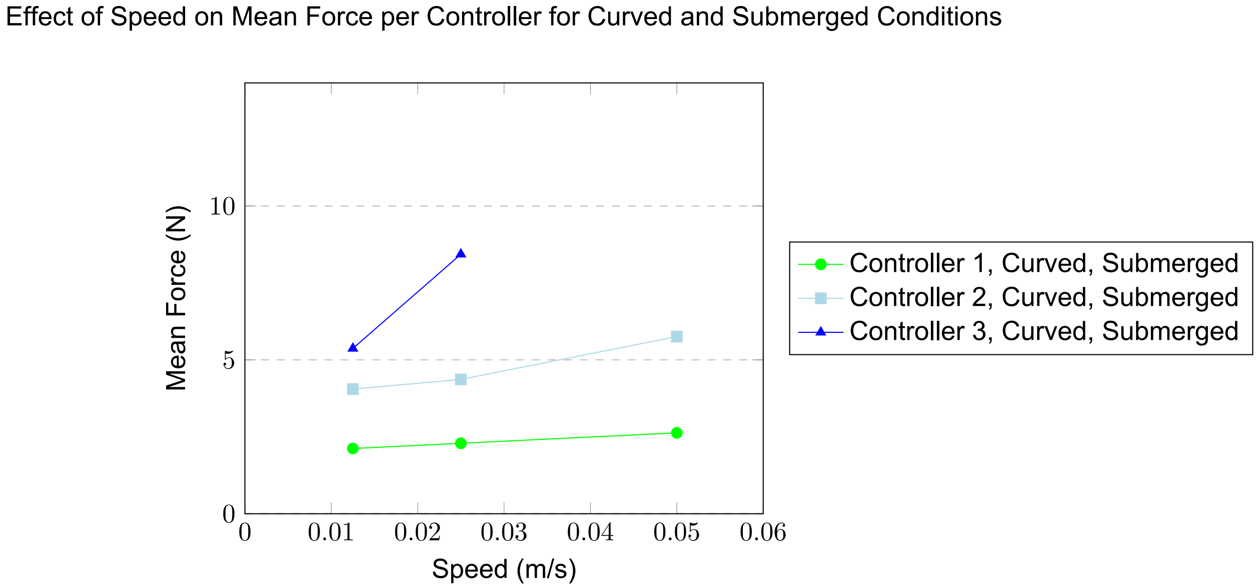


Figure 24: Graphical representation of Mean Force as a function of speed for different controllers and conditions.

0.14. Force Standard Deviation vs. Speed

The relationship between speed and force standard deviation across different controllers is analyzed using Figures 25,26 and 27. Observations are as follow:

- Force standard deviation increases with speed across all controllers, indicating that higher velocities introduce greater force fluctuations.
- Controller 3 (Stiffness = 100 N/m) exhibits the lowest overall force standard deviation, particularly at higher speeds. This suggests that lower stiffness allows for more flexibility and better absorption of force variations, maintaining stability even as speed increases.

- Controller 2 (Stiffness = 200 N/m) shows moderate force variations, aligning with its intermediate stiffness level. However, force standard deviation rises more steeply compared to Controller 3.
- Controller 1 (Stiffness = 300 N/m) presents the highest force standard deviation, especially at higher speeds, reflecting increased resistance and reduced control stability under high stiffness conditions.
- Submersion reduces force variability across all controllers, confirming the damping effect of the fluid medium. This effect is particularly pronounced for Controller 3, where the lowest stiffness combines with fluid damping to minimize force fluctuations.

To better understand the role of fluid, a distinction is made between the effects of the water column and the sandbed. The water column primarily acts as an external damping medium, absorbing sudden force spikes and reducing fluctuations caused by rapid movements. This is evident in the reduction of force standard deviation in submerged conditions across all controllers. Conversely, the sandbed influences force variability through its granular structure, where variations in compaction and sediment displacement contribute to force instability. In dry conditions, the lack of water resistance leads to increased force fluctuations due to uncontrolled sand displacement upon tool interaction.

Additionally, the performance of force standard deviation is analyzed relative to the mean force. The ratio of standard deviation to mean force provides insight into the consistency of force application. In dry conditions, this ratio is higher, indicating more erratic force variations. In submerged conditions, the ratio is lower, meaning force application remains more stable across the trajectory.

These results indicate that while higher stiffness can improve force stability in certain conditions, it also increases resistance during contact, leading to higher force standard deviations at increased speeds. Conversely, lower stiffness allows for greater flexibility and better absorption of force variations, resulting in lower force standard deviations, especially in submerged environments.

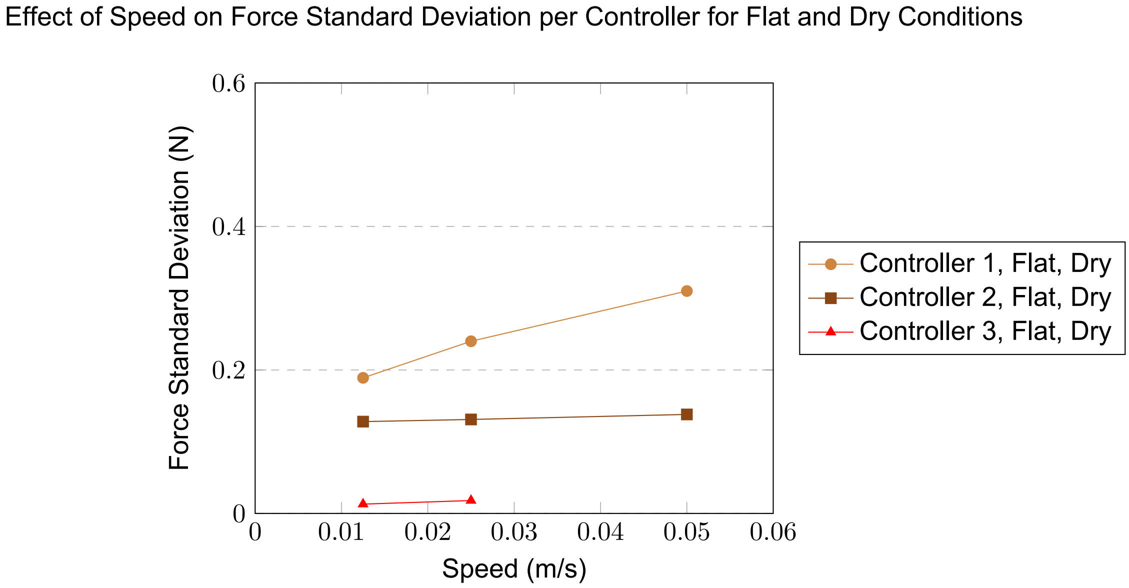


Figure 25: Graphical representation of Force Standard Deviation as a function of speed for different controllers and conditions.

Effect of Speed on Force Sandard Deviation per Controller for Curved and Dry conditions

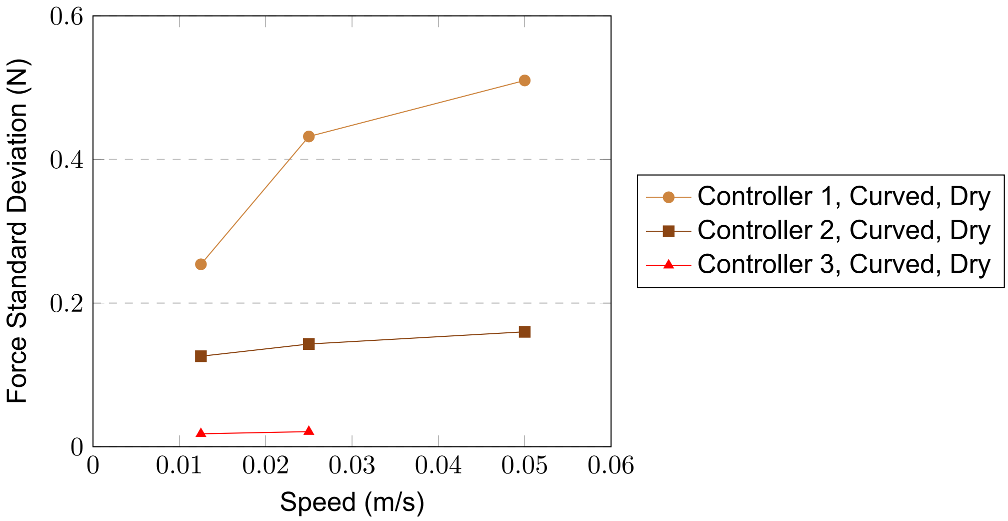


Figure 26: Graphical representation of Force Standard Deviation as a function of speed for different controllers and conditions.

Effect of Speed on Mean Force per Controller for Flat and Submerged Conditions

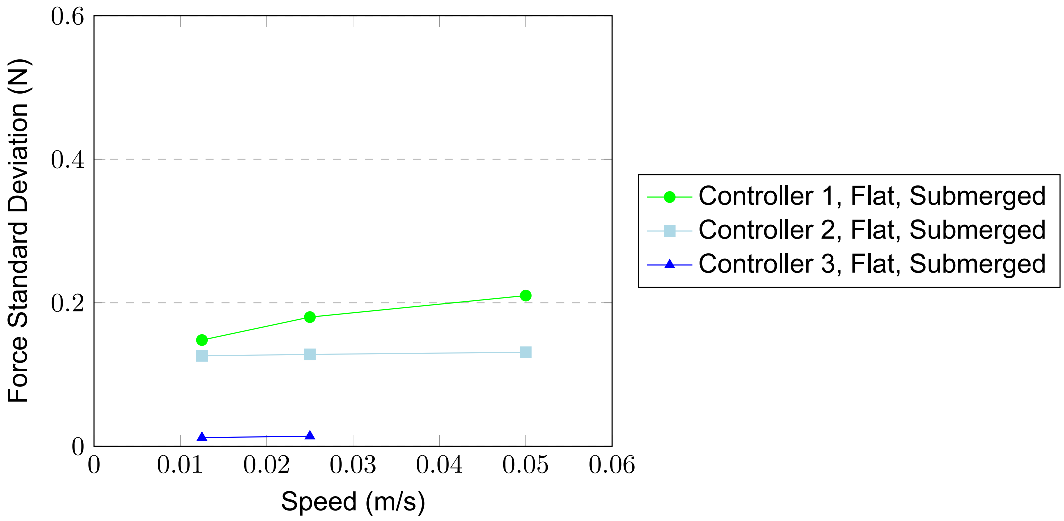


Figure 27: Graphical representation pf Force Standard Deviation as a function of speed for different controllers and conditions.

