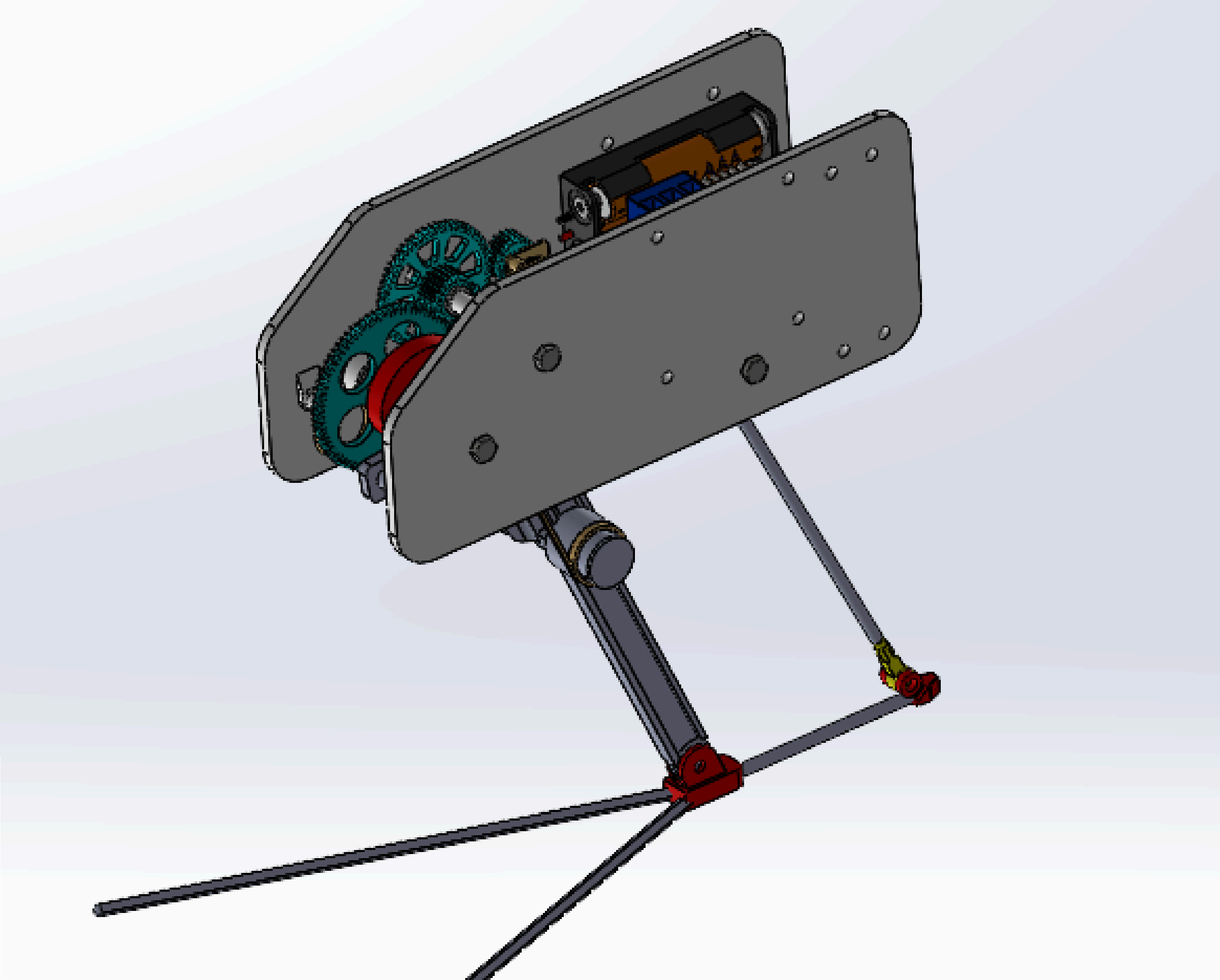
Data-driven design & fabrication methods



