



Tactile perception method development for a mobile walking robot

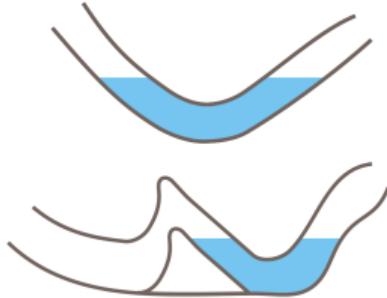
Student: Oleg Bulichev

Supervisor: Alexander Maloletov

Motivation: why do we need to explore caves by robots



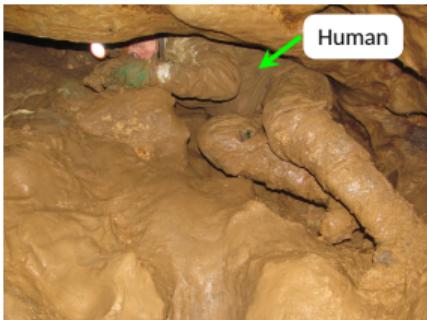
Salt



Siphon



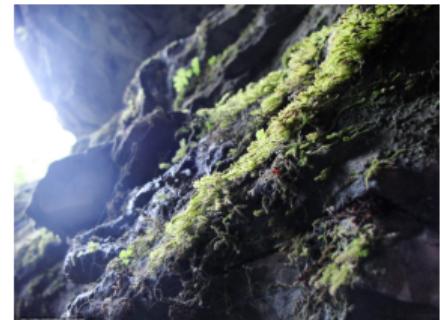
Glacier cave



Clay



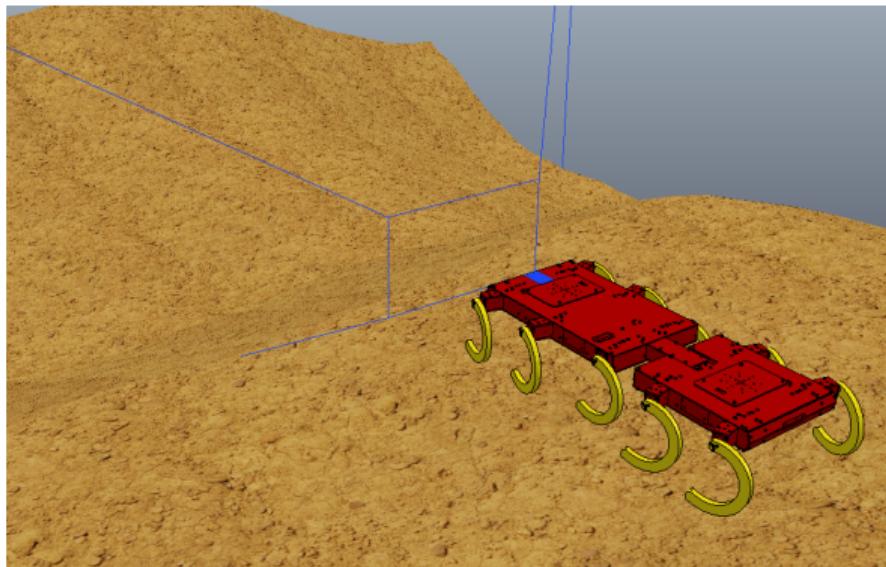
Puddle



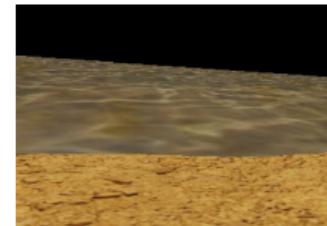
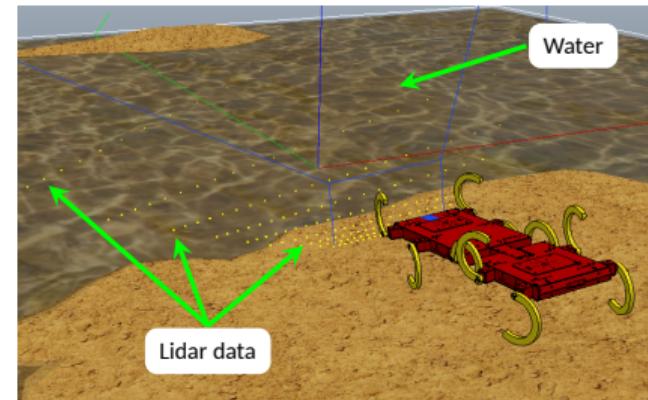
Moss

Motivation: unsolvable problem for cameras and lidars

Question: how to make a map when you have a puddle above a surface?



Terrain without water



Camera view



Problem statement

Problem 1

How to obtain a useful information on terrain, **when we have a SLAM based on lidars, cameras?**



Problem statement

Problem 1

How to obtain a useful information on terrain, **when we have a SLAM based on lidars, cameras?**

Obtain map and terrain type



Proposed solutions

Problem 1

*Map can be built using tactile sensors on each leg of the robot and create a **dense point cloud** using sampling from generated mesh from **modified Delaunay triangulation**.*

*Terrain type can be obtained solving **terrain classification** problem using **machine learning**.*



Literature review

Consider issues:

- Cave environment: obstacles, dimensions.
- Robots for cave exploration: from zeppelins, to quadruped robots.
- Methods for map creation: using classical and haptic. Based on cameras, lidars, tactile sensors, etc.

Existing problems

- Robotics systems for exploring loose caves
- Object mesh creation using tactile sensor installed on manipulator
- Map creation using lidars and cameras



Literature review

Consider issues:

- Cave environment: obstacles, dimensions.
- Robots for cave exploration: from zeppelins, to quadruped robots.
- Methods for map creation: using classical and haptic. Based on cameras, lidars, tactile sensors, etc.

Existing problems

- Robotics systems for exploring loose caves
- Object mesh creation using tactile sensor installed on manipulator
- Map creation using lidars and cameras

New problem is proposed



Robot design

Requirements

Problem – choose robot mover type. This robot should:

- should have *small dimensions* to sneak through holes;
- have enough *off-road passability* to pass a granular terrain;
- should *traverse through small water obstacles*;
- can *climb on big stones*;



Robot design

Requirements

Problem – choose robot mover type. This robot should:

- should have *small dimensions* to sneak through holes;
- have enough *off-road passability* to pass a granular terrain;
- should *traverse through small water obstacles*;
- can *climb on big stones*;

Cycling robot with 1 DoF leg



Robot design

Structural synthesis problem

Question

What the optimal number of legs should be in such robot?



Robot design

Structural synthesis problem

Question

What the optimal number of legs should be in such robot?

Answer

Robot should have **8-14 legs** in total!

Robot design

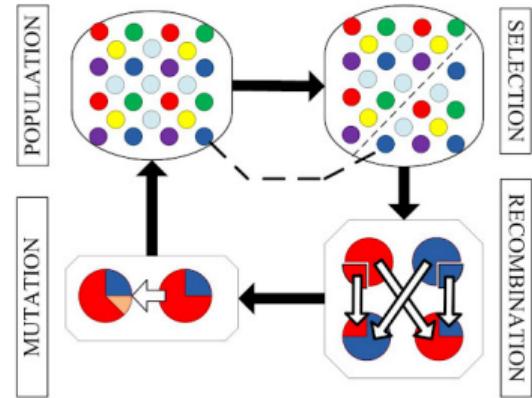
Technological stack



*Generating terrain approach
(Robot traverse an **artificial**
terrain based on **generating**
parameters)*



Robot simulator



*Genetic algorithm
(DEAP Library)*



Robot design

Assumptions

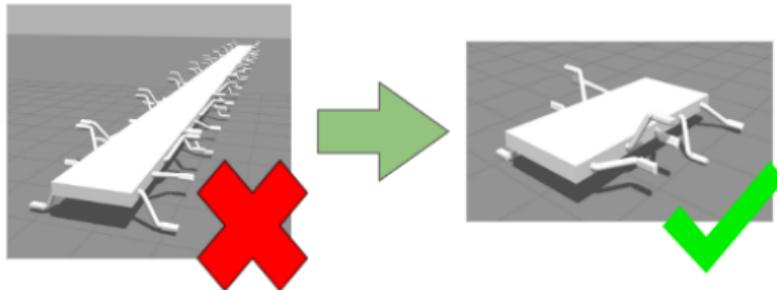
- Generated terrain family with the same constants has the same complexity.
Parameters:

- Cell width and length
- Cell height range
- Distribution parameter

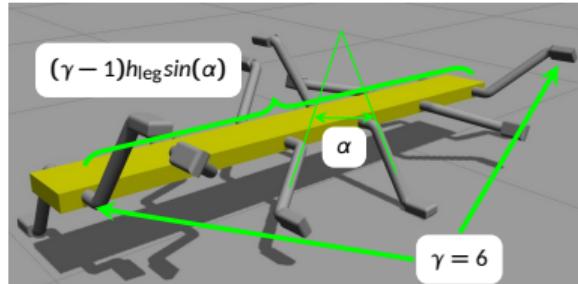


Robot design

Proposed solution



Idea: Minimize number of legs without losing off-road passability



$$F \rightarrow \max = \beta \left(\omega_1 \cdot \underbrace{\delta}_{\text{Distance}} + \omega_2 \cdot \frac{\overbrace{1}{\text{Simplified body length}}}{(\gamma - 1)h_{\text{leg}}\sin(\alpha)} \right) + \\ + (1 - \beta)\delta^{\omega_1} \left(\frac{1}{(\gamma - 1)h_{\text{leg}}\sin(\alpha)} \right)^{\omega_2}$$