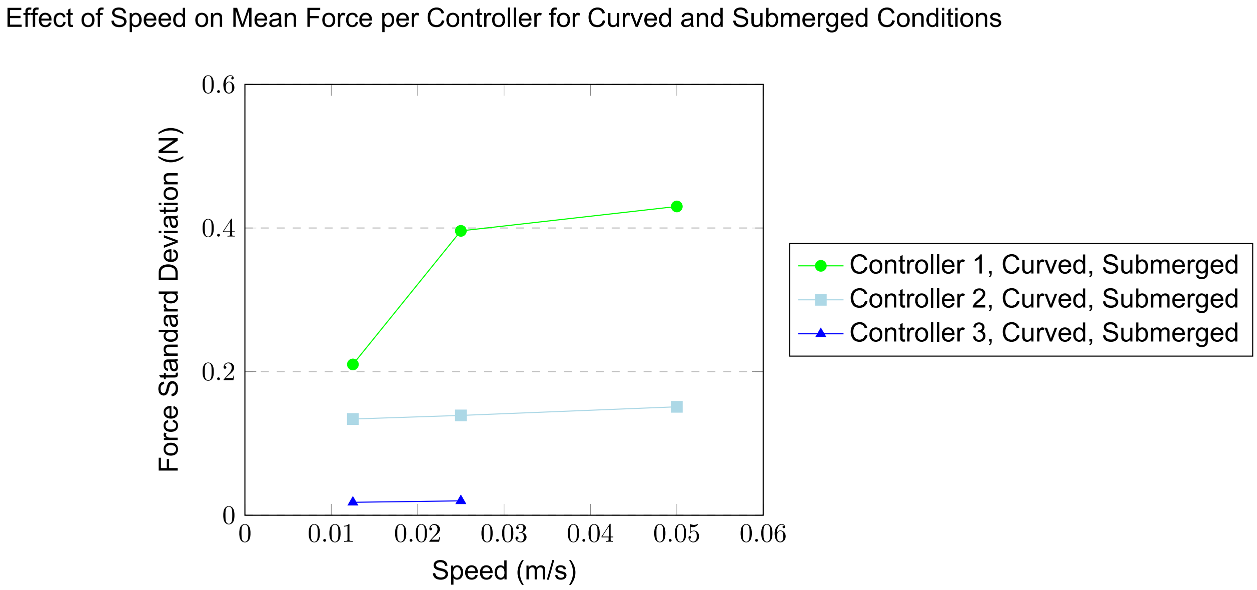


Figure 28: Graphical representation of Force standard Deviation as a function of speed for different controllers and conditions.

0.14.1. Effect of Surface on Penetration Error Variation

To assess the influence of surface topology and fluid conditions on penetration error variation, two distinct analyses were conducted: one comparing flat and curved surfaces and another examining dry versus submerged conditions.

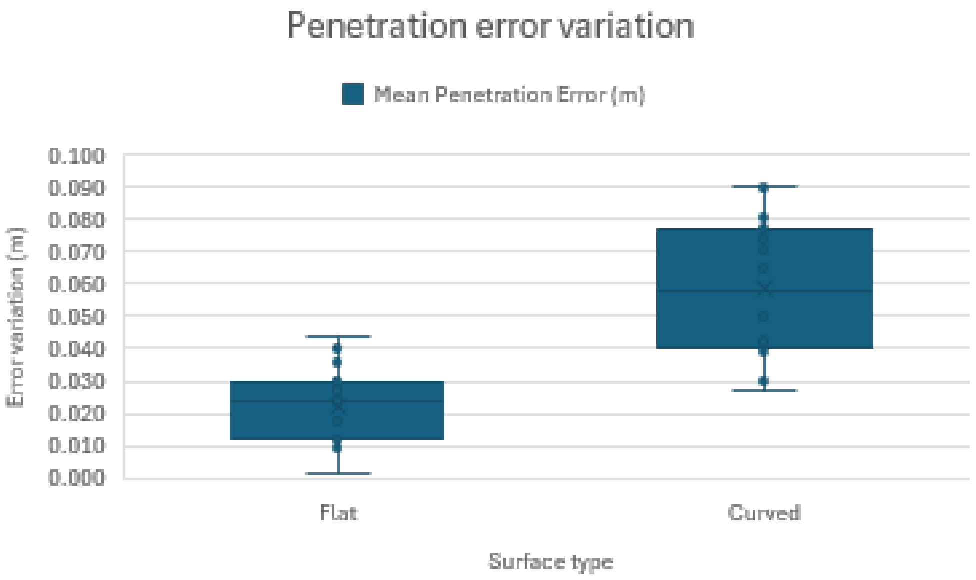


Figure 29: Penetration error variation as a function of surface topology (flat vs. curved).

As shown in Figure 29, penetration error exhibits greater variability on curved surfaces compared to flat ones. The increased deviation observed in curved surfaces suggests that maintaining trajectory accuracy is more challenging when the contact surface is uneven. This effect can be attributed to variations in force distribution across the surface, which affects the controller's ability to maintain a steady penetration depth.

Figure 30 demonstrates the impact of fluid conditions on penetration error variation. Submerged conditions result in reduced penetration error variability compared to dry conditions. The presence of fluid appears to dampen fluctuations, likely due to hydrodynamic resistance and increased stability in the interaction

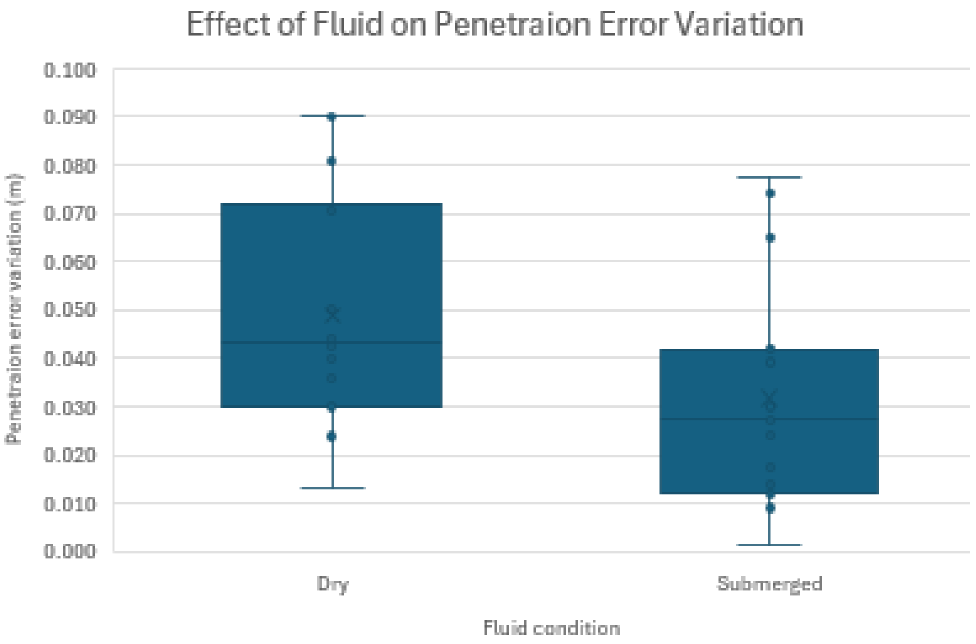


Figure 30: Effect of fluid conditions on penetration error variation (dry vs. submerged).

force. This reduction in variation suggests that underwater conditions inherently contribute to improved consistency in penetration depth, which is beneficial for robotic applications requiring precise substrate interaction.

Overall, these results highlight the trade-offs between different surface and environmental conditions. While submerged conditions mitigate penetration error variability, curved surfaces introduce greater challenges in trajectory adherence, necessitating optimized control strategies to compensate for these deviations.

0.15. Sources of Experimental Error

Understanding the potential sources of error in the experimental setup is crucial for refining impedance control strategies and improving the accuracy of future studies. Several key factors were identified that may have contributed to deviations in penetration error, force variability, and trajectory accuracy.

One significant factor was the effect of low stiffness on rod inclination. Due to the reduced transverse stiffness in lower stiffness controllers, the rod was more prone to bending, particularly at higher velocities. This caused the rod angle to deviate from the ideal vertical position normal to the tank bottom, influencing penetration depth accuracy and force application.

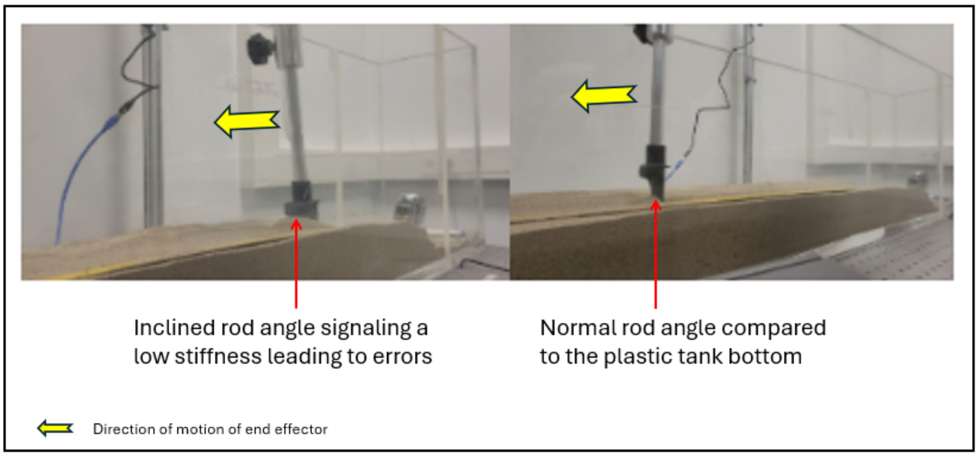

Inclined rod angle signaling a low stiffness leading to errors
Normal rod angle compared to the plastic tank bottom
Direction of motion of end effector

Figure 31: Rod inclination due to lower transverse stiffness left photo, Rod in desired straight up position right photo.