Jumping robot

Final Presentation

Jade Therras

Ali Fuat Sahin

Davide Lisi

Antonio Ruiz

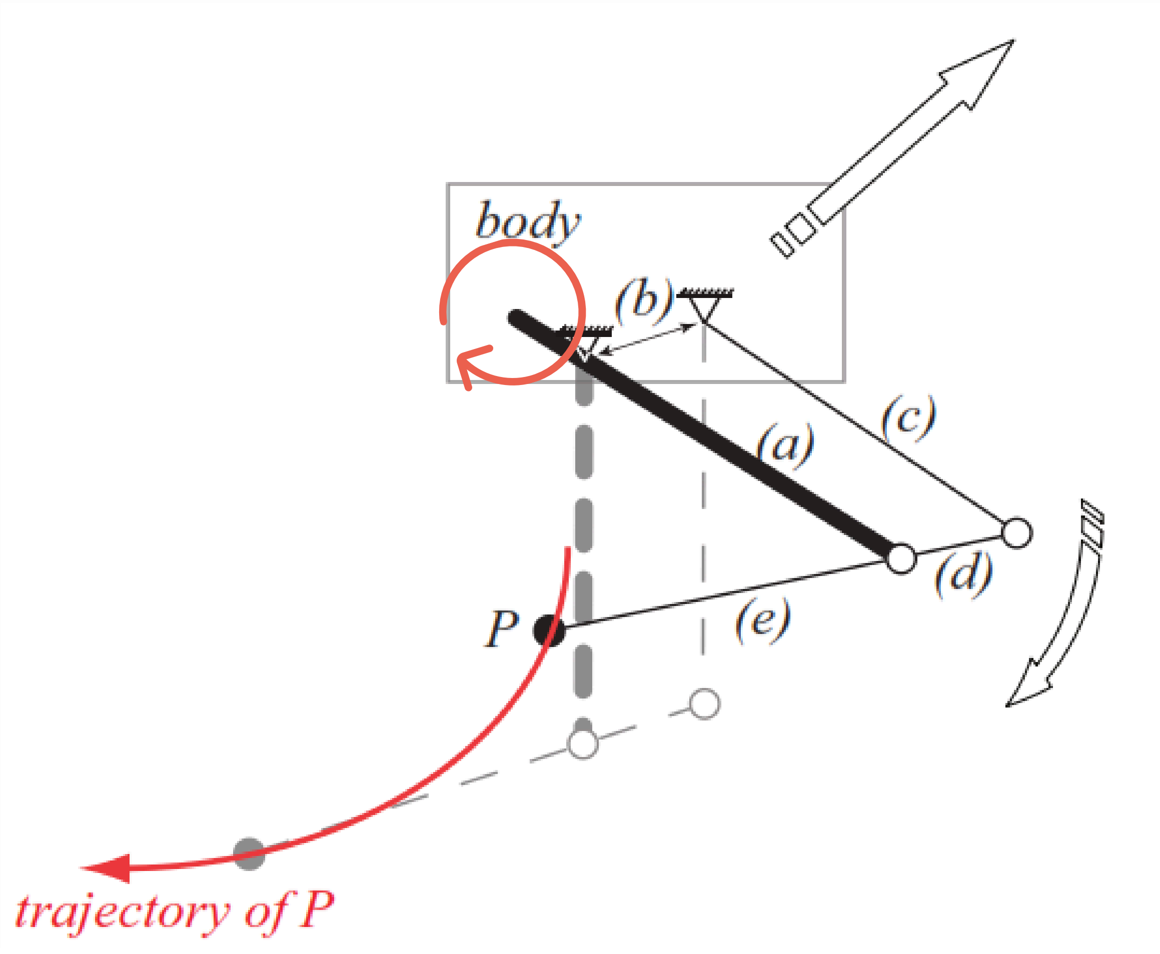
RECALL – Design problem description

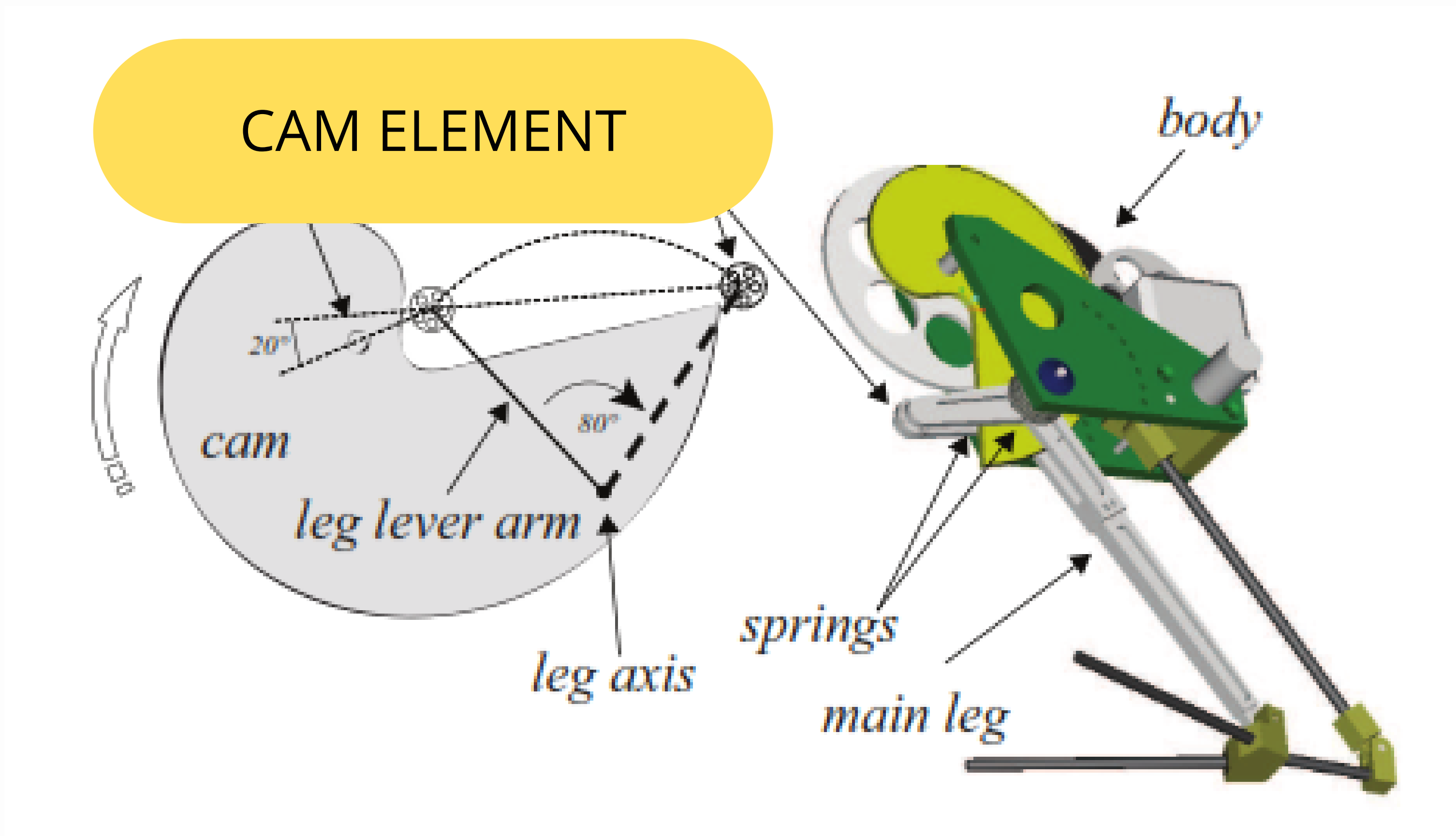