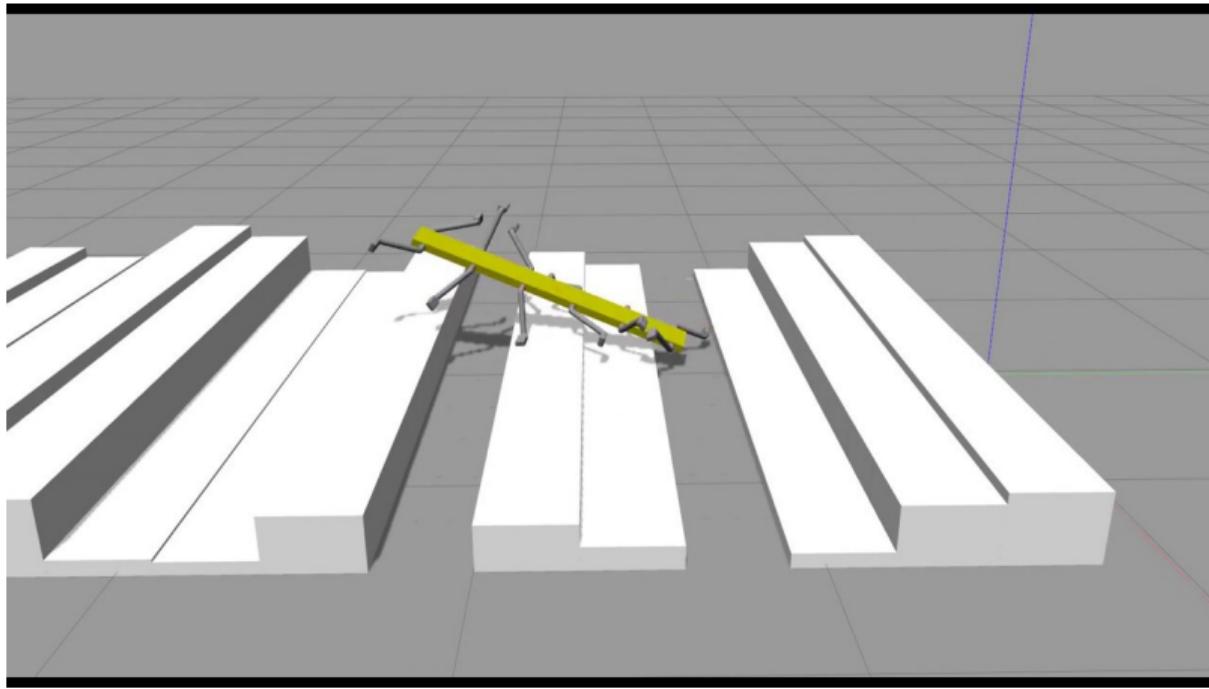
β is adaptive parameter,

$\omega_{1,2} \in [0..1]$ are the weight coefficients.



Robot design

Video: the story of one generated robot





Robot design

Result interpretation

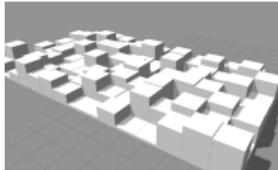
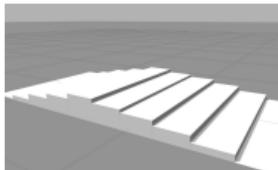
Based on fitness function the number of legs range starts from 8. It can be explained by static stability criteria. In such case 4 legs will touch the ground.

TODO



Robot design

Results

	Terrain types	No. legs per side	Angle b/w neighbor legs	No. individuals
Phase 1		6	73	200
			72	
Phase 2		5	68	
			77	55

Summary: created robot should have 10-12 legs in total



Robot design

Increasing maneuverability

Question

1. Long robot can stuck in a cave hole, while rotating. How to avoid it?
2. How to climb on big stones?



Robot design

Increasing maneuverability

Question

1. Long robot can stuck in a cave hole, while rotating. How to avoid it?
2. How to climb on big stones?

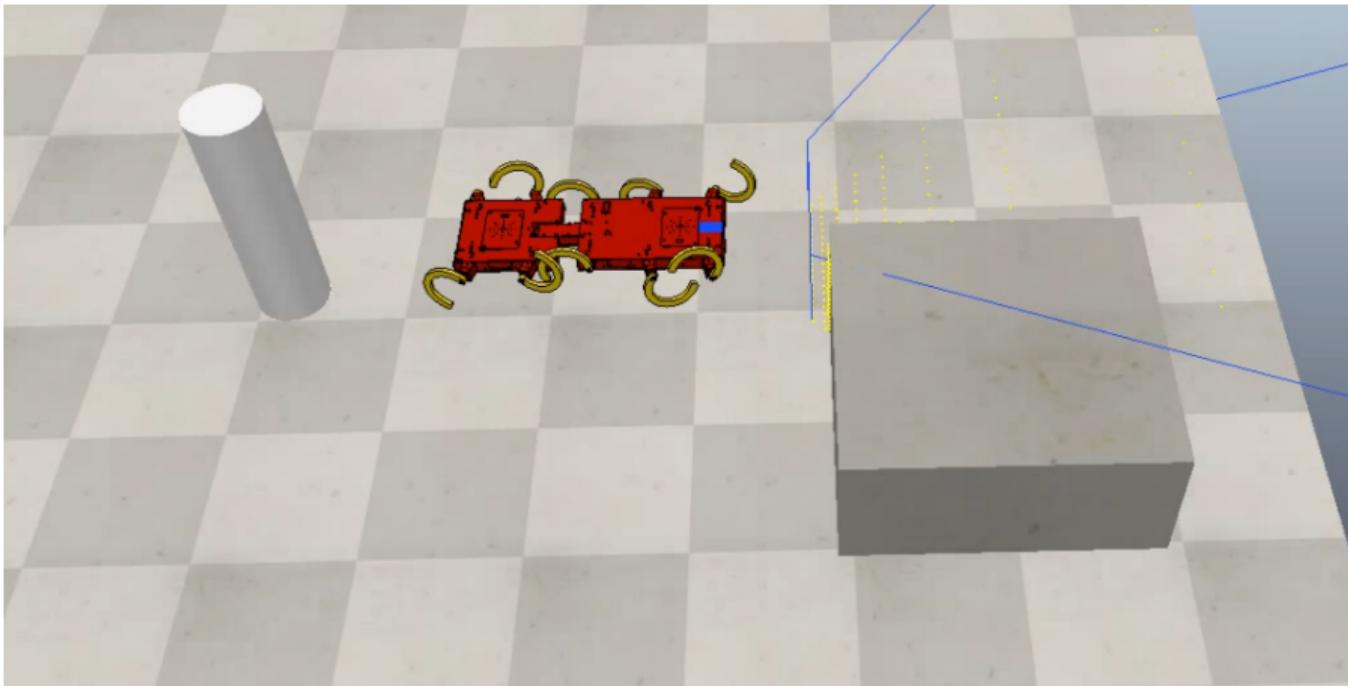
Answer

1. Add an ability to sidestep without changing orientation.
2. Make a segmented body.



Robot design

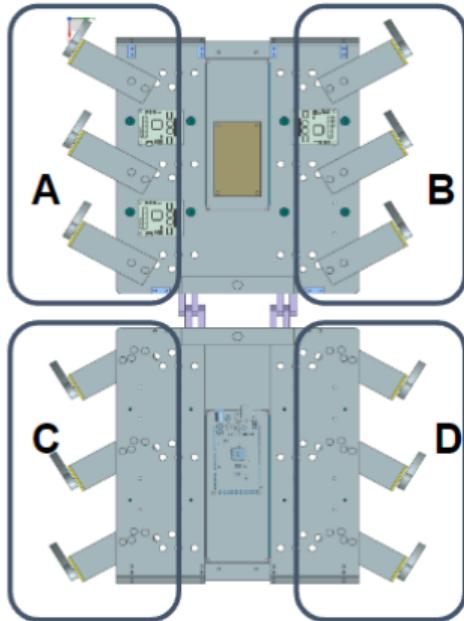
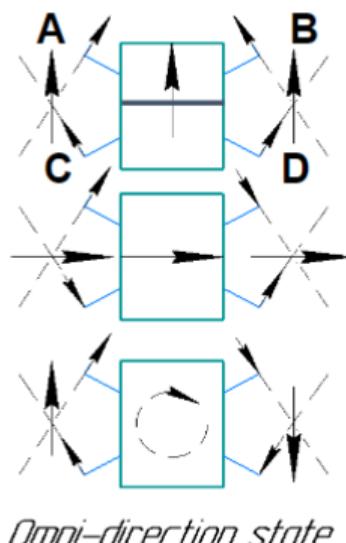
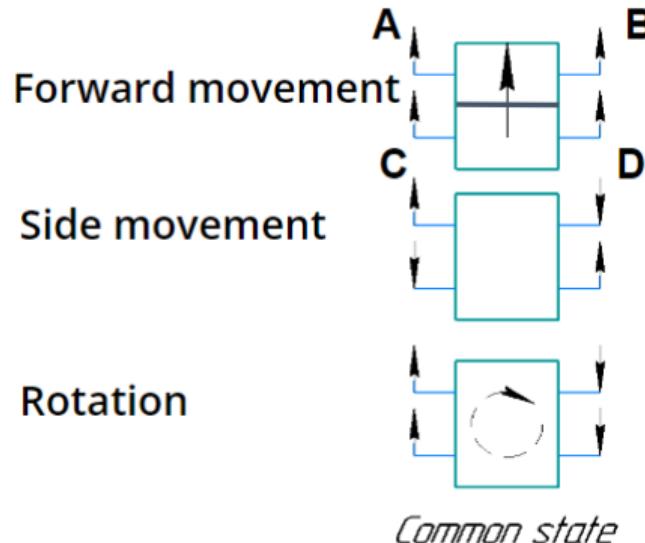
Video