Additionally, the rod's geometry contributed to sand accumulation during movement. The shape of the rod, combined with its movement trajectory, resulted in significant sand buildup around the leading edge, which increased penetration resistance and may have influenced force readings. This accumulation effect was more pronounced at lower stiffness levels, where the rod experienced greater flexibility and deformation.

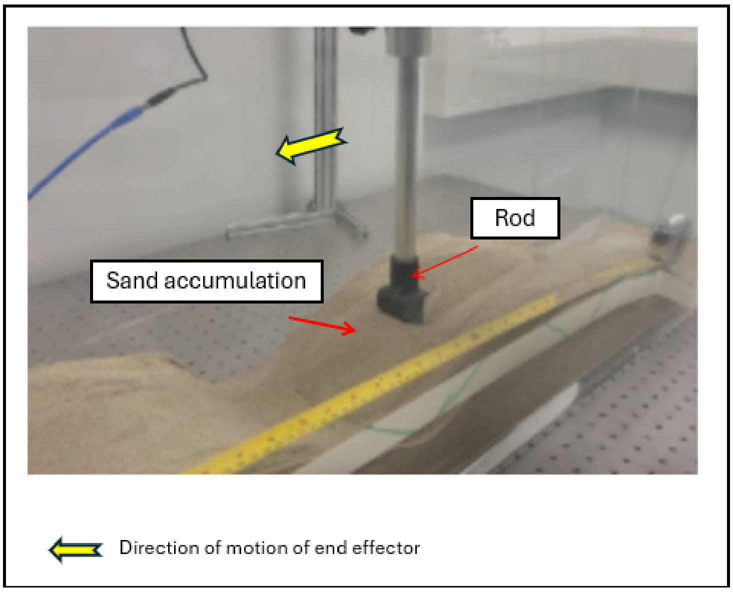


Figure 32: Sand accumulation on the rod during penetration.

Another critical source of error was water turbidity caused by sand displacement during the tests. As the rod moved through the sandbed, the disturbance created suspended particles in the water column. If not given sufficient time to settle between tests, these suspended particles could interfere with trajectory tracking and depth perception from the 3D camera, potentially introducing errors in recorded penetration depths and force measurements.

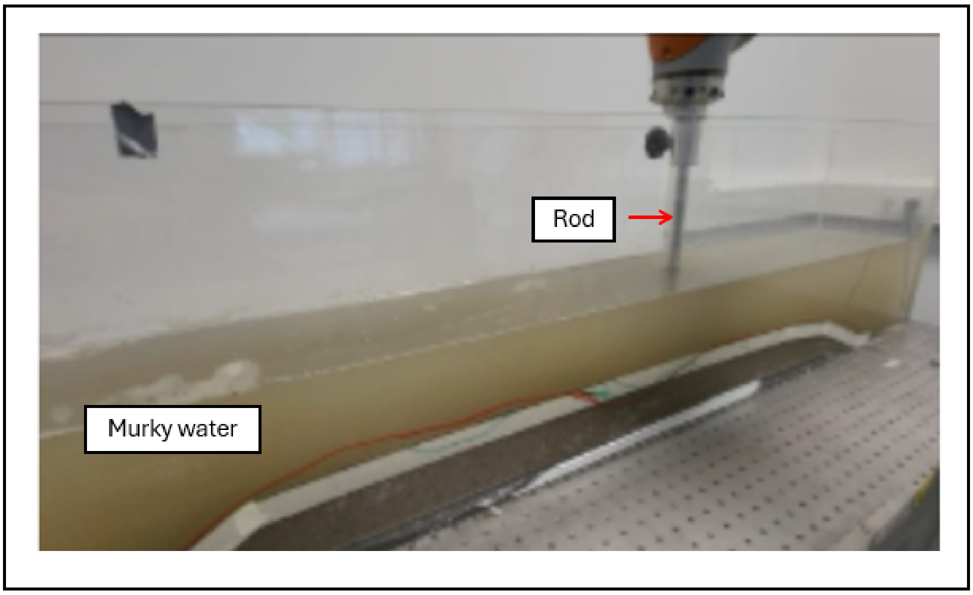


Figure 33: Example of poor water turbidity level affecting visibility and trajectory tracking.

One notable source of error observed in the experiments is the formation of a ripple pattern along the sandbed during robotic motion. This effect, as shown in Figure 34, occurs because the controller follows a discrete trajectory, moving in small incremental steps rather than a perfectly continuous motion.

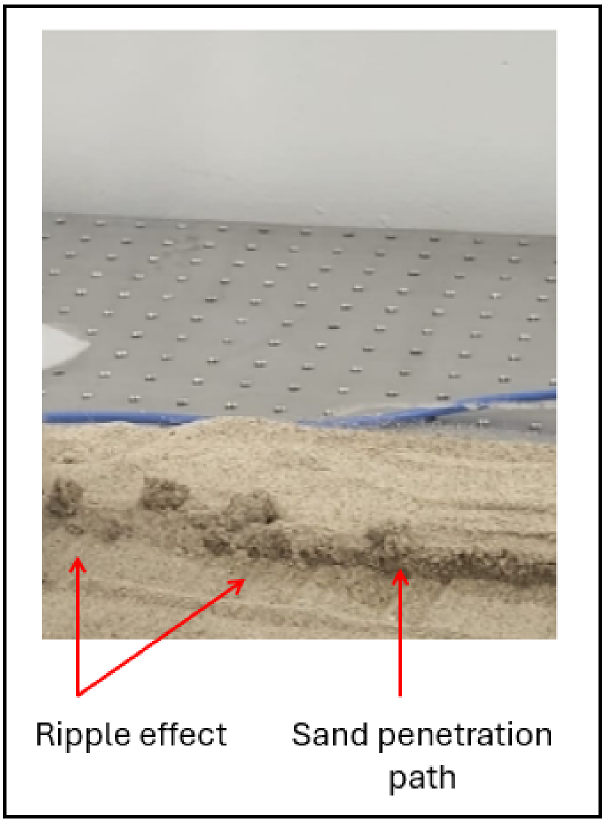


Figure 34: Ripple effect caused by discrete point-to-point trajectory execution.

Each step in the trajectory slightly displaces sand, creating small ridges and valleys along the movement path. Over time, these small accumulations result in a noticeable ripple pattern, particularly in dry conditions where there is no water to smooth out the disturbances. This effect may lead to deviations in penetration depth readings and increase variability in force measurements, as the rod encounters varying resistance across the rippled surface.

Additionally, the ripple effect may compound with other sources of error, such as sand accumulation on the rod and turbidity in submerged conditions, further influencing trajectory tracking accuracy. Future improvements could involve refining the control algorithm to generate smoother trajectories or introducing interpolation techniques to minimize abrupt changes in movement.

Beyond the identified environmental and mechanical error sources, inherent limitations within the KUKA robotic system also contributed to certain test failures and unexpected variations in motion execution. The robot's 7-degree-of-freedom (DOF) architecture introduces additional complexity in force distribution and trajectory tracking. Unlike the single-DOF actuators typically used in real-world applications such as deep-sea mining, the KUKA system's redundant degrees of freedom create uncertainties in how joint-level motions translate to end-effector behavior. This makes it difficult to precisely estimate the impact of joint movement variations on the rod's trajectory, particularly when interacting with deformable media like sand.

Additionally, safety features integrated into the KUKA system played a crucial role in limiting certain test conditions. The robot is equipped with joint angle tolerance limits and force threshold mechanisms that automatically halt motion if excessive forces are detected. This was particularly evident in low-stiffness controller tests, where the lack of rigidity led to greater rod deflection under force. At the highest tested speed of 0.05 m/s, all trials using Controller 3 (100 N/m) failed, as the robot's safety features triggered emergency stops when forces exceeded allowable limits. These failures highlight the constraints of using an industrial robotic arm with built-in safeguards in experiments designed to simulate applications where external force variations are expected and must be tolerated.

While these limitations ensured the protection of the robotic hardware, they also prevented the full exploration of high-speed behavior under low-stiffness conditions. Future studies could mitigate these constraints by utilizing robotic platforms specifically designed for controlled force applications, allowing for more accurate replication of impedance-controlled interactions in real-world environments.

These errors are important to identify as they play an important role in data error, and may lead to a better understanding of certain data patterns.

0.16. Summary of Findings

The experiments conducted in this study varied three key parameters: stiffness, speed, and environmental conditions. Three different stiffness settings (300 N/m, 200 N/m, and 100 N/m) were tested across three speeds (0.0125 m/s, 0.025 m/s, and 0.05 m/s) in four environmental conditions (dry flat, dry curved, submerged flat, and submerged curved). The results provided insights into the trade-offs between penetration accuracy, force stability, and adaptability.

- Higher stiffness reduces penetration error but does not necessarily ensure the best overall performance. While stiffer controllers followed the trajectory more accurately, they also exhibited higher force fluctuations, which could lead to instability when encountering obstacles.
- Submerged conditions reduce force variability and penetration error, improving trajectory tracking. The fluid damping effect stabilizes interactions, allowing for better depth control and minimizing force deviations compared to dry conditions.
- Higher speeds increase both penetration error and force variation across all conditions. This is particularly evident in dry environments, where higher speeds amplify force fluctuations due to increased inertia effects in granular media.
- Lower stiffness leads to lower force magnitudes and allows greater compliance. In this context, compliance refers to the ability of the system to absorb external disturbances, reducing abrupt force variations when encountering uneven terrain or obstacles. This trade-off, however, results in higher penetration error, particularly in curved surfaces.