MODEL

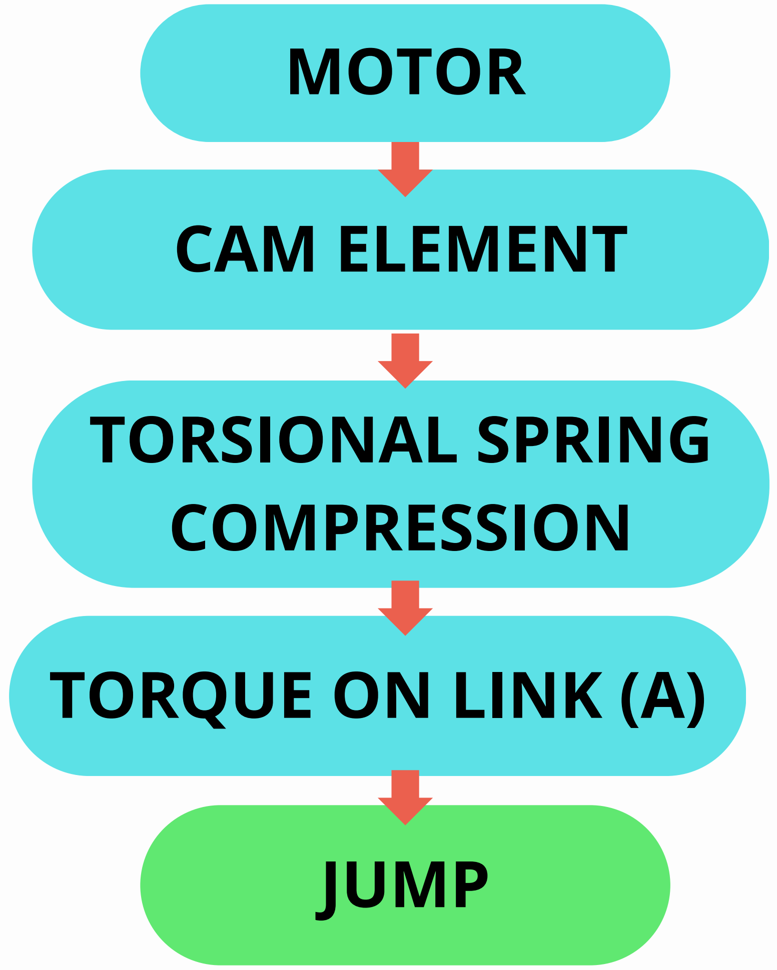
3-links Jumping Robot

OBJECTIVE

Design a robot capable of executing jumps by applying torque to link (a) as depicted in the Figure.