Robot design

Proposed solution

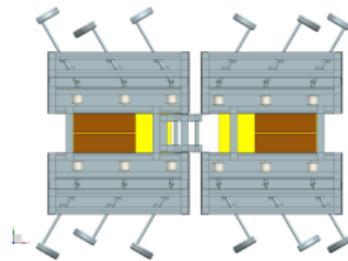
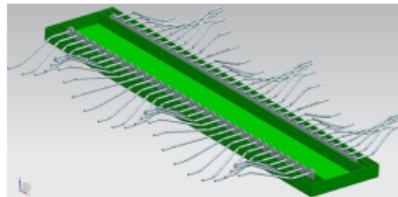


Vector representation of forces in the conventional and omni-direction states



Robot design

StriRus prototypes (1)

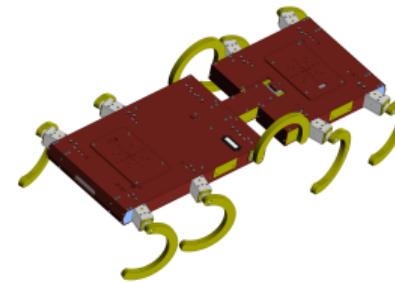
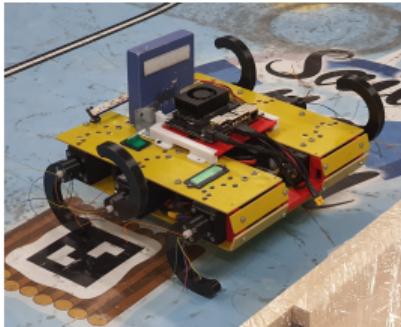


Generation	1	2	3
No. legs	54	12	12
No. segments	1	2	2
Connect. joint	—	Pitch	Pitch, Yaw
Rel. body-leg angle, deg	0	0-45	0, 15, 30, 45
Leg height, mm	54	60	60
Features	Wave propelled robot	Continuous mechanism	2 DoF connection joint mech.
Withdraws	<ul style="list-style-type: none">- Cannot install force sensors- high friction	<ul style="list-style-type: none">- Too complicated changing relative angle mechanism	<ul style="list-style-type: none">- Small legs- Useless 2nd DoF in connection joint



Robot design

StriRus prototypes (2)



Generation	3+	4
No. legs	6	10
No. segments	1	2
Connect. joint	—	Pitch
Rel. body-leg angle, deg	0	0, 15
Leg height, mm	90	180
Features	Larger legs	Largest legs
Withdraws	- Still small legs - 1 segment	-



Force transducer design

Question

How to receive a reaction force from the ground?



Force transducer design

Question

How to receive a reaction force from the ground?

Answer

- Current/voltage measurements from motors
- Installing torque sensor on a shaft
- Installing a force sensor on each leg



Force transducer design

Force sensor types

- **F/T sensors:** too massive and expensive for small robots.
- **Optical:** too thick.
- **Magnetic:** too thick.
- **Capacitive:** expensive, but the best in terms of requirements.
- **Piezoresistive sensors based on conductive inks or polymers:** inexpensive and robust, but has problems with hysteresis
- **Stain gages:** challenge for wiring between continuously rotating legs and the robot body.



Force transducer design

Force sensor types

- **F/T sensors:** too massive and expensive for small robots.
- **Optical:** too thick.
- **Magnetic:** too thick.
- **Capacitive:** expensive, but the best in terms of requirements.
- **Piezoresistive sensors based on Velostat:** inexpensive and robust, but has problems with hysteresis
- **Stain gages:** challenge for wiring between continuously rotating legs and the robot body.

Force transducer design

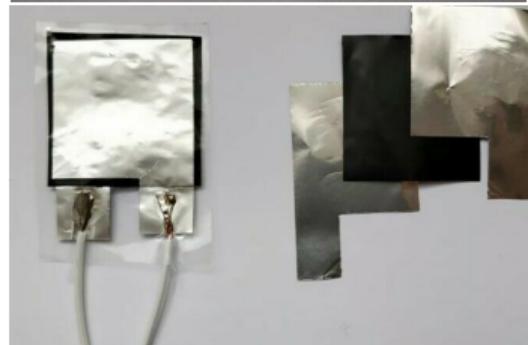
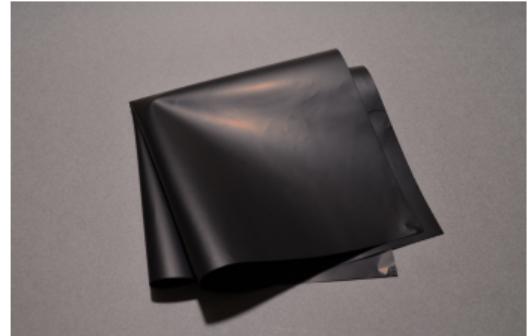
Velostat

Definition

The Velostat is a polymer material filled with carbon black.

Expected effects:

- Quantum tunnelling (Precipitation) – Diod is using such effect;
- Piezoresistive – electrical resistivity of semiconductor is changed by mechanical strain;
- Viscoelasticity – material can damp vibrations.



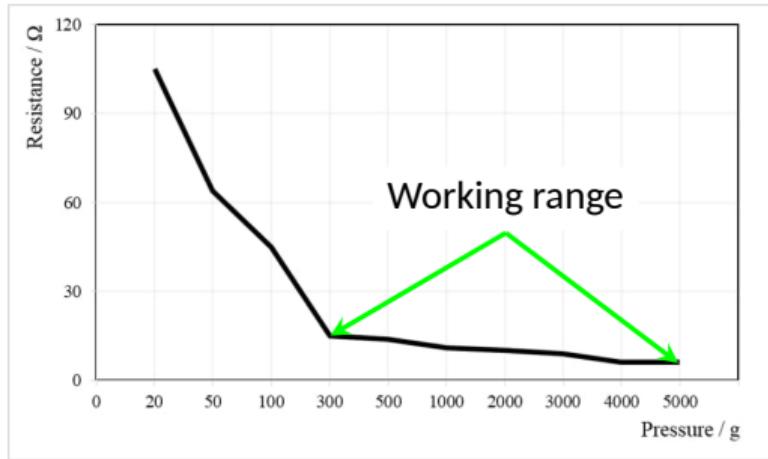
Simplest force transducer



Force transducer design

Velostat: Faced problems

- Hysteresis
- Nonlinearity
- Different values with the same pressure if the square of load is less than the sensor



Scientific Problem Statement

To characterize Velostat material for cases when point load is less than sensor size and propose solutions for avoiding such issues.



Force transducer design

Experimental setup requirements

- Force control.
- Position and force repeatability.
- To have an ability to apply force only on a particular part of a sensor.



Force transducer design

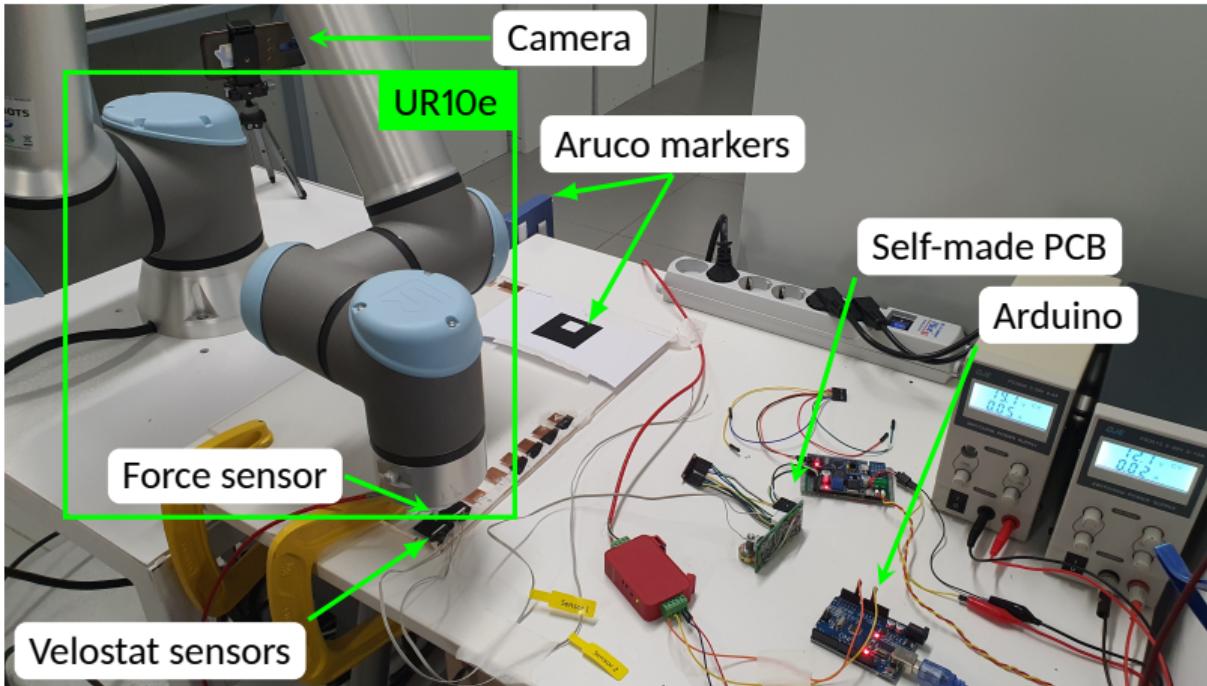
Experimental setup requirements

- Force control.
Solved by Impedance Control
- Position and force repeatability.
Solved by adding a manipulator and a camera
- To have an ability to apply force only on a particular part of a sensor.
Solved by adding several end-effectors