- Curved, dry surfaces produce the highest penetration errors and force variations. The lack of fluid support and increased terrain irregularities create higher deviations in both penetration and force stability, making control more challenging.

Based on the combined evaluation of penetration accuracy and force stability, the best overall performance was observed with the 200 N/m stiffness setting in submerged conditions at moderate speeds (0.025 m/s). This configuration provided a balance between accuracy and compliance, minimizing penetration error while avoiding excessive force fluctuations. The findings suggest that an optimal setup should consider not only spatial accuracy but also stability in force application, especially in unstructured environments where external disturbances are common.

Discussion

The data collected provids valuable insights into the trade-offs associated with different impedance control settings in underwater environments, particularly in the context of robotic interactions with granular media such as sandbeds. The data reveal that while higher stiffness in an impedance controller generally leads to better spatial precision and reduced penetration errors, this is not necessarily the optimal configuration for all underwater applications. The main limitation arises from increased force fluctuations and instability when encountering unexpected terrain variations or obstacles, particularly in high-velocity conditions. Additionally, excessive stiffness can result in higher resistance upon impact with hard surfaces, which may disrupt the control system's stability in unstructured environments.

0.16.1. Penetration Error and Controller Stiffness

The data show that higher stiffness values correlate with lower penetration errors, with Controller 1 (300 N/m) exhibiting the lowest mean penetration error of 0.013 m, followed by Controller 2 (200 N/m) at 0.025 m, and Controller 3 (100 N/m) at 0.039 m. This suggests that stiffer controllers more closely follow the desired trajectory, reducing deviation. However, the benefits of higher stiffness are not uniform across all conditions. For instance, in submerged environments, the damping effect of the fluid medium significantly mitigates the impact of increasing velocity on penetration error. This is evident in the reduced penetration errors observed in submerged conditions compared to dry conditions, indicating that fluid resistance plays a stabilizing role in the robotic interaction.

The distribution of penetration errors suggests that errors are not normally distributed but skewed, particularly in lower stiffness controllers. More flexible controllers cause larger errors because compliance allows for greater deformation under applied forces, leading to instability in trajectory tracking. The fluctuations observed are not limited to the vertical direction but also include trajectory deviations due to unintended tilting of the end effector. In theory, penetration error should remain 2 cm vertically along the rod, which is expected to stay perpendicular to the bottom of the tank. However, in practice, at lower stiffness settings (Controller 3), the rod experienced noticeable bending, particularly at 0.05 m/s. This bending introduced additional deviations along the trajectory and, in extreme cases, led to failure, causing the robot to stop mid-test due to excessive force build-up and instability.

These findings emphasize the importance of selecting an appropriate balance between compliance and stiffness in impedance control design. While higher stiffness improves spatial accuracy, it may also amplify force instability, whereas lower stiffness improves adaptability but at the cost of increased penetration error and potential system failures at high velocities.

It is important to note that the system was tested at a fixed penetration depth. However, in practical applications, an impedance-controlled system should not only ensure contact but also dynamically limit excessive penetration. When operating above the bed, the system should allow for rapid repositioning back to the top of the bed. Conversely, as soon as penetration occurs, flexibility is needed to enable smooth movement out of or onto the bed. Implementing an adaptive stiffness strategy that adjusts based on penetration depth could improve operational efficiency in real-world conditions.

0.16.2. Effect of velocity on Penetration Error

The relationship between velocity and penetration error is particularly noteworthy. Across all controllers, an increase in velocity consistently results in greater penetration error, indicating reduced tracking accuracy at higher velocities. This trend is evident regardless of the controller's stiffness settings. At lower velocities, all controllers maintain relatively low penetration errors, with Controller 1 performing the best overall. However, as velocity increases, penetration errors become more pronounced, with Controller 3 experiencing the highest errors, particularly at the highest velocities. This highlights the challenges of high-velocity operation and the need for careful consideration of velocity in the design of impedance control strategies.

The distribution of penetration errors indicates that the errors are not normally distributed but skewed, particularly in lower stiffness controllers. The more flexible control causes larger errors because compliance leads to increased deformation under force, making it difficult to maintain a stable trajectory. These errors are not limited to the vertical direction but also include fluctuations along the trajectory path. Ideally, the penetration error should remain at 2 cm vertically along the rod, which is theoretically perpendicular to the bottom of the tank. However, in practice, at lower stiffness settings (Controller 3), the rod experienced noticeable bending, particularly at 0.05 m/s. This bending resulted in additional deviations along the trajectory and, in extreme cases, led to failure, causing the robot to stop mid-test due to excessive force build-up and loss of trajectory stability. This suggests that at low stiffness and high velocity, both translational and tilting errors contribute to increased penetration error, emphasizing the importance of maintaining an appropriate balance between compliance and stiffness in impedance control design.

.

0.16.3. Mean Force and Environmental Conditions

The variation of mean force under different velocity, surface, and environmental conditions further underscores the complexities of impedance control in underwater environments. Increasing velocity leads to a rise in mean force, suggesting that at higher velocities, the robot exerts greater interaction forces on the sandbed. This effect is more pronounced on curved surfaces, which generally lead to higher mean forces than flat surfaces, particularly in dry conditions. Submerged environments, however, result in lower mean forces compared to dry conditions, likely due to fluid damping effects reducing resistance.

Breaking down the results per controller, Controller 1 maintains relatively stable mean force levels across conditions, with a gradual increase at higher velocities, especially in curved, dry scenarios. Controller 2 shows greater sensitivity to velocity changes, with a sharper increase in mean force at higher velocities. Controller 3 exhibits the highest mean force values across all conditions, particularly in dry environments. This is mainly due to the deeper penetration depth of the rod in the sand, which creates more matter to go through and therefore requires more force.

Additionally, the increased effect of mean force on curved surfaces can be partially attributed to the complexity of the robotic system used in this experiment. The KUKA robot, with its 7 degrees of freedom, allows for highly adaptable movements, which differ from the simpler single-degree-of-freedom (1 DOF) mechanisms typically found in real-life applications such as seabed excavation and deep-sea mining. In an actual operational setting, a 1 DOF system would likely experience less variability in force application, as it would not have the same range of kinematic adjustments to compensate for changes in terrain. This means that while the increased forces on curves observed in this study are relevant, they may be amplified due to the flexibility of the robotic arm, rather than being solely a result of surface topology.

0.16.4. Force Standard Deviation and Stiffness

The relationship between speed and force standard deviation across different controllers provides additional insights. Force standard deviation increases with speed across all controllers, indicating that higher velocities introduce greater force fluctuations. Notably, Controller 3 (100 N/m) exhibits the lowest overall force standard deviation, particularly at higher speeds. This suggests that lower stiffness allows for more flexibility and better absorption of force variations, maintaining stability even as speed increases. In contrast, Controller 1 (300 N/m) presents the highest force standard deviation, especially at higher speeds, reflecting increased resistance and reduced control stability under high stiffness conditions.

Submersion reduces force variability across all controllers, confirming the damping effect of the fluid medium. This effect is particularly pronounced for Controller 3, where the lowest stiffness combines with fluid damping to minimize force fluctuations. These findings indicate that while higher stiffness can improve force stability in certain conditions, it also increases resistance during contact, leading to higher force standard deviations at increased speeds. Conversely, lower stiffness allows for greater flexibility and better absorption of force variations, resulting in lower force standard deviations, especially in submerged environments.

0.16.5. Damping Effect of Water in Submerged Conditions

The damping effect of water in submerged conditions plays a critical role in reducing penetration errors and force variability. Fluid resistance acts as a natural stabilizer, attenuating the impact of external disturbances and minimizing force fluctuations. This is evident in the data, where submerged conditions consistently show lower penetration errors and force standard deviations compared to dry conditions. The fluid medium's ability to absorb and dissipate energy helps maintain more stable interactions between the robotic system and the sandbed, even at higher velocities.

This damping effect is particularly beneficial in underwater environments, where the presence of water provides a consistent and predictable medium for robotic operations. The results indicate that variability in force measurements, which has been interpreted as noise in the results section, is significantly dampened in submerged conditions. This suggests that the water column acts as a filtering medium that smooths out force fluctuations, leading to more consistent robotic interactions. By leveraging this natural stabilizing effect, impedance controllers can be optimized to enhance accuracy and reliability in underwater applications, reducing the sensitivity to rapid force disturbances encountered in dry conditions.

0.16.6. Surface Topology and Fluid Conditions

The influence of surface topology and fluid conditions on penetration error variation is also significant. Penetration error exhibits greater variability on curved surfaces compared to flat ones, suggesting that maintaining trajectory accuracy is more challenging when the contact surface is uneven. This effect can be attributed to variations in force distribution across the surface, which affects the controller's ability to maintain a steady penetration depth. Submerged conditions result in reduced penetration error variability compared to dry conditions, as the presence of fluid dampens fluctuations, likely due to hydrodynamic resistance and increased stability in the interaction force.

In light of these findings, the damping effect of water in submerged conditions emerges as a crucial factor in optimizing impedance control strategies for underwater robotic applications. The ability of the fluid medium to stabilize interactions and reduce force variability suggests that lower stiffness controllers, which allow for greater flexibility and adaptability, may be particularly well-suited for underwater environments. This insight underscores the importance of considering environmental conditions when designing impedance control strategies for robotic systems operating in dynamic and unpredictable underwater settings.

The experimental findings reveal important trade-offs between impedance control stiffness, penetration accuracy, force variability, and environmental adaptability. While high-stiffness controllers consistently yield lower penetration depth deviations, this does not necessarily equate to optimal performance in real-world underwater conditions. The assumption that stiffer control leads to better accuracy must be reconsidered in the context of external disturbances, unforeseen obstacles, and operational constraints.

A high-stiffness controller exhibits superior spatial accuracy; however, it also presents increased vulnerability to external impacts. In an unpredictable seabed environment, where obstacles such as rocks or debris may be present, excessive stiffness can result in force spikes upon collision, potentially disrupting trajectory control or causing the robotic arm to halt movement abruptly. This could be detrimental to applications such as deep-sea mining (DSM), where maintaining continuous operation is critical.