CAM ELEMENT


MOTOR

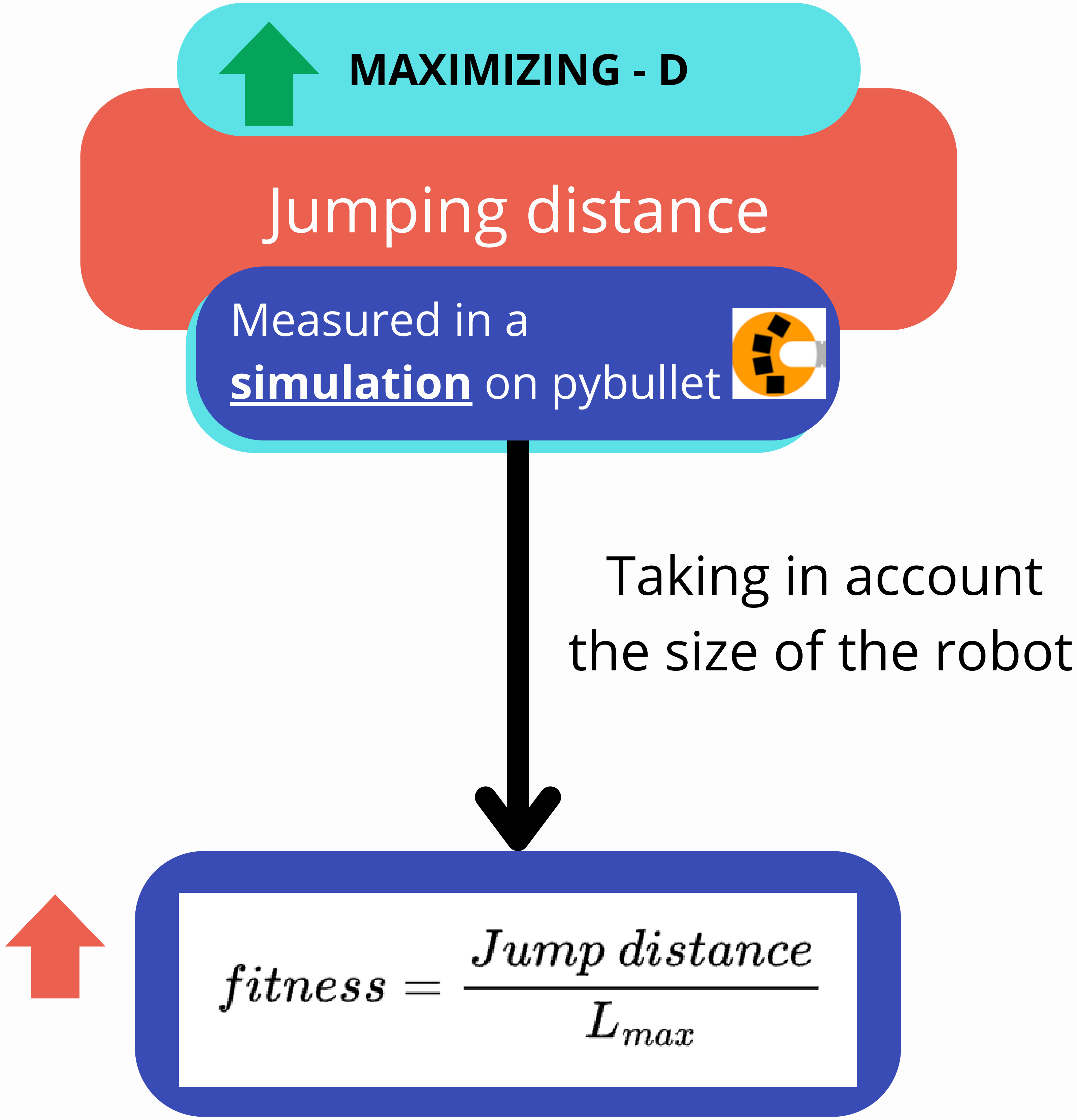
CAM ELEMENT

TORSIONAL SPRING COMPRESSION

TORQUE ON LINK (A)

JUMP

RECALL – Objective function and encoding



MAXIMIZING - D

Jumping distance

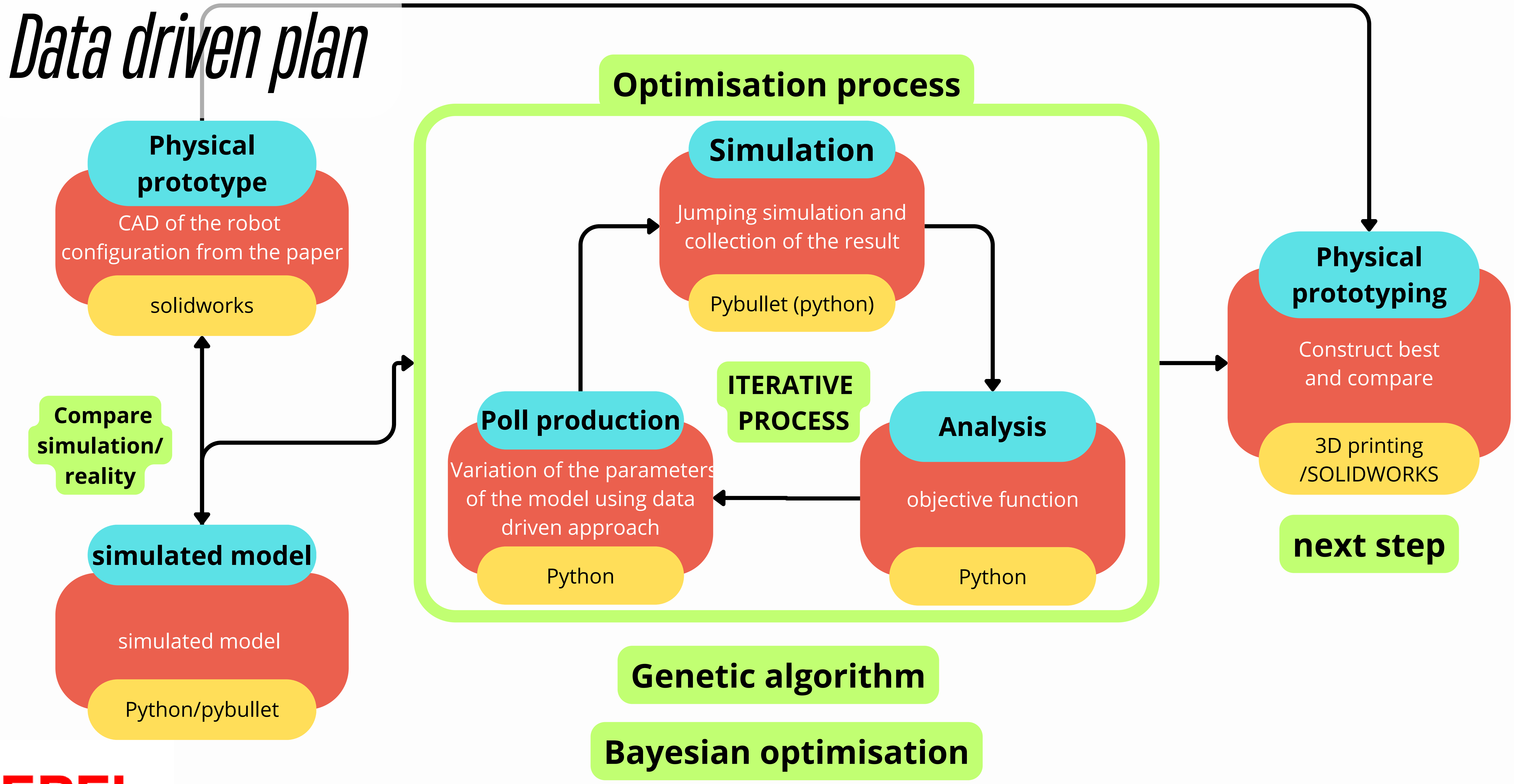
Measured in a **simulation** on pybullet