All setup requirements are fulfilled

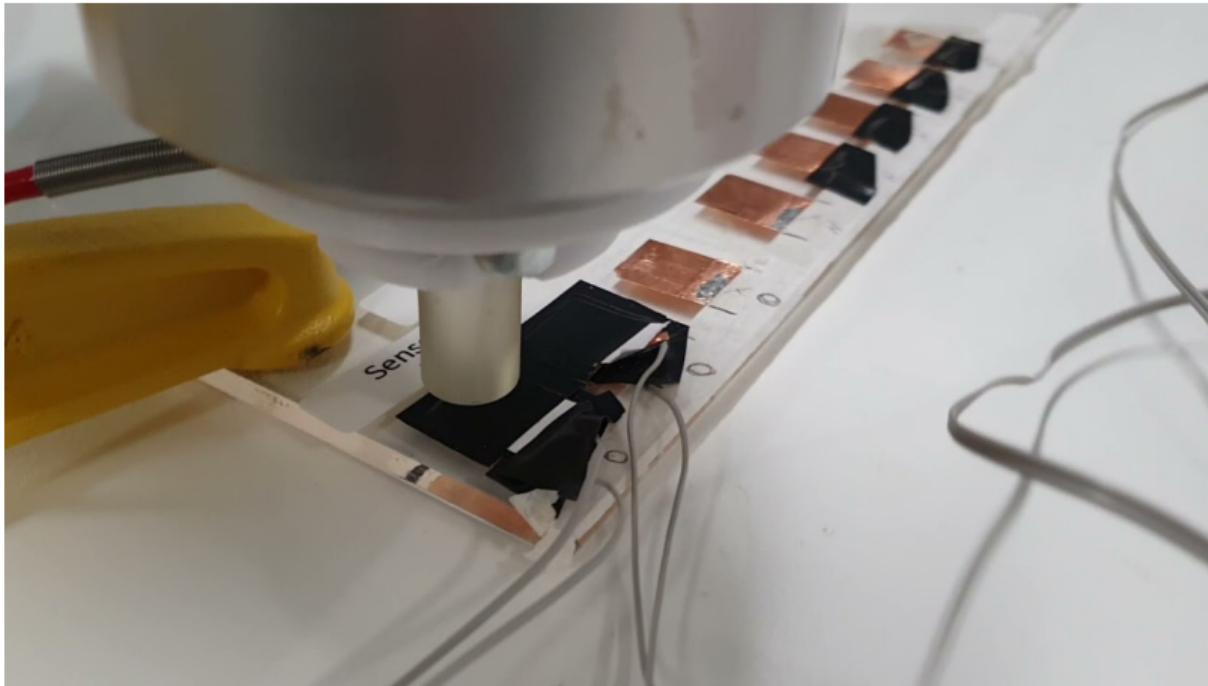
Force transducer design

Experimental Setup: Hardware overall



Experiment Design

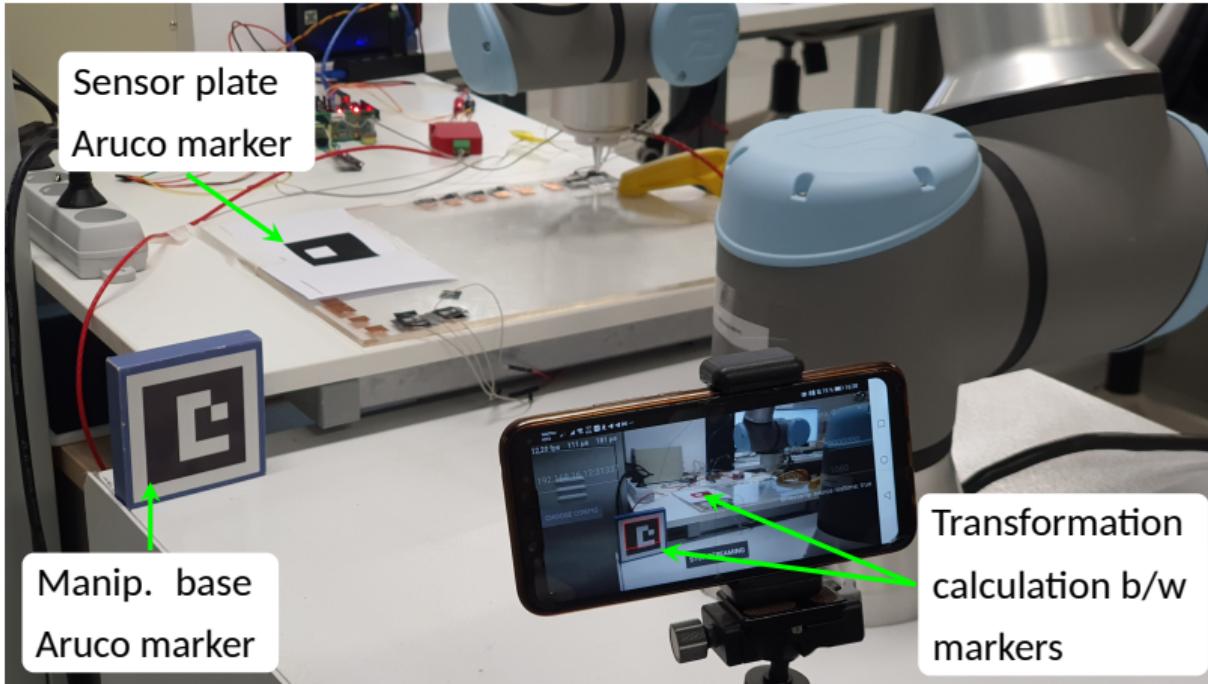
Experimental Setup: Hardware, video





Force transducer design

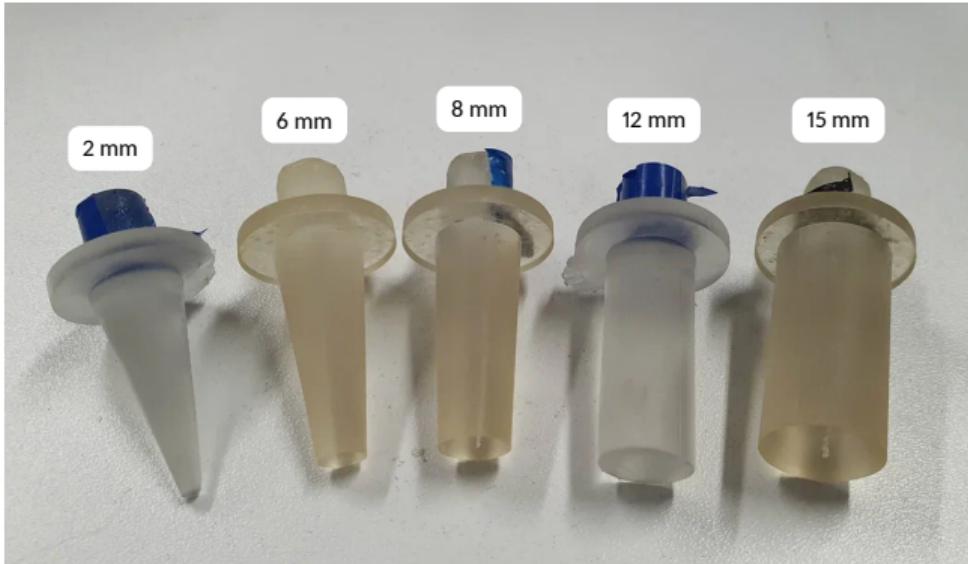
Experimental Setup: Hardware, Aruco markers





Force transducer design

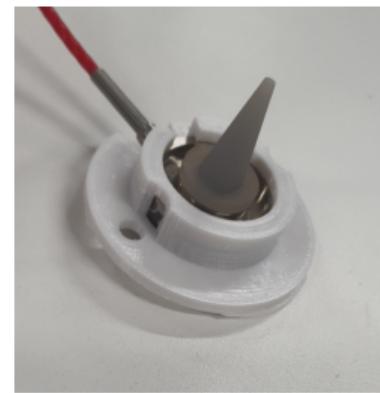
Experimental Setup: Hardware, end-effector



All end-effectors



*Ground Truth
Force sensor*

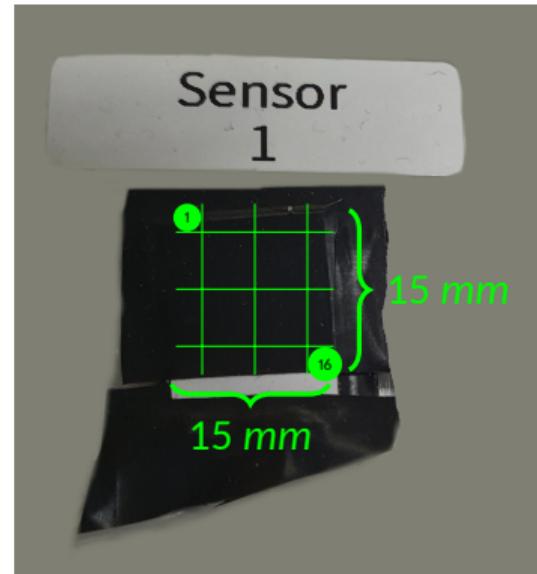


*End-effector in
assembly*

Force transducer design

Experiments

1. **Static experiment.** The goal is to identify the coefficients for the transducer model.
2. **Dynamic experiment.**
 - We are representing a transducer as a 4×4 grid. We touch with the same pressure using five different end-effectors (area starting from 2mm till 15mm).
 - We are using 2 mm and 15 mm end-effectors. We touch with different forces (5, 10, 20, 30, 40 H).