Conversely, lower-stiffness controllers allow for greater compliance when interacting with deformable surfaces, mitigating the risk of system blockage in response to sudden external forces. However, the trade-off comes in the form of higher penetration depth deviations, particularly in dry conditions where environmental damping effects are absent.

The accuracy of the 3D camera and the ability of the KUKA robot to follow a predefined trajectory play a significant role in shaping these findings. The RealSense D455 camera provides depth perception for trajectory mapping, but its precision is affected by water clarity, lighting conditions, and the reflectivity of the sandbed. In submerged conditions, distortions due to light refraction and variations in water clarity may introduce small deviations in the planned trajectory, impacting the accuracy of the depth estimation. Similarly, the KUKA robot, with its 7DOF control system, is highly precise in trajectory execution but differs from real-world applications, where simpler 1DOF mechanisms are more common. These factors suggest that some of the observed variations in penetration depth and force fluctuations may partially stem from the combined limitations of the sensing and actuation systems, rather than being solely attributable to the impedance control settings.

0.16.7. The Influence of Speed on Performance

The relationship between velocity and impedance control effectiveness is another crucial factor observed in the experiments. As velocity increases, the accuracy of penetration depth decreases, and force variability becomes more pronounced. This is particularly evident at higher stiffness levels, where increased velocity exacerbates force fluctuations, making it difficult for the controller to maintain precise contact forces with the sandbed.

Despite this, the damping effects of fluid environments help mitigate force instabilities, reducing fluctuations in measured forces. This suggests that higher velocities may be more feasible in submerged conditions, as natural damping provides additional stabilization. However, due to the safety limitations of the KUKA robotic system, high-velocity testing was not possible in this study, restricting the analysis to lower operational velocities. Future experiments should assess how impedance control behaves in higher-velocity conditions to determine whether dynamic stiffness adjustments could optimize performance while maintaining stability.

A potential avenue for optimization is the implementation of a hybrid control strategy. In this context, hybrid control refers to a combination of impedance control with an adaptive feedback mechanism that adjusts stiffness dynamically based on sensed environmental conditions. The criteria for this dynamic adjustment could be derived from real-time force measurements, penetration depth deviations, and interaction stability. For instance, if excessive force fluctuations are detected, stiffness could be lowered to allow for better compliance, whereas in stable conditions, higher stiffness could be maintained to optimize accuracy. Implementing such a strategy would require integrating force sensors and environmental mapping techniques to adjust control parameters in real-time, ensuring an optimal balance between stability, compliance, and precision in varying seabed conditions.

0.16.8. Effect of Fluid Damping on Force Consistency

A significant observation in submerged conditions is the attenuation of force variability due to fluid damping. The presence of water naturally reduces oscillations in applied forces, suggesting that underwater environments may be better suited for more compliant (lower-stiffness) control strategies without significantly sacrificing precision. This is supported by the data, where submerged conditions consistently show lower force standard deviation compared to dry conditions.

One of the key contributors to this damping effect is the dilation phenomenon in granular media. When the robotic system interacts with a submerged sandbed, the grains undergo shearing and rearrangement, which temporarily increases pore volume and reduces internal resistance. This dilation effect leads to a localized drop in pore pressure, affecting the force transmission through the medium. As a result, the system experiences a form of passive force regulation, where the effective stiffness of the sandbed changes dynamically in response to penetration depth and applied forces. This effect is particularly relevant for impedance control, as it highlights the role of granular mechanics in modulating interaction forces.

In addition to dilation, the surrounding fluid also contributes to force attenuation through viscous damping. Water provides resistance to sudden force fluctuations by exerting drag on the moving components and filling the voids created by grain displacement. While this effect is secondary to the dilation mechanism in the given experimental conditions, it further stabilizes force interactions, particularly at lower speeds. Understanding the interplay between these damping mechanisms is essential for refining impedance control strategies in underwater environments, ensuring better adaptability and interaction stability in real-world applications.

0.16.9. Trade-offs Between Stiffness, Accuracy, and Adaptability

The observed trade-offs between stiffness, force control, and adaptability emphasize the need for a balanced approach to impedance control in underwater robotics. The following table summarizes the advantages and disadvantages of high- and low-stiffness configurations:

Table 1: Trade-offs Between Stiffness and Speed in Underwater Conditions

Factor	**Higher Stiffness**	**Lower Stiffness**
Penetration Accuracy	Higher accuracy	Lower accuracy
Adaptability to Obstacles	Risk of blockage if encountering rocks	More compliant, better adaptability
Force Stability	Higher force spikes	Smoother force transitions
Response to Fluid Damping	More stable due to damping	More stable due to damping
Suitability for Unknown Terrain	Less forgiving, high impact risks	Allows safer interactions
Performance at Higher Speeds	Less stable, increased force spikes	More stable, reduced force buildup

This analysis shows that a single, fixed stiffness configuration may not be optimal for real-world seabed interactions. Instead, a hybrid control approach that adjusts stiffness dynamically based on sensed environmental conditions may provide a superior balance between accuracy, stability, and adaptability.

0.17. Implications for Underwater Robotics

0.17.1. The Role of Adaptive Impedance Control

The findings of this study suggest that a pre-set high-stiffness control strategy is not always optimal for seabed robotic operations. In environments where unexpected obstacles exist, a dynamically adjustable impedance controller that can modulate stiffness in real-time based on force feedback and sensor inputs would significantly improve the robot's resilience to external disturbances.

For DSM applications, where maintaining consistent penetration depth is critical while avoiding excessive force spikes, modulating stiffness instead of using a fixed high-stiffness setting may improve both efficiency and safety. A robotic system that increases stiffness in stable conditions (e.g., uniform sand) but lowers stiffness upon detecting an obstacle (e.g., rock or debris) could ensure continuous operation without excessive disruptions.

0.17.2. Velocity Considerations for Practical Applications

In practical underwater operations, efficiency is a crucial factor. Lower velocities reduce force instability, making high-stiffness configurations more effective. However, in real-world applications where operational time is a constraint, slower velocities may not be viable. This highlights the need for an optimal balance between velocity and stiffness, where a moderate stiffness setting combined with an appropriate velocity threshold could provide stable yet efficient robotic performance.

Furthermore, since submerged conditions provide natural damping effects, the need for excessively high stiffness may be reduced, suggesting that a more compliant controller may be preferable in fluidic environments.

Additionally, the effect of velocity is significantly influenced by the geometry of the robotic end-effector used. The shape and orientation of the tool determine the magnitude of the damping force exerted by the fluid medium. In this study, the rod's 60-degree angle affected how water resistance interacted with the robotic movement. A different geometry, such as a broader or more streamlined shape, could lead to variations in damping force, affecting both force stability and penetration accuracy. This emphasizes that tool design should be considered alongside control parameters when optimizing robotic interactions in underwater environments.

0.18. Experimental Setup Considerations

The experimental setup relied on a KUKA robotic arm, which, while highly adaptable, introduces additional complexity that may not be necessary for simplified impedance control testing. A setup allowing for 1DOF operation, such as a manipulator mounted on a moving carriage, would reduce system variability and better reflect real-world applications. While the KUKA arm provides precise control, a simplified setup would allow for more direct assessments of impedance control parameters without additional kinematic adjustments influencing force and penetration behaviors.

0.19. Suggestions for Improvement

Several improvements could enhance the experimental methodology and broader applicability of the findings:

- Adaptive Impedance Control: Implementing a real-time adaptive stiffness controller that adjusts based on force feedback and environmental conditions could improve both accuracy and stability.
- Extended Speed Ranges: Exploring lower and higher speed ranges could provide a more comprehensive understanding of the speed-stiffness trade-off.
- Material Variability: Future studies should investigate different granular compositions, including varying particle sizes and moisture levels, to assess the broader applicability of impedance control in non-uniform environments.
- Rod shape adaptation to prevent sand accumulation as this is not an important variable for the study.
- Strict horizontal stiffening of the rod to prevent bending and therefore skewed errors.
- Improved trajectory planning tool by setting reliable intrinsics in the 3D camera.

These improvements would contribute to more robust robotic systems capable of interacting effectively with complex, deformable environments.

Conclusion

This study set out to answer the question: How does fine-tuning joint stiffness in an impedance controller influence penetration accuracy and force stability in sandbeds under different environmental conditions?

The results demonstrate that while higher stiffness controllers minimize penetration errors, they also introduce higher force variations, particularly in dry environments. Conversely, lower stiffness controllers exhibit higher penetration errors but provide greater compliance, reducing the risk of trajectory deviations in uneven terrain. The optimal stiffness setting is not a fixed value but rather depends on the operational constraints, such as seabed consistency, velocity requirements, and obstacle presence.

0.20. Summary of Findings

This study investigated the effects of stiffness, speed, and environmental conditions on penetration accuracy and force stability in robotic interactions with granular media. The findings indicate that higher stiffness improves penetration accuracy and force stability, whereas increased speed leads to greater force variability and reduced accuracy. Submerged conditions were found to attenuate penetration variability, largely due to the damping effects of the surrounding fluid, which help stabilize robotic interactions. Additionally, the failure of Controller 3 at the highest speed highlights the limitations of low-stiffness impedance control in high-speed operations, reinforcing the need for an optimal balance between compliance and rigidity.

0.21. Contributions

This research contributes to the advancement of robotic impedance control by systematically analyzing the trade-offs between stiffness, speed, and environmental factors in deformable terrain interactions. The findings provide critical insights for the design and optimization of robotic systems operating in granular, fluidic, and soft environments, with implications for applications such as planetary exploration, excavation robotics, deep-sea mining, and minimally invasive robotic surgery. Specifically, this study:

- Demonstrates the quantitative relationship between stiffness and penetration accuracy across different speeds and environmental conditions.
- Identifies the trade-offs between accuracy and force stability in impedance-controlled robotic systems.
- Highlights the stabilizing effects of submerged environments, which mitigate force variability and improve consistency in robotic operations.
- Provides empirical evidence of failure points in low-stiffness controllers operating at high speeds, reinforcing the need for adaptive control strategies.