Taking in account the size of the robot

$$fitness = \frac{Jump\ distance}{L_{max}}$$

Parameters	Upper Bound	Lower Bound
Carrying Link Length	20 [cm]	10 [cm]
Compression Ratio	30 [deg]	5 [deg]
Rest Angle	80 [deg]	10 [deg]
Spring Stiffness	20 [N.mm/deg]	5 [N.mm/deg]
Ground Link Angle	80 [deg]	10 [deg]
Link (c) Coefficient	1.2	0.8
Link (d) Coefficient	0.5	0.9
Link (e) Coefficient	0.6	1.4

Data driven plan



Physical prototype

CAD of the robot configuration from the paper

solidworks

Compare simulation/ reality

simulated model

simulated model

Python/pybullet

Optimisation process

Simulation

Jumping simulation and collection of the result

Pybullet (python)

ITERATIVE PROCESS

Poll production

Variation of the parameters of the model using data driven approach

Python

Analysis

objective function

Python

Genetic algorithm

Bayesian optimisation

Physical prototyping

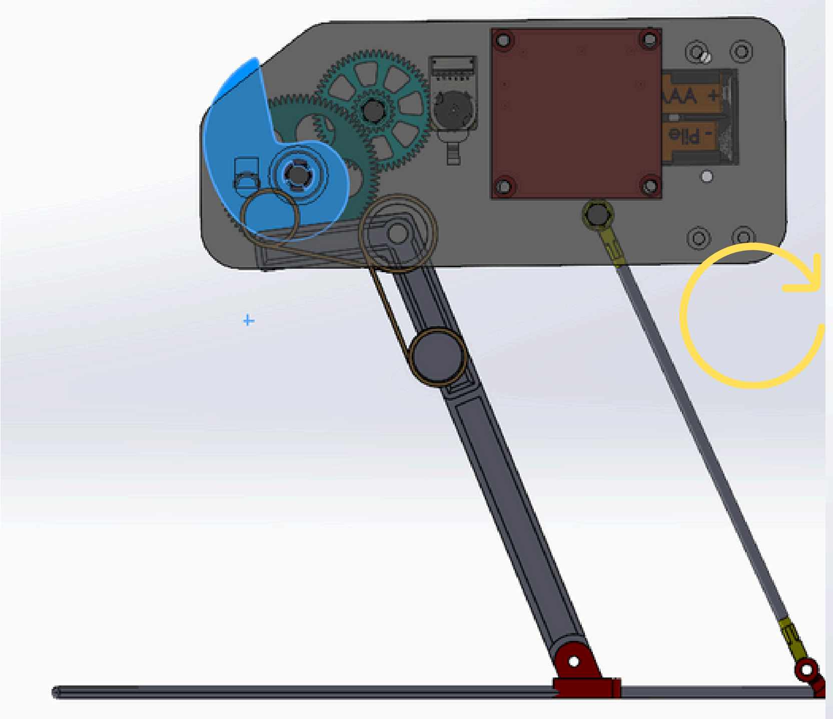
Construct best and compare

3D printing /SOLIDWORKS

next step

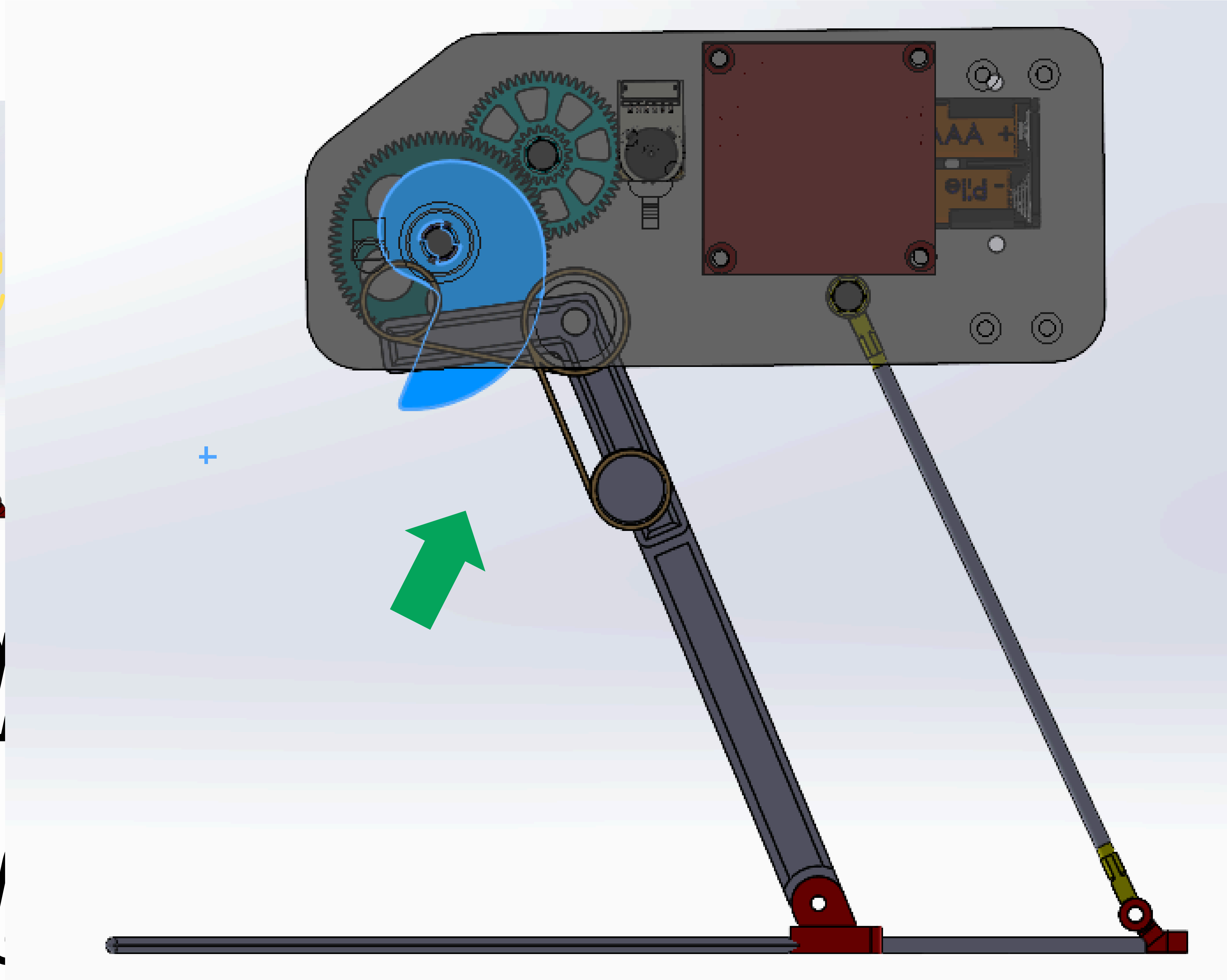
EPFL

First Prototype

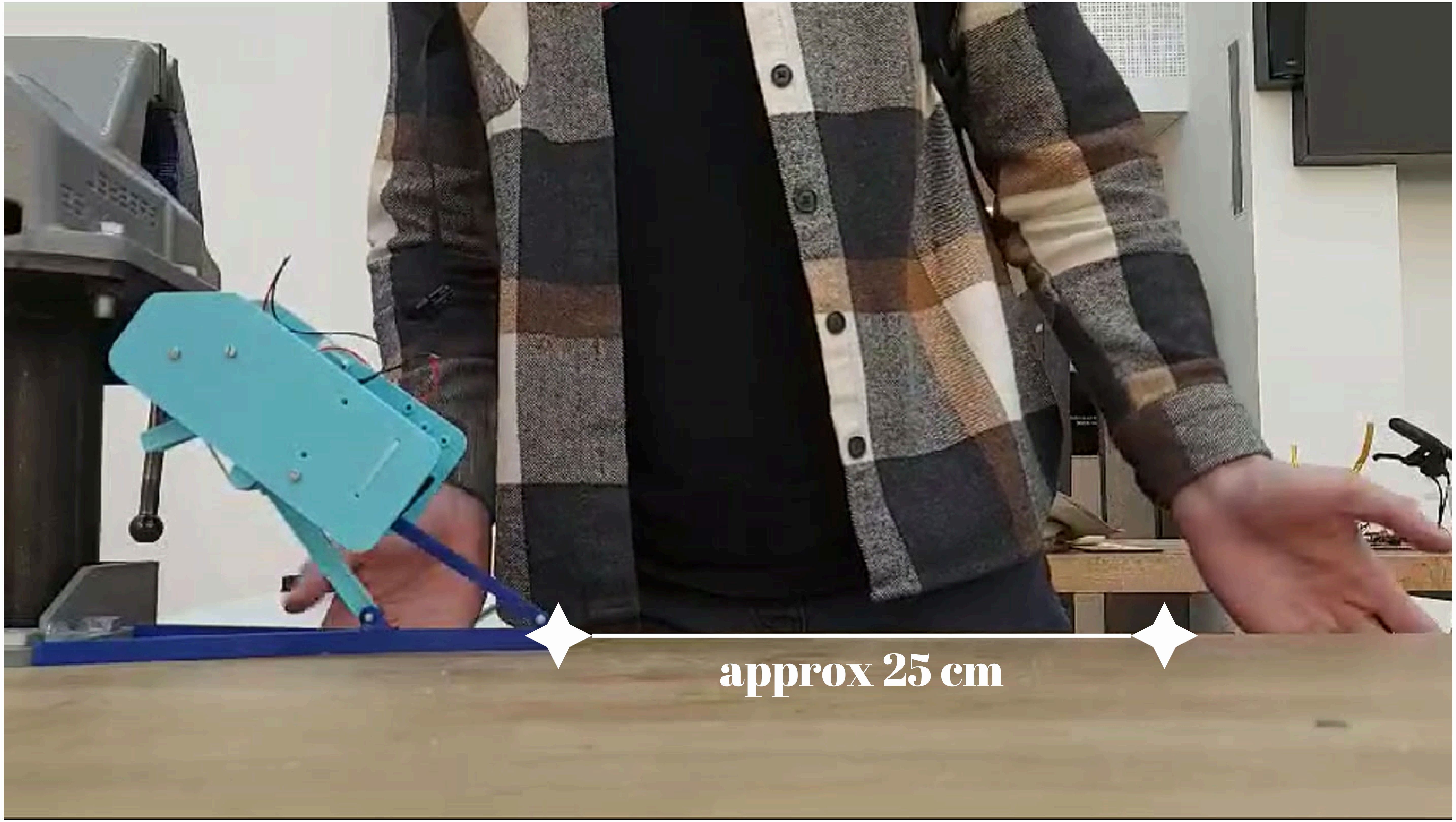


Kena

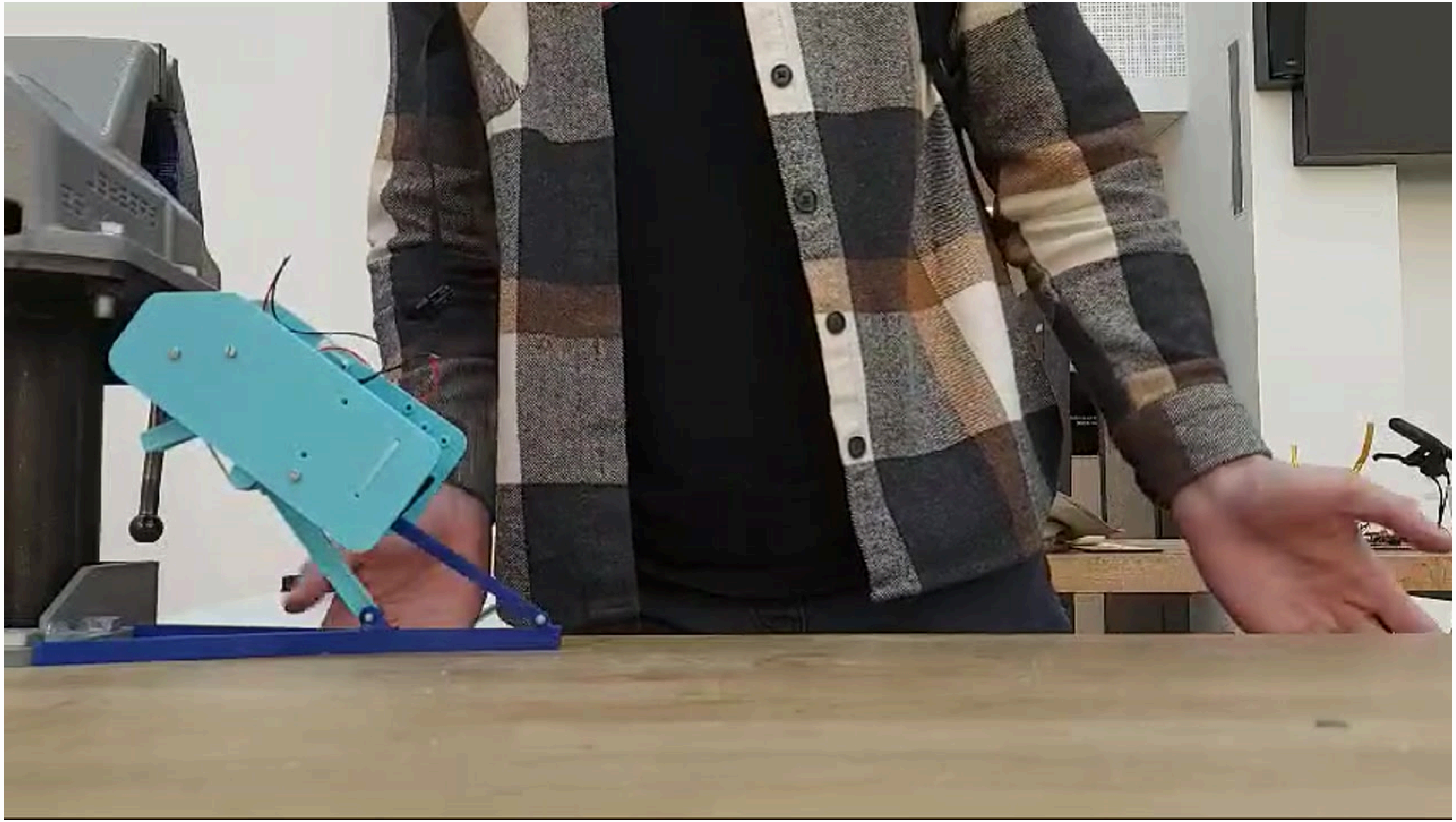
1.