*Sensor representation
as 4×4 grid*



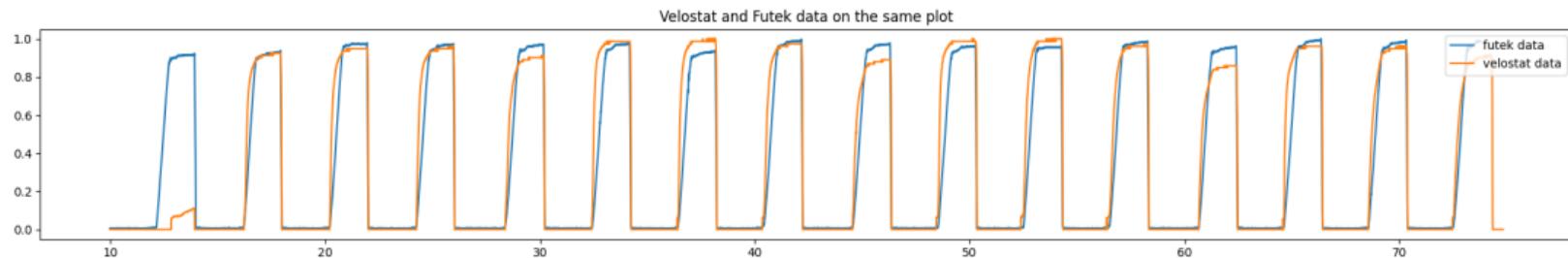
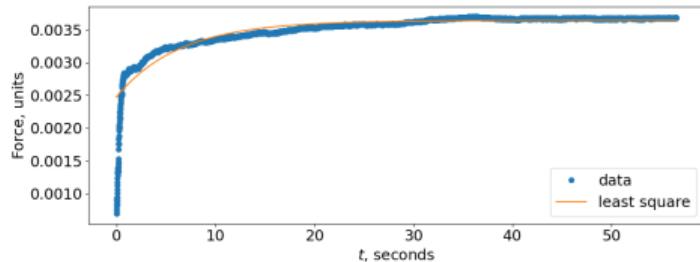
Force transducer design

Result: static experiment

$$V_{out} = V_0 + p[k_p + k_e(1 - e^{\frac{-(t-t_0)}{\tau_{res}}})](1 - e^{-\frac{A}{p}})$$

$$k_p = A_1 e^{-A_2 p}; \tau_{res} = B_0 + B_1 e^{-\frac{p}{B_2}}$$

Where V_0 - initial voltage, p , A_i , B_i , τ_{res} , k_i are constants, t - current time, t_0 - the time when the pressure appeared.

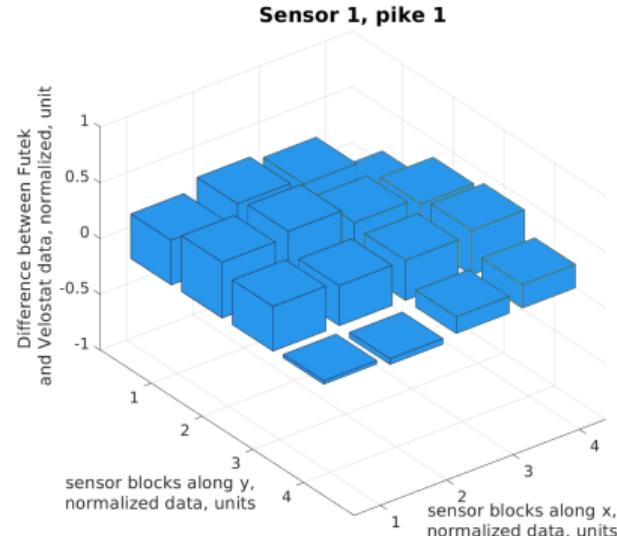


Normalized force data from sensor and transducer

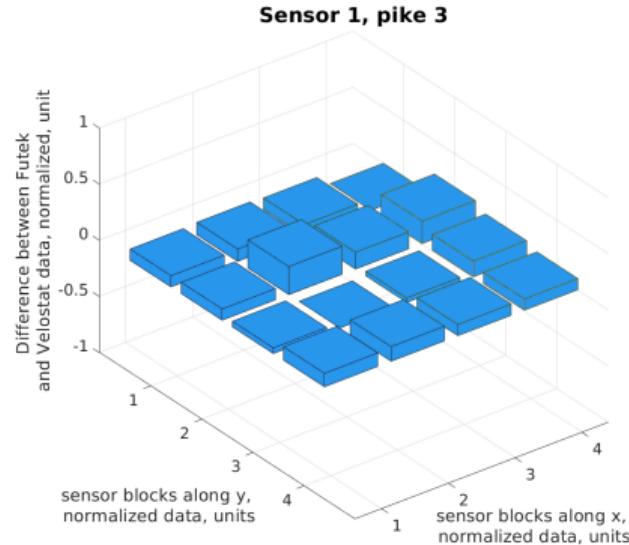


Force transducer design

Result: dynamic experiment



2mm end-effector diam



8mm end-effector diam



Force transducer design

Summary

1. Static experiment: transducers coefficients p , A_i , B_i , τ_{res} , k_i were identified.
2. Dynamic experiment: a transducer can be represented as a single body, when pressure area is higher than 50% of the sensor area.



Terrain classification

Question

How to define the terrain type during the movement on such terrain?



Terrain classification

Question

How to define the terrain type during the movement on such terrain?

Answer

Solving terrain classification problem using Machine learning



Terrain classification

Experimental setup requirements

- To have a possibility to install new surfaces and change it quickly.
- Infinite movement.
- Movement part should be the same as on the StriRus.



Terrain classification

Experimental setup requirements

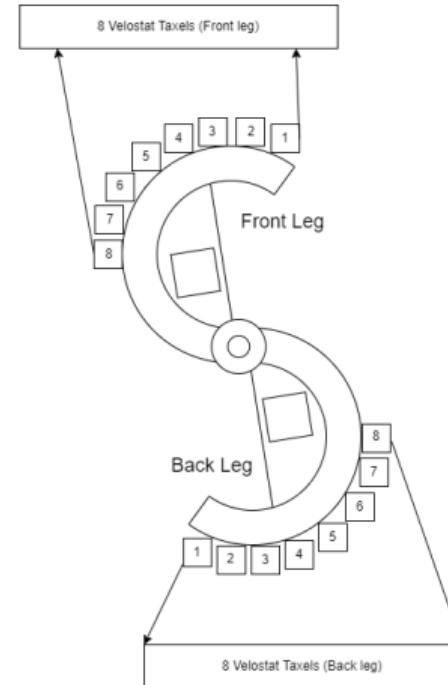
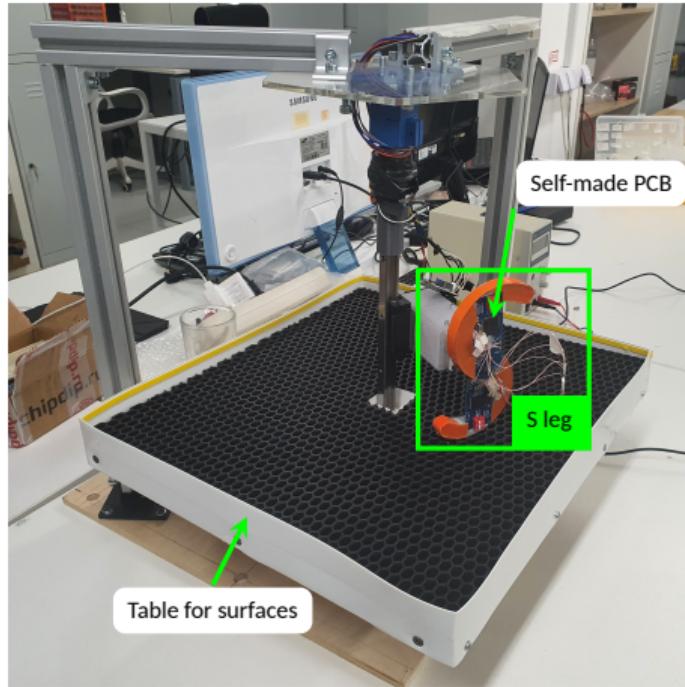
- To have a possibility to install new surfaces and change it quickly.
Solved by quick detachable table
- Infinite movement.
Solved by creating the 2 DoF mechanism and a S-shape leg
- Movement part should be the same as on the StriRus.
Solved by creating a mount for a leg assembly