These contributions serve as a foundation for future advancements in impedance-controlled robotics, particularly for dynamic and unstructured environments where material consistency and external force interactions introduce additional complexity.

0.22. Practical Implications

For underwater robotic applications such as deep-sea mining and subsea infrastructure maintenance, these findings highlight the need for a dynamically adaptive impedance control system rather than a single fixed-stiffness configuration. Given that submerged environments inherently dampen force fluctuations, controllers with moderate stiffness settings may achieve sufficient penetration accuracy while maintaining adaptability in unstructured terrains.

Additionally, this study revealed that at low stiffness settings and high velocities, the robotic system may experience failures due to excessive force accumulation. This suggests that operational velocity constraints must be carefully calibrated based on stiffness selection to prevent system instability.

0.23. Future Work

Future research should focus on several areas to further develop the applicability of impedance control in underwater robotics:

- **Dynamic Impedance Control Strategies:** Investigate real-time stiffness and damping adjustments based on force feedback, integrating feedback loop models to optimize control responses dynamically.
- **High-Speed Testing:** Extend experiments to higher operational speeds, particularly in submerged conditions, to evaluate how impedance control settings behave under rapid movement constraints.
- **Testing on Heterogeneous Seabeds:** Implement stratified seabeds with variable sediment compositions, embedded obstacles, and varying compaction levels to better simulate real-world seabed interactions.
- **Single-DOF Actuator Studies:** Conduct experiments using simplified robotic systems with single-DOF actuators, which are more representative of actual DSM machinery, ensuring that the findings translate effectively to industry applications.
- **Integration with Advanced Sensing Technologies:** Enhance real-time environmental awareness using advanced sonar, LiDAR, or other methods. **Millimetric Penetration Depth Analysis:** Explore the precise control of penetration depth at a millimetric level using single-DOF actuators in underwater robotics tailored for deep-sea mining. This study would focus on the precise adjustments required to manage tool entry into varying sediment types, which is crucial for optimizing extraction processes and minimizing environmental impact. Developing specialized robotic platforms with enhanced sensing and control capabilities could provide the necessary precision and adaptability for these operations.

  These advancements would provide a more comprehensive understanding of how impedance control can be optimized for subsea robotics, leading to safer, more efficient, and more adaptable underwater robotic systems.

Bibliography

[1] N. Hogan, ”Impedance Control: An Approach to Manipulation: Part I - Theory,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 1-24, 1985.

[2] A. Albu-Schäffer and G. Hirzinger, ”Cartesian Impedance Control Techniques for Torque Controlled Lightweight Robots,” *Proceedings of ICRA*, pp. 657-663, 2003.

[3] N. Hogan and S. Buerger, ”Impedance Control: Stability and Performance,” *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1048-1057, 2008.

[4] I. D. Walker, ”Underwater Robots,” *IEEE Robotics & Automation Magazine*, vol. 12, no. 3, pp. 52-63, 2005.

[5] C. Ferrari and A. Schiele, ”Impedance Control for Robotic Systems in Contact with Compliant Environments,” *International Journal of Robotics Research*, vol. 22, no. 8, pp. 635-652, 2003.

[6] R. D. Howe and M. R. Cutkosky, ”Dynamic tactile sensing: Perception of fine surface features with stress rate sensing,” *IEEE Transactions on Robotics and Automation*, vol. 9, no. 2, pp. 156-168, 1993.

[7] M. B. Allen, ”Granular Materials: A Basic Introduction to Their Mechanical Behavior,” *Journal of Geotechnical Engineering*, vol. 120, no. 4, pp. 706-727, 1994.

[8] A. C. Santomaso, ”Granular Materials: Mechanical Properties and Modeling,” *Granular Matter*, vol. 6, no. 1, pp. 1-22, 2004.

[9] M. Boerner and D. Krejci, ”Robotic Systems for Soil and Sand Penetration: Mechanics and Control,” *Journal of Robotics Research*, vol. 26, pp. 543-559, 2007.

[10] E. Mochida, ”Sand Penetration Mechanics and Control Strategies for Robotic Systems,” *Journal of Terramechanics*, vol. 49, no. 5, pp. 297-308, 2013.

[11] F. Pomerleau, F. Colas, and R. Siegwart, ”Point Cloud Data for Navigation in Complex Environments,” *Proceedings of IROS*, pp. 245-250, 2013.

[12] H. Nguyen, ”Point Cloud Processing for Robotic Navigation,” *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 897-908, 2013.

[13] R. B. Rusu and S. Cousins, ”3D is Here: Point Cloud Library (PCL),” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1-4, 2011.

[14] M. Elbanhawi and R. Simmons, ”Sampling-Based Motion Planning for Autonomous Robots,” *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 652-666, 2015.

[15] J. Ren, ”Deep Learning for Robotic Perception and Control,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1879-1894, 2019.

[16] Y. Lin, ”Underwater Mapping and Navigation Using Point Clouds,” *IEEE Journal of Oceanic Engineering*, vol. 45, no. 2, pp. 456-467, 2020.

Appendices

This appendix contains the Python script used for controlling the KUKA robot, recording force data, and executing the trajectory.

0.24. Full Robot Controller Python Script

```
import rospy
import actionlib
import numpy as np
from datetime import datetime
from cor_tud_msgs.msg import ControllerAction, ControllerGoal
from sensor_msgs.msg import JointState
from roboticstoolbox.robot.ERobot import ERobot

# Initialize the robot model
kuka = ERobot.URDF("/home/josepht/catkin_ws/iiwa_ros/src/iiwa_description/urdf/iiwa7.urdf.xacro")

# Global variables for force recording
recorded_data = []
current_positions = None
current_torques = None
start_time = None
last_recorded_time = None  # For time-based recording

# Callback for joint states
def joint_states_callback(msg):
    global current_positions, current_torques
    current_positions = np.array(msg.position)
    current_torques = np.array(msg.effort).reshape(-1, 1)  # Reshape torques to column vector

# Record data including test time, Cartesian position, and forces in X and Y directions at the tip of the rod
def record_data():
    global recorded_data, current_positions, current_torques, start_time, last_recorded_time
    if current_positions is not None and current_torques is not None:
        current_time = rospy.get_time()

        # Only record if 2 seconds have passed since the last recorded time
        if last_recorded_time is None or (current_time - last_recorded_time) >= 2.0:
            last_recorded_time = current_time

            # Compute Jacobian and force
            J = np.array(kuka.jacob0(current_positions))
            J_transpose_pinv = np.linalg.pinv(J.T)
            end_effector_force_vector = np.dot(J_transpose_pinv, current_torques).flatten()

            # Compute Cartesian position of the end effector
            cartesian_position = kuka.fkine(current_positions).t.flatten()

            # Extract forces in x and y directions and round to nearest integer
            force_in_x_direction_N = round(end_effector_force_vector[0])
            force_in_y_direction_N = round(end_effector_force_vector[1])

            # Record data with end effector's Cartesian z-coordinate
            recorded_data.append([
                round(current_time - start_time, 2),  # Relative time
                round(cartesian_position[0], 2),     # X
                round(cartesian_position[1], 2),     # Y
                round(cartesian_position[2], 2),     # Z (unchanged end effector position)
                force_in_x_direction_N,
                force_in_y_direction_N
            ])

if __name__ == "__main__":
    try:
        rospy.init_node('trajectory_with_force_recording')

        # Set up joint states subscriber
        joint_states_sub = rospy.Subscriber('/iiwa7/joint_states', JointState, joint_states_callback)
        rospy.wait_for_message('/iiwa7/joint_states', JointState)

        # Define action client
        ns = rospy.get_param('/namespaces', default='robot')
        client = actionlib.SimpleActionClient(ns + '/torque_controller', ControllerAction)
        client.wait_for_server()

        # Initial joint position setup
        goal = ControllerGoal()
        goal.mode = 'joint_ds'
        goal.time = 4
        goal.rate = 200
        goal.reference = np.zeros(7)
        goal.stiffness = 0.7 * np.array([100.0, 100.0, 50.0, 50.0, 25.0, 25.0, 10.0])
        goal.damping = 2 * np.sqrt(goal.stiffness)

        print("Press a key to go to initial joint position")
        input()
        client.send_goal(goal)
        client.wait_for_result()

        # Waypoints for intermediate joint positions
        current_position = np.zeros(7)
        waypoints = [
            (0, np.radians(-47)),
            (1, np.radians(42)),
            (3, np.radians(-68)),
            (5, np.radians(65))
        ]

        print("Press a key to move through intermediate joint waypoints")
        input()

        for joint_index, angle in waypoints:
            current_position[joint_index] = angle
            goal = ControllerGoal()
            goal.mode = 'joint_ds'
            goal.time = 3
            goal.rate = 200
            goal.reference = current_position
            goal.stiffness = 0.7 * np.array([100.0, 100.0, 50.0, 50.0, 25.0, 25.0, 10.0])
            goal.damping = 2 * np.sqrt(goal.stiffness)
            client.send_goal(goal)
            client.wait_for_result()

        print("Intermediate joint waypoints completed. Press a key to move to final Cartesian position.")
        input()

        # Final target Cartesian position
        goal = ControllerGoal()
        goal.mode = 'ee_cartesian_ds'
        goal.time = 4
        goal.rate = 200
        goal.reference = np.array([0.48, -0.5, 0.45, -np.pi, 0, -np.pi])
        goal.stiffness = np.array([300.0, 300.0, 300.0, 10.0, 10.0, 10.0])
        goal.damping = 2 * np.sqrt(goal.stiffness)
        client.send_goal(goal)
        client.wait_for_result()

        print("Final Cartesian position reached!")

        # City-like path
        print("Press a key to start city-like path")
        input()
        start_time = rospy.get_time()  # Start recording relative time

        # Generated trajectory
        path_segments = [
            [-0.500, 0.487],
            [-0.495, 0.476],
        ]

        steps_per_segment = 10
        target_velocity = 0.025

        for i in range(len(path_segments) - 1):
            start_y, start_z = path_segments[i]
            end_y, end_z = path_segments[i + 1]
            distance = np.linalg.norm(np.array([end_y, end_z]) - np.array([start_y, start_z]))
            time_per_segment = distance / target_velocity

            for step in range(steps_per_segment):
                progress = step / float(steps_per_segment)
                current_y = start_y + (end_y - start_y) * progress
                current_z = start_z + (end_z - start_z) * progress

                goal.mode = 'ee_cartesian_ds'
                goal.time = time_per_segment / steps_per_segment
                goal.reference = np.array([0.48, current_y, current_z, -np.pi, 0, -np.pi])
                goal.stiffness = np.array([300.0, 300.0, 300.0, 40.0, 40.0, 40.0])
                goal.damping = 2 * np.sqrt(goal.stiffness)
                client.send_goal(goal)
                client.wait_for_result()
                record_data()

        print("City-like path completed!")

        # Save recorded data to file
        timestamp = datetime.now().strftime('%Y-%m-%d_%H-%M-%S')
        file_name = f'recorded_data_{timestamp}.py'
        with open(file_name, 'w') as f:
            f.write("# Recorded trajectory data\n")
            f.write("# Time\tX\tY\tZ\tForce_X\tForce_Y\n")
            for entry in recorded_data:
                f.write(f"{entry[0]}\t{entry[1]}\t{entry[2]}\t{entry[3]}\t{entry[4]}\t{entry[5]}\n")

        print(f"Data recorded and saved to {file_name}")

        # Return home
        print("Press a key to return home")
        input()
        goal = ControllerGoal()
        goal.mode = 'joint_ds'
        goal.time = 4
        goal.rate = 200
        goal.reference = np.zeros(7)
        goal.stiffness = 0.7 * np.array([100.0, 100.0, 50.0, 50.0, 10.0, 10.0, 10.0])
        goal.damping = 2 * np.sqrt(goal.stiffness)
        client.send_goal(goal)
        client.wait_for_result()

        print("Done, pizza!")

    except rospy.ROSInterruptException:
        pass
```