All setup requirements are fulfilled



Terrain classification

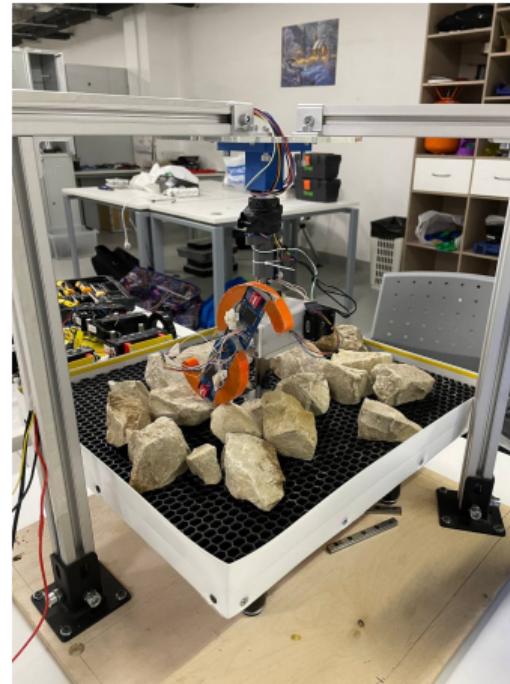
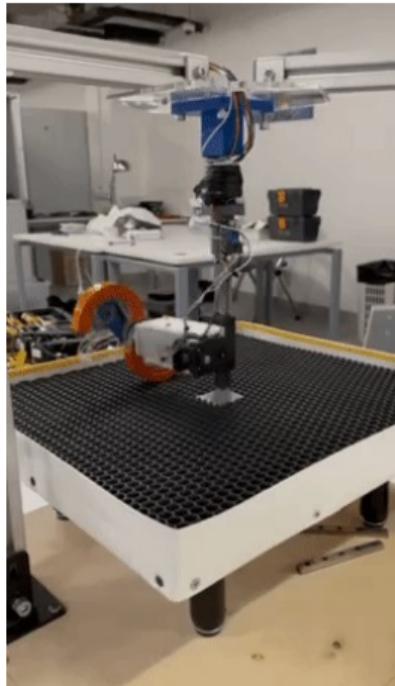
Experimental Setup





Terrain classification

Experimental setup: surface types, video





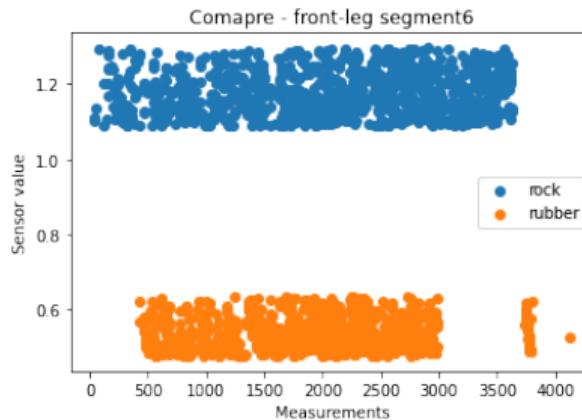
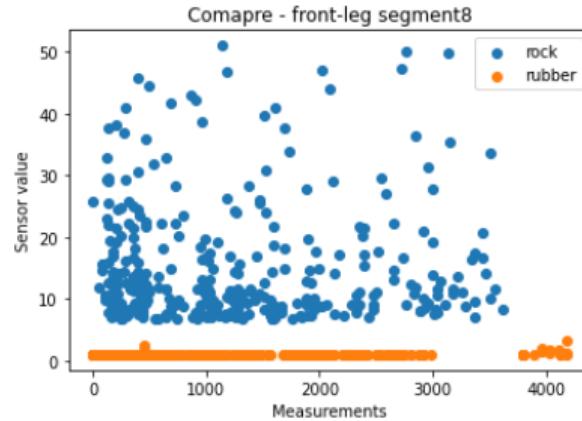
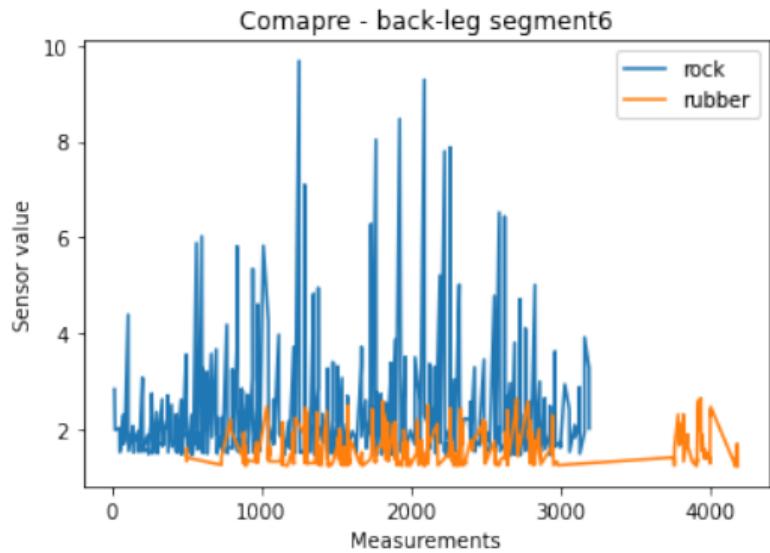
Terrain classification

Velostat transducer properties

- Because of high hysteresis and difficulties with calibration, we have to work with relative data.

Terrain classification

Obtained data from one experiment





Terrain classification

Summary

- We can distinct rubber and rock surfaces
- Terrain classification parameters was chosen:
 - RPM
 - Motor Torque
 - Acceleration from IMU
 - Force data which are represented as Sensor valuesegment, Peak amplitude, Average amplitude
- The force transducer has been proven to work



Map creation based on tactile data

Question

How to create a dense point cloud, using sparse data from legs?



Map creation based on tactile data

Question

How to create a dense point cloud, using sparse data from legs?

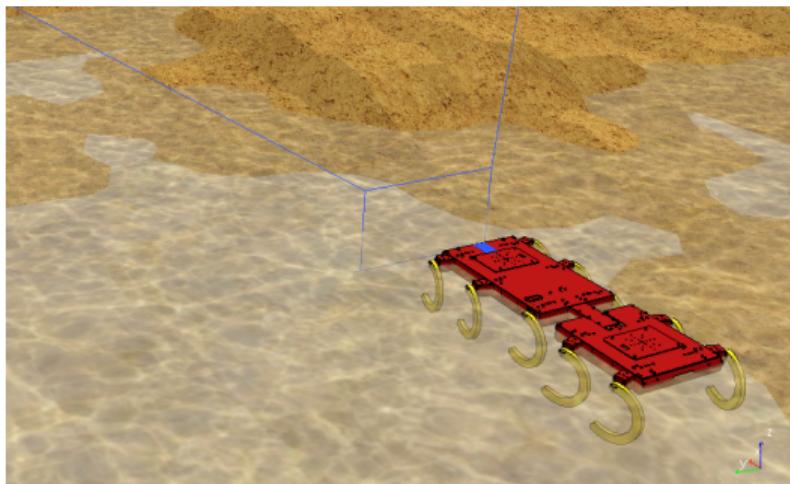
Answer

*Create a mesh, using concave hull Delaunay triangulation using sparse data,
sampling it and return to navigation*

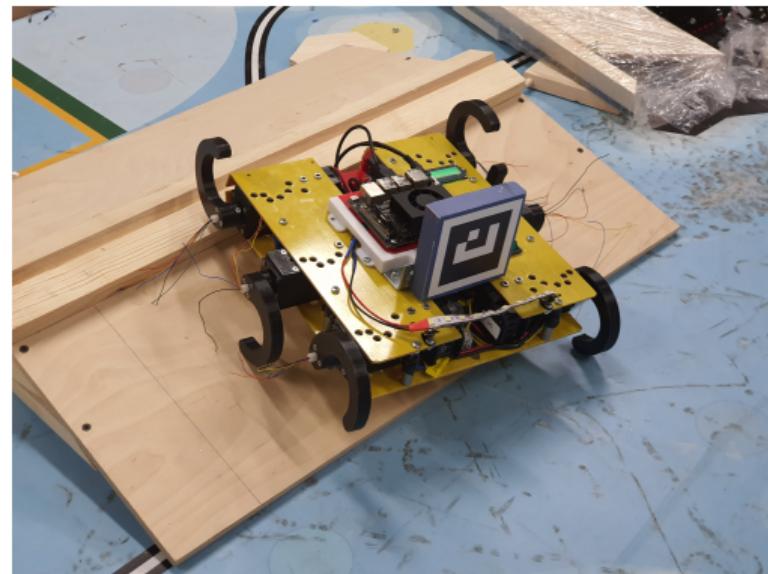


Map creation based on tactile data

Experimental setup



CoppeliaSim simulator, 4th gen StriRus



IRL, 3th+ gen StriRus



Map creation based on tactile data

Assumptions

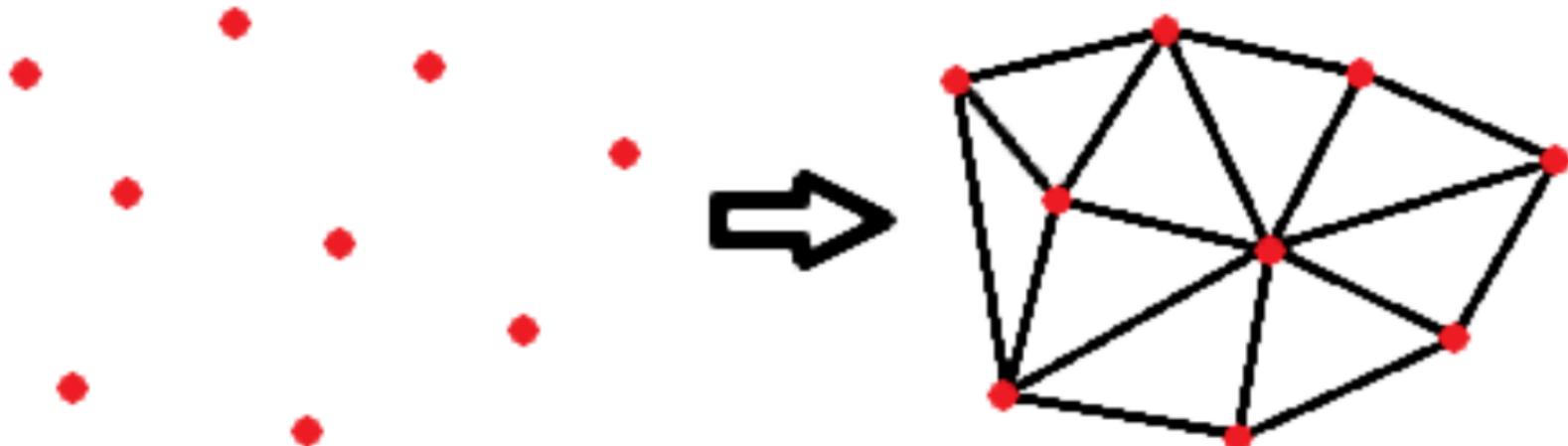
Current solution considering such assumptions:

- Our terrain can be represented $z = f(x, y)$. We can use 2D Delaunay triangulation (projected points on a plane)
- All simulation data are preprocessed by white noise



Map creation based on tactile data

Delaunay triangulation

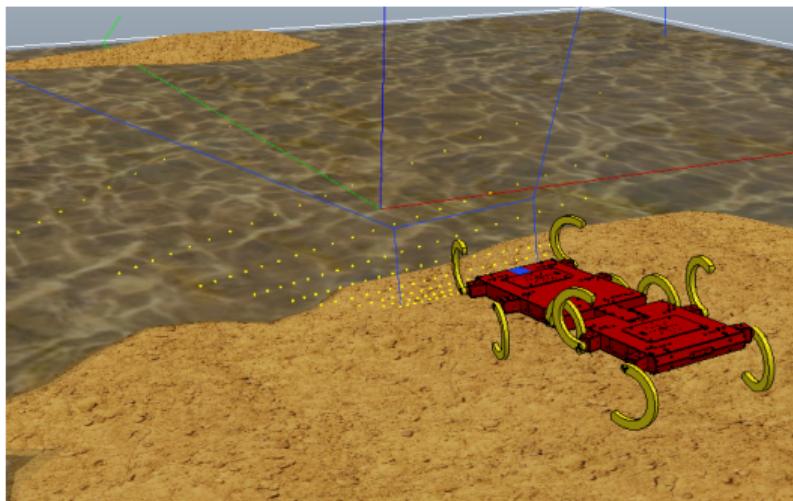


Common 2D Delaunay triangulation (Convex Hull)
From Point Cloud to mesh

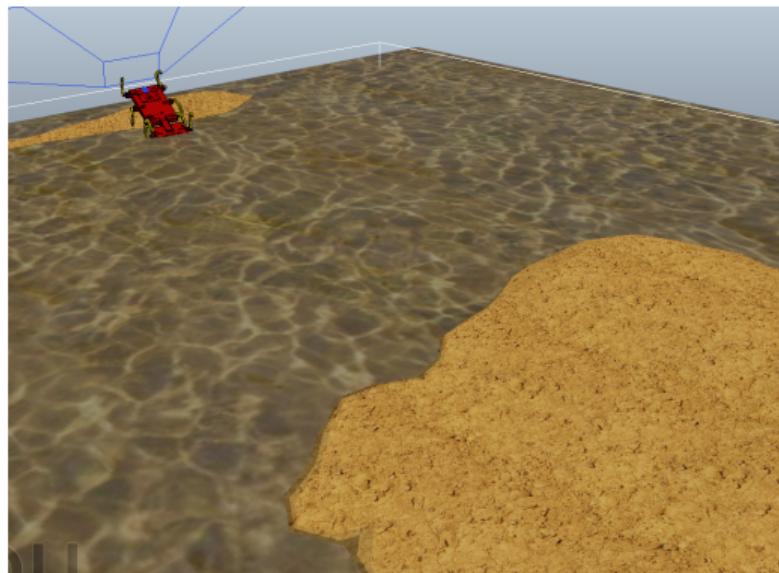


Map creation based on tactile data

Result: simulator



Start point

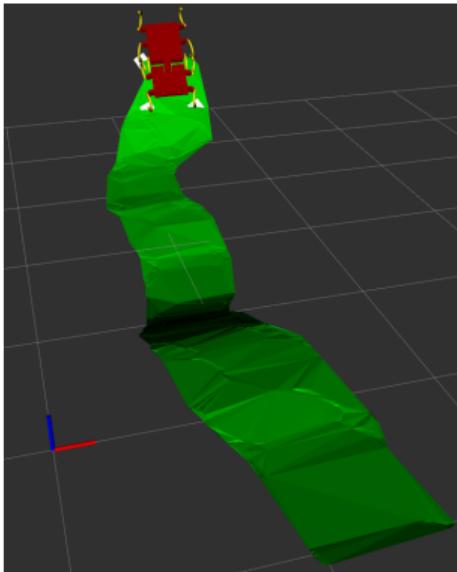


End point

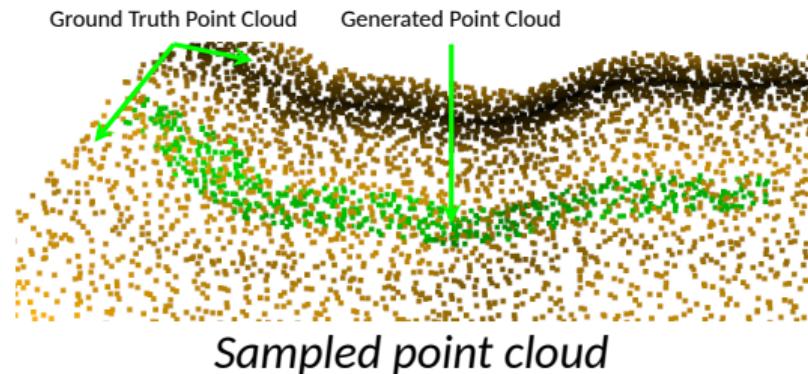


Map creation based on tactile data

Result: Mesh



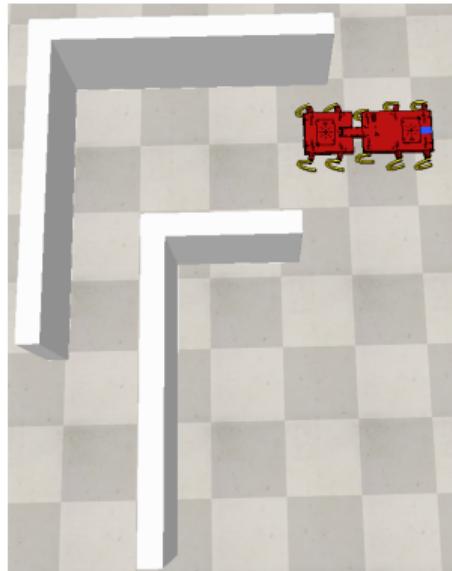
*Mesh created using concave hull 2D
Delaunay triangulation*