Listing 1: Full Robot Controller Python Script

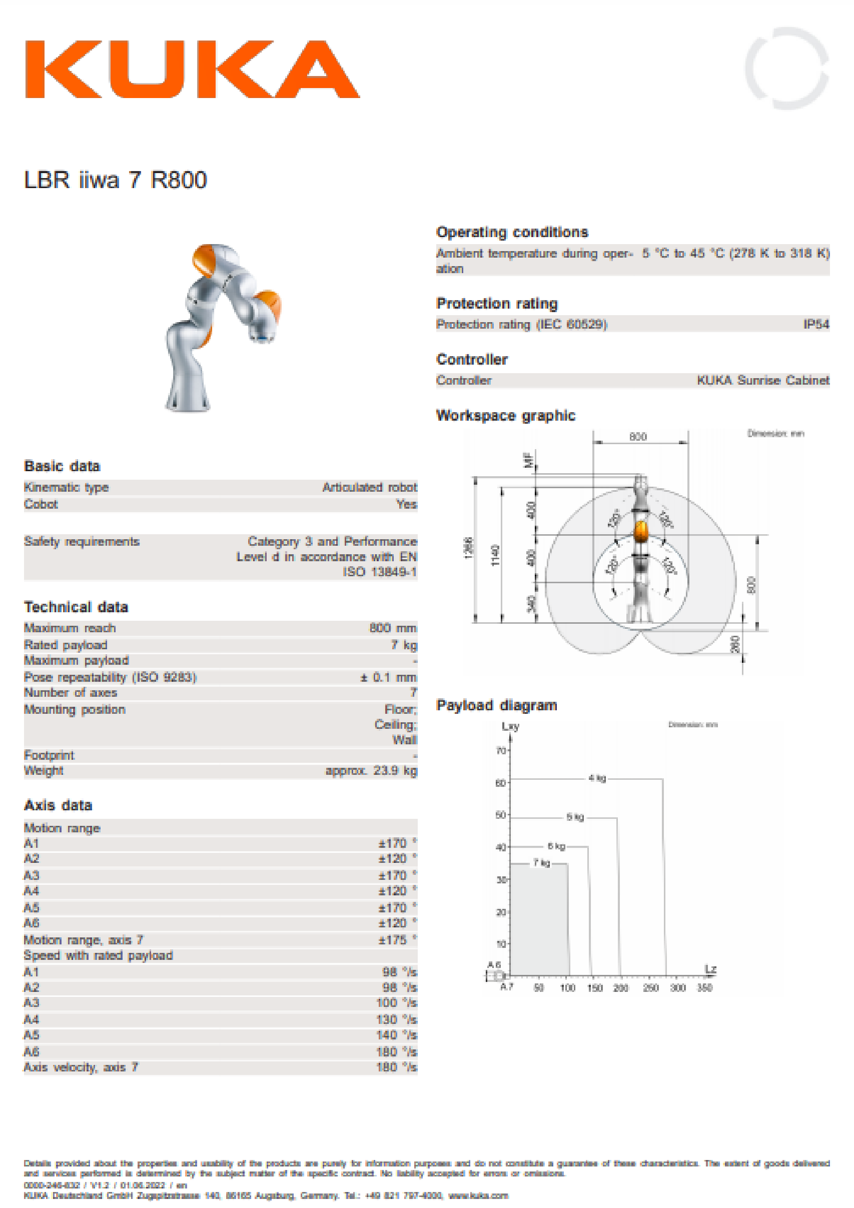
0.25. Full Trajectory Defining Script

```
import pyrealsense2 as rs
import numpy as np
import matplotlib.pyplot as plt
import cv2
import os

# Initialize RealSense pipeline
pipeline = rs.pipeline()
config = rs.config()
config.enable_stream(rs.stream.depth, 640, 480, rs.format.z16, 30)
config.enable_stream(rs.stream.color, 640, 480, rs.format.bgr8, 30)

# Start streaming
pipeline.start(config)

# Align RGB and depth frames
align_to = rs.stream.color
align = rs.align(align_to)

# Global variables
calibration_points = []  # Green points for scaling factor
measurement_points = []  # Red points for depth profile
scaling_factor = None  # Meters per pixel

# Mouse callback function
def select_point(event, x, y, flags, param):
    global calibration_points, measurement_points, scaling_factor

    if event == cv2.EVENT_LBUTTONDOWN:
        # Step 1: Collect calibration points (Green)
        if scaling_factor is None:
            if len(calibration_points) < 2:
                calibration_points.append((x, y))
                print(f"Calibration Point {len(calibration_points)} selected at ({x}, {y})")

            if len(calibration_points) == 2:
                calculate_scaling_factor()
                print("Calibration complete. Now select two red points for the graph.")

        # Step 2: Collect red points after calibration
        elif len(measurement_points) < 2:
            measurement_points.append((x, y))
            print(f"Measurement Point {len(measurement_points)} selected at ({x}, {y})")

            if len(measurement_points) == 2:
                print("Two points selected. Calculating depth profile...")
                plot_depth_profile()

# Function to calculate the scaling factor
def calculate_scaling_factor():
    global scaling_factor, calibration_points

    x1, y1 = calibration_points[0]
    x2, y2 = calibration_points[1]

    # Pixel distance between calibration points
    pixel_distance = np.hypot(x2 - x1, y2 - y1)
    known_distance = 1.0  # 1 meter

    # Scaling factor in meters per pixel
    scaling_factor = known_distance / pixel_distance
    print(f"Scaling factor: {scaling_factor:.6f} meters/pixel")

# Function to plot the depth profile and export coordinates
def plot_depth_profile():
    global measurement_points, scaling_factor, depth_frame

    x1, y1 = measurement_points[0]
    x2, y2 = measurement_points[1]

    # Interpolate points along the line
    num_points = int(np.hypot(x2 - x1, y2 - y1))
    x_values = np.linspace(x1, x2, num_points).astype(int)
    y_values = np.linspace(y1, y2, num_points).astype(int)

    # Collect depths and scaled distances
    depths = []
    distances = [0.0]  # Start at 0 for the first point

    for i in range(1, len(x_values)):
        depth = depth_frame.get_distance(x_values[i], y_values[i])
        if 0.1 < depth < 10:  # Exclude invalid depths
            depths.append(depth)
        else:
            # Replace invalid depth values with the last valid value or 0
            depths.append(depths[-1] if depths else 0)
        pixel_distance = np.hypot(x_values[i] - x_values[i-1], y_values[i] - y_values[i-1])
        real_distance = pixel_distance * scaling_factor
        distances.append(distances[-1] + real_distance)

    # Handle outliers: Replace values that deviate too much from neighbors
    cleaned_depths = replace_outliers(depths, threshold=0.1)

    # Smooth depths using weighted moving average
    smoothed_depths = weighted_moving_average(cleaned_depths, window_size=7)

    # Invert depths for heights
    max_depth = max(smoothed_depths)
    inverted_heights = [max_depth - d for d in smoothed_depths]
    distances = distances[:len(inverted_heights)]

    # Export as city-like path
    export_path_script(distances, inverted_heights)

    # Plot the inverted depth profile
    plt.figure()
    plt.plot(distances, inverted_heights, label="Height Profile", color="blue")
    plt.xlabel("Distance Along Line (meters)")
    plt.ylabel("Height (meters)")
    plt.title("Height Profile Along Selected Line")
    plt.legend()
    plt.show()
    measurement_points.clear()

# Function to replace outliers in depth values
def replace_outliers(data, threshold=0.1):
    """Replace values that deviate too much from their neighbors."""
    cleaned_data = data[:]
    for i in range(1, len(data) - 1):
```

```python
        if abs(data[i] - data[i-1]) > threshold and abs(data[i] - data[i+1]) > threshold:
            # Replace outlier with the average of neighbors
            cleaned_data[i] = (data[i-1] + data[i+1]) / 2
    return cleaned_data

# Function to apply weighted moving average
def weighted_moving_average(data, window_size=7):
    """Smooth data using a weighted moving average."""
    half_window = window_size // 2
    smoothed_data = []

    for i in range(len(data)):
        # Define the window range
        start = max(0, i - half_window)
        end = min(len(data), i + half_window + 1)

        # Apply weights (linear weights for smoother transitions)
        weights = np.arange(1, end - start + 1)
        weighted_sum = np.dot(data[start:end], weights)
        smoothed_value = weighted_sum / weights.sum()

        smoothed_data.append(smoothed_value)

    return smoothed_data

# Function to export the path script to the 'realsens' folder
def export_path_script(distances, heights):
    save_folder = "realsens"
    os.makedirs(save_folder, exist_ok=True)

    # Shift the starting point to x = -0.5
    x_shift = -0.5 - distances[0]
    shifted_distances = [d + x_shift for d in distances]

    # Add 0.429m + 0.1m to each height coordinate (1mm penetration depth)
    adjusted_heights = [h + 0.437 for h in heights]

    # Add an extra 0.04m to the first 3 points
    for i in range(min(3, len(adjusted_heights))):
        adjusted_heights[i] += 0.05

    # Generate points for every 0.005 meters (half a centimeter) along the x-axis
    interpolated_distances = np.arange(shifted_distances[0], shifted_distances[-1] + 0.005, 0.005)
    interpolated_heights = np.interp(interpolated_distances, shifted_distances, adjusted_heights)

    # Create the path script
    file_path = os.path.join(save_folder, "city_path.py")
    script_content = "path_segments = [\n"
    for d, h in zip(interpolated_distances, interpolated_heights):
        script_content += f"    [{d:.3f}, {h:.3f}],\n"
    script_content += "]\n"

    # Save the script to a file
    with open(file_path, "w") as file:
        file.write("# Generated City-Like Path Script\n")
        file.write(script_content)

    print(f"Path script saved as '{file_path}'.")

# OpenCV window setup
cv2.namedWindow("RGB Image")
cv2.namedWindow("Depth Image")
cv2.setMouseCallback("RGB Image", select_point)

try:
    while True:
        frames = pipeline.wait_for_frames()
        aligned_frames = align.process(frames)
        depth_frame = aligned_frames.get_depth_frame()
        color_frame = aligned_frames.get_color_frame()

        if not depth_frame or not color_frame:
            continue

        depth_image = np.asanyarray(depth_frame.get_data())
        color_image = np.asanyarray(color_frame.get_data())
        depth_colormap = cv2.applyColorMap(cv2.convertScaleAbs(depth_image,
            alpha=0.03), cv2.COLORMAP_JET)

        for point in calibration_points:
            cv2.circle(color_image, point, 5, (0, 255, 0), -1)
            cv2.circle(depth_colormap, point, 5, (0, 255, 0), -1)

        for point in measurement_points:
            cv2.circle(color_image, point, 5, (0, 0, 255), -1)
            cv2.circle(depth_colormap, point, 5, (0, 0, 255), -1)

        cv2.imshow("RGB Image", color_image)
        cv2.imshow("Depth Image", depth_colormap)

        if cv2.waitKey(1) & 0xFF == ord('q'):
            break

finally:
    pipeline.stop()
    cv2.destroyAllWindows()
```