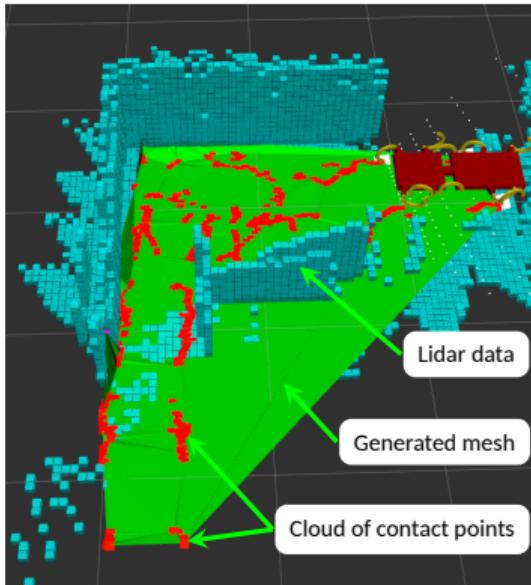


Map creation based on tactile data

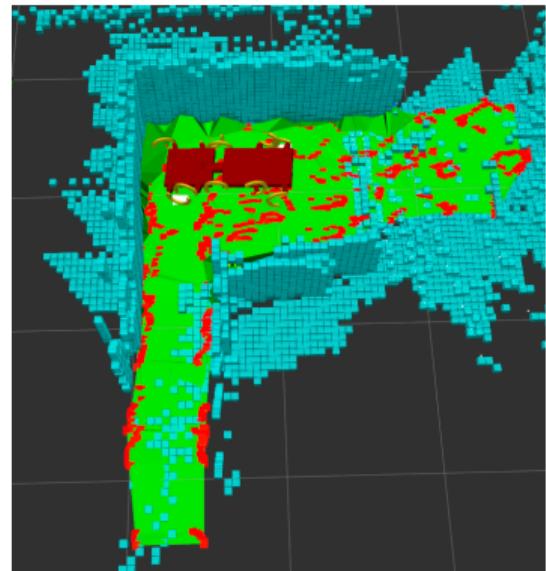
Why do we need a concave hull solution



Case study



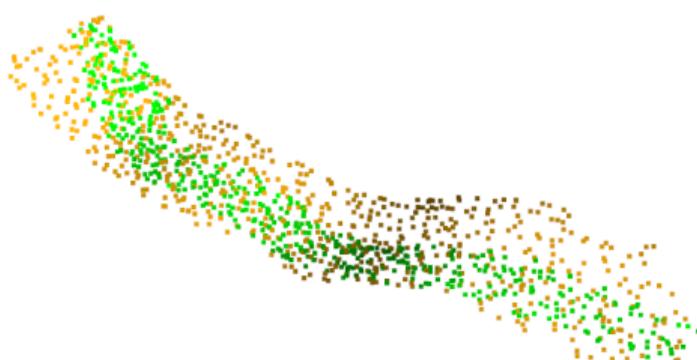
Convex Hull



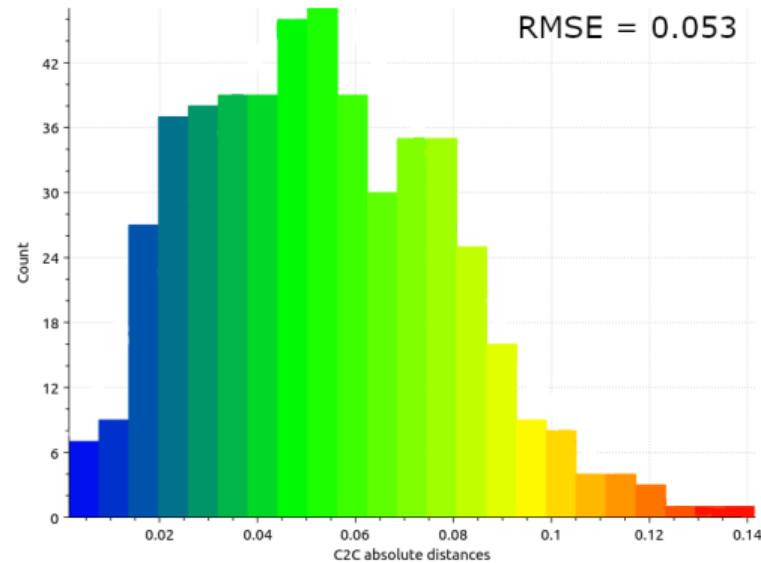
Concave Hull

Map creation based on tactile data

Metric C2C: reference and ground truth point cloud comparison



Overlaid Point Clouds

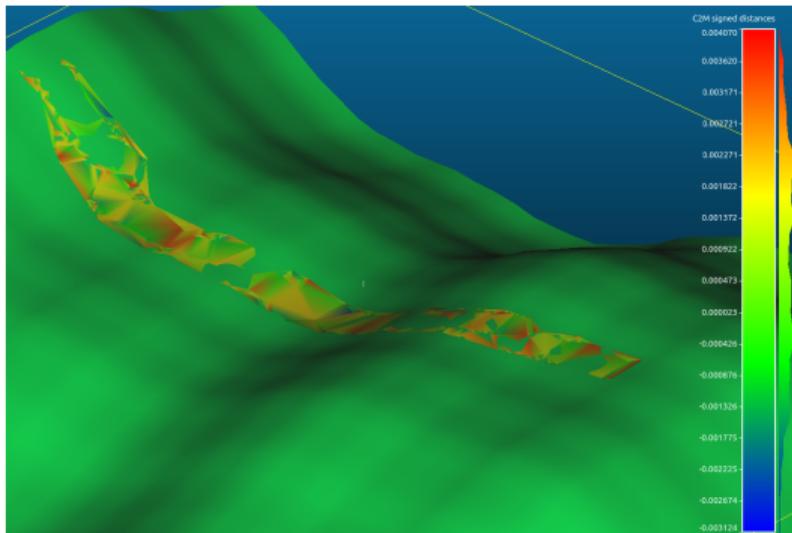


Error histogram (distance from a point to closest ground truth point)

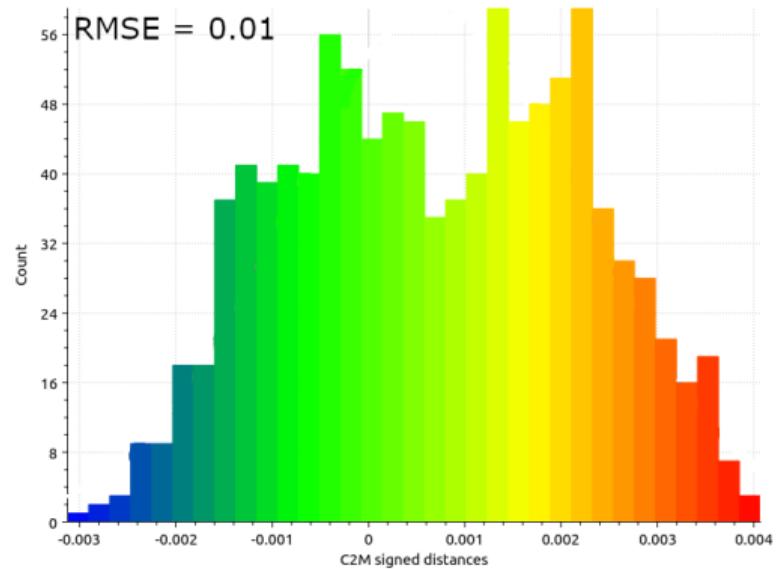


Map creation based on tactile data

Metric C2M: reference and ground truth mesh comparison



Overlaid Meshes

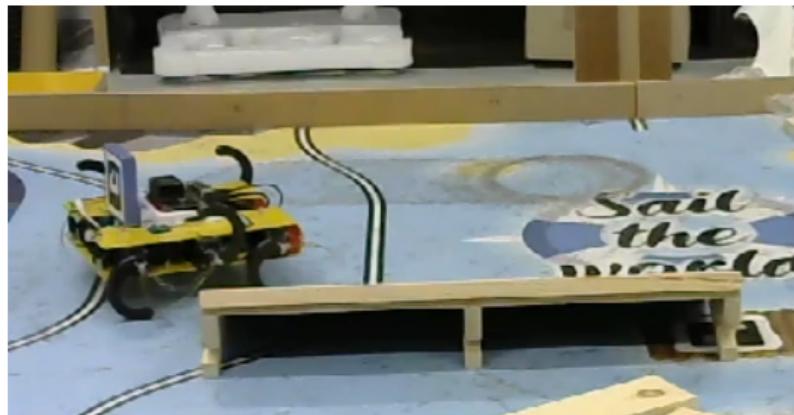


Error histogram (distance from a point to closest ground truth point)

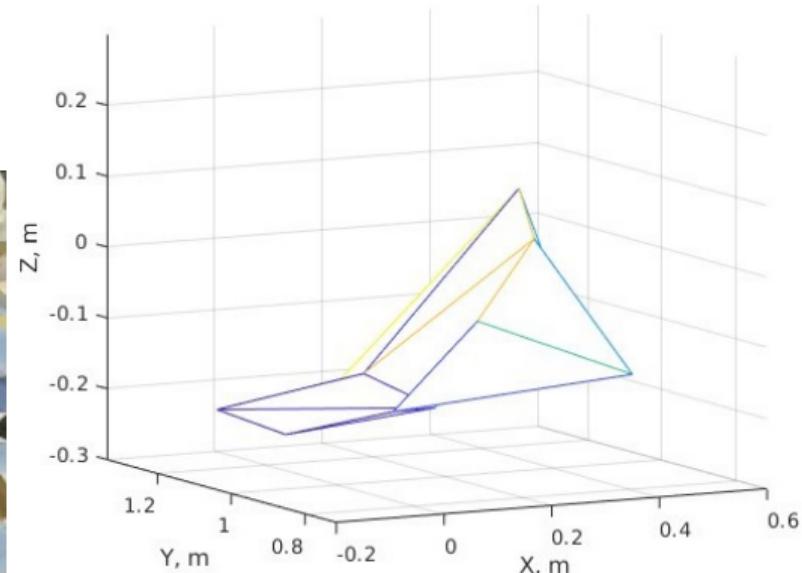


Map creation based on tactile data

Result: real world experiment, video



Robot is passing the obstacle



Mesh, obtained from legs



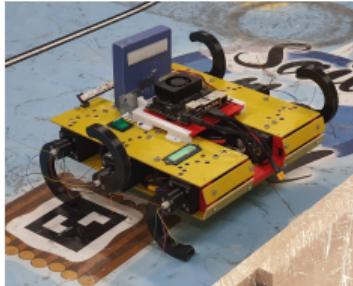
Map creation based on tactile data

Summary

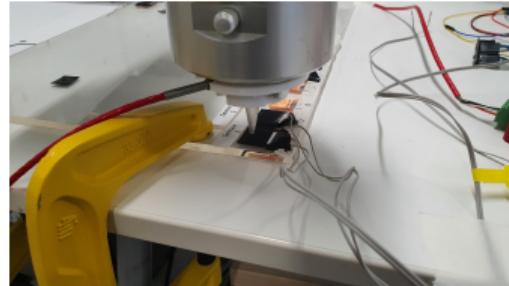
- Map can be built using concave hull 2D Delaunay triangulation.
A Sparse point cloud obtained from force sensors, installed on legs.
 - *Simulator:*
 - Avg. Point cloud comparison RMSE is about 5 cm.
 - Avg. Mesh comparison RMSE is about 1 cm.
 - *Real world experiment:*
 - Avg. Point cloud comparison RMSE is about 8 cm.
- It is appropriate accuracy for such task.



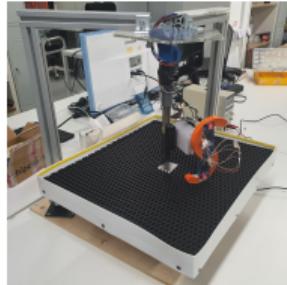
Summary



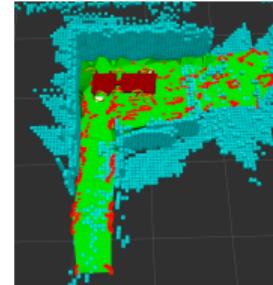
1. *StriRus was designed and assembled*



2. *Force transducer based on Velostat was created and was investigated*



3. *Robot distinct rubber and rock surfaces*



4. *Build a map using tactile sensors*