Listing 2: Full Trajectory Defining Script

0.26. System Specifications



KUKA

LBR iiwa 7 R800

Basic data

Kinematic type	Articulated robot
Cobot	Yes
Safety requirements	Category 3 and Performance Level d in accordance with EN ISO 13849-1

Technical data

Maximum reach	800 mm
Rated payload	7 kg
Maximum payload	-
Pose repeatability (ISO 9283)	± 0.1 mm
Number of axes	7

Mounting position	Floor;
	Ceiling;
	Wall

Footprint	-
Weight	approx. 23.9 kg

Axis data

Motion range	
A1	±170 °
A2	±120 °
A3	±170 °
A4	±120 °
A5	±170 °
A6	±120 °
Motion range, axis 7	±175 °
Speed with rated payload	
A1	98 °/s
A2	98 °/s
A3	100 °/s
A4	130 °/s
A5	140 °/s
A6	180 °/s
Axis velocity, axis 7	180 °/s

Operating conditions

Ambient temperature during operation	5 °C to 45 °C (278 K to 318 K)

Protection rating

Protection rating (IEC 60529)	IP54

Controller

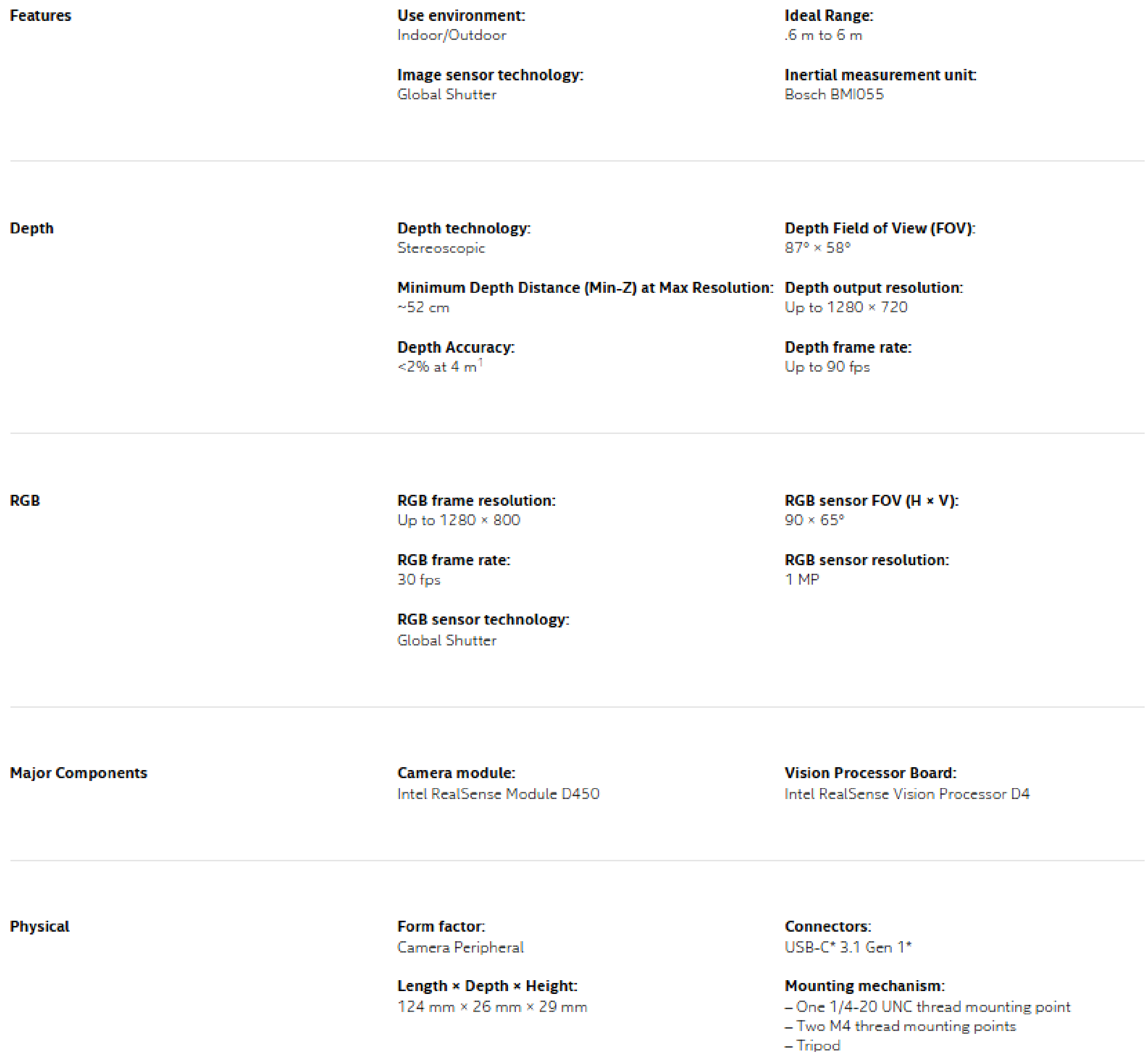
Controller	KUKA Sunrise Cabinet

Workspace graphic

Payload diagram

Details provided about the properties and usability of the products are purely for information purposes and do not constitute a guarantee of these characteristics. The extent of goods delivered and services performed is determined by the subject matter of the specific contract. No liability accepted for errors or omissions.
0000-246-832 / V1.2 / 01.06.2022 / en
KUKA Deutschland GmbH Zugspitzstrasse 140, 86165 Augsburg, Germany. Tel.: +49 821 797-4000, www.kuka.com

Figure 35: KUKA robot technical sheet



Features	**Use environment:** Indoor/Outdoor	**Ideal Range:** .6 m to 6 m
	Image sensor technology: Global Shutter	**Inertial measurement unit:** Bosch BMI055
Depth	**Depth technology:** Stereoscopic	**Depth Field of View (FOV):** 87° × 58°
	Minimum Depth Distance (Min-Z) at Max Resolution: ~52 cm	**Depth output resolution:** Up to 1280 × 720
	Depth Accuracy: <2% at 4 m[1]	**Depth frame rate:** Up to 90 fps
RGB	**RGB frame resolution:** Up to 1280 × 800	**RGB sensor FOV (H × V):** 90 × 65°
	RGB frame rate: 30 fps	**RGB sensor resolution:** 1 MP
	RGB sensor technology: Global Shutter	
Major Components	**Camera module:** Intel RealSense Module D450	**Vision Processor Board:** Intel RealSense Vision Processor D4
Physical	**Form factor:** Camera Peripheral	**Connectors:** USB-C* 3.1 Gen 1*
	Length × Depth × Height: 124 mm × 26 mm × 29 mm	**Mounting mechanism:** – One 1/4-20 UNC thread mounting point – Two M4 thread mounting points – Tripod

Figure 36: RealSense D455 technical sheet