First Prototype

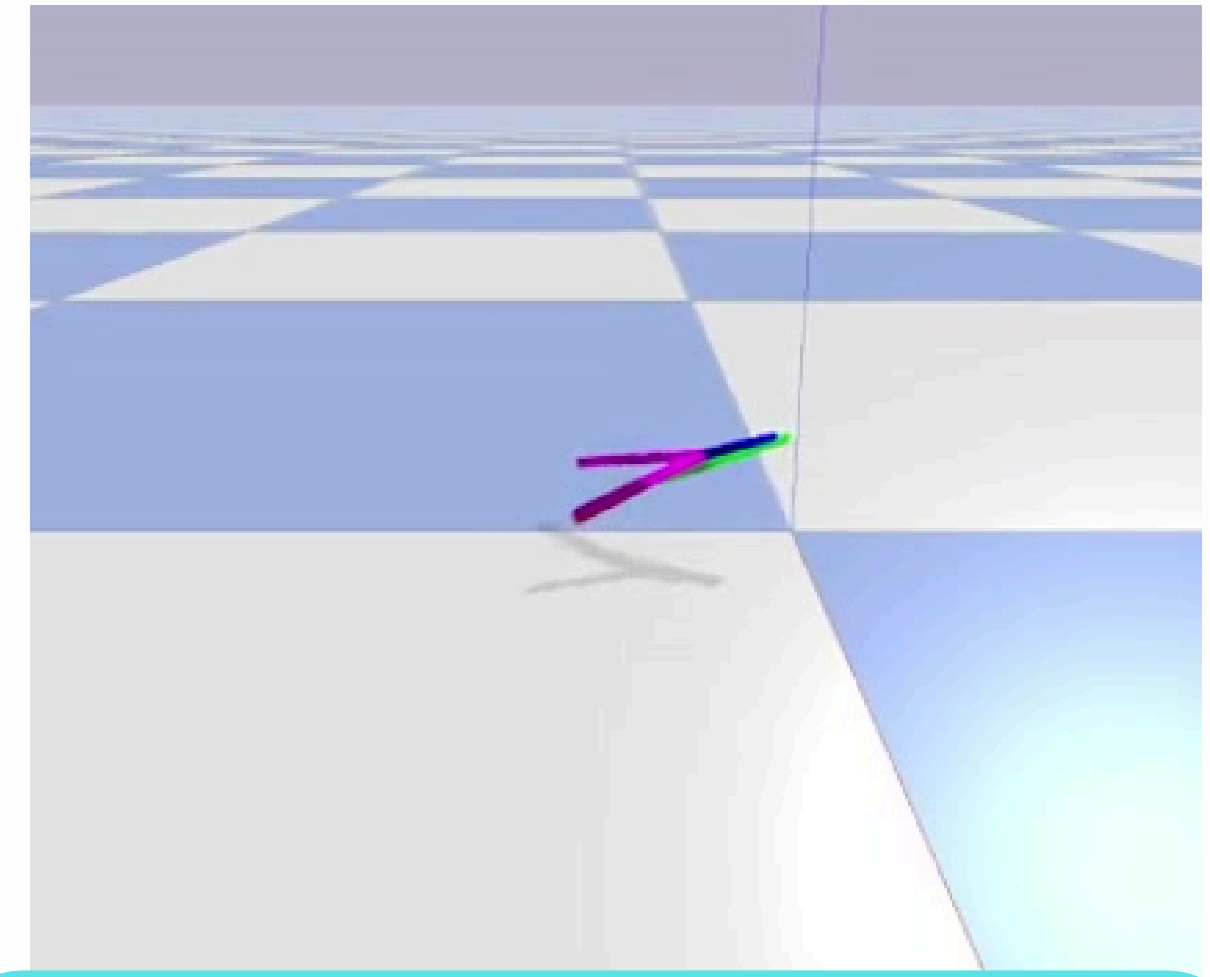


First Prototype



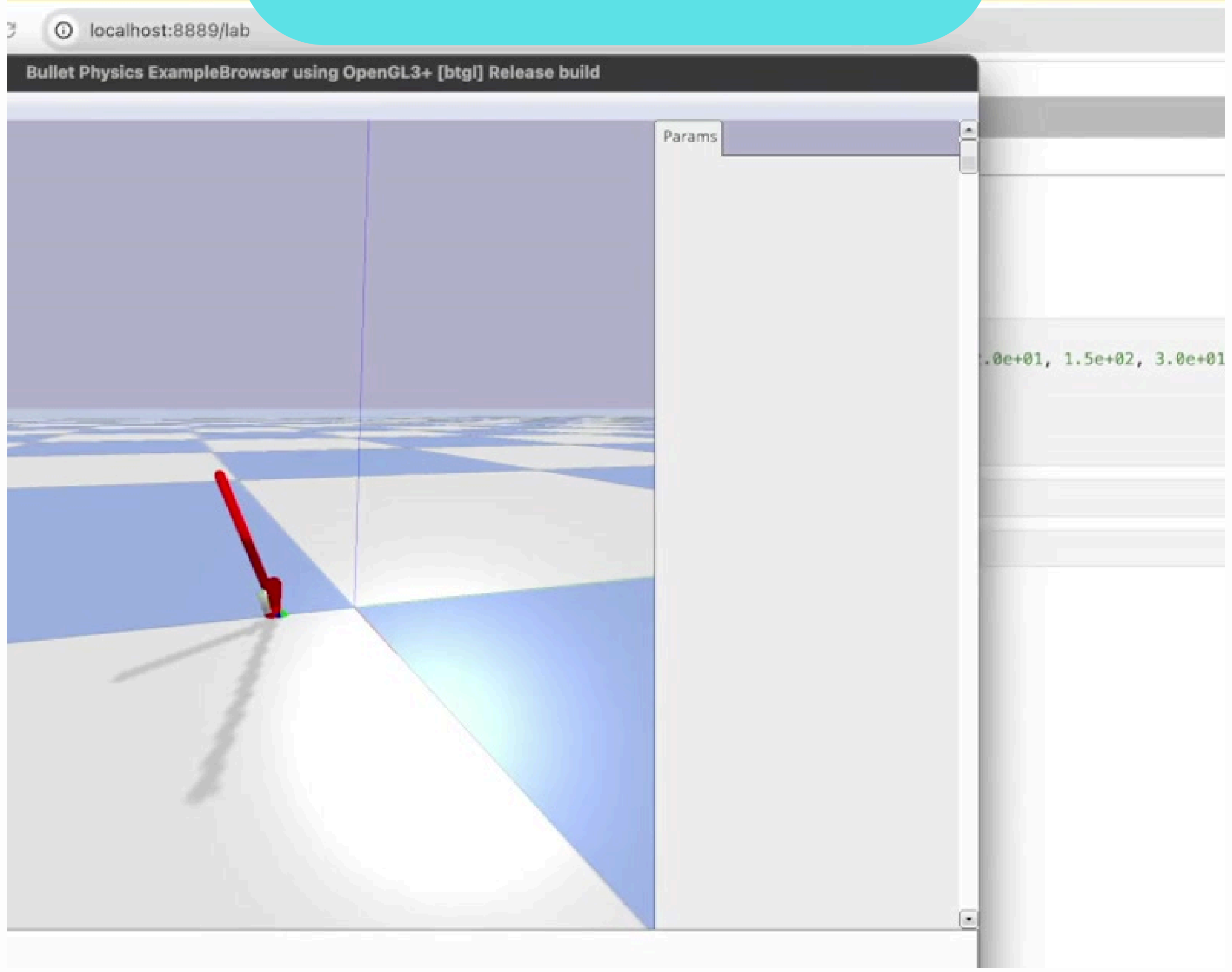
Data generation – simulate the robot

some adventures...



The helicopter one

The flying one



localhost:8889/lab

Bullet Physics ExampleBrowser using OpenGL3+ [btgl] Release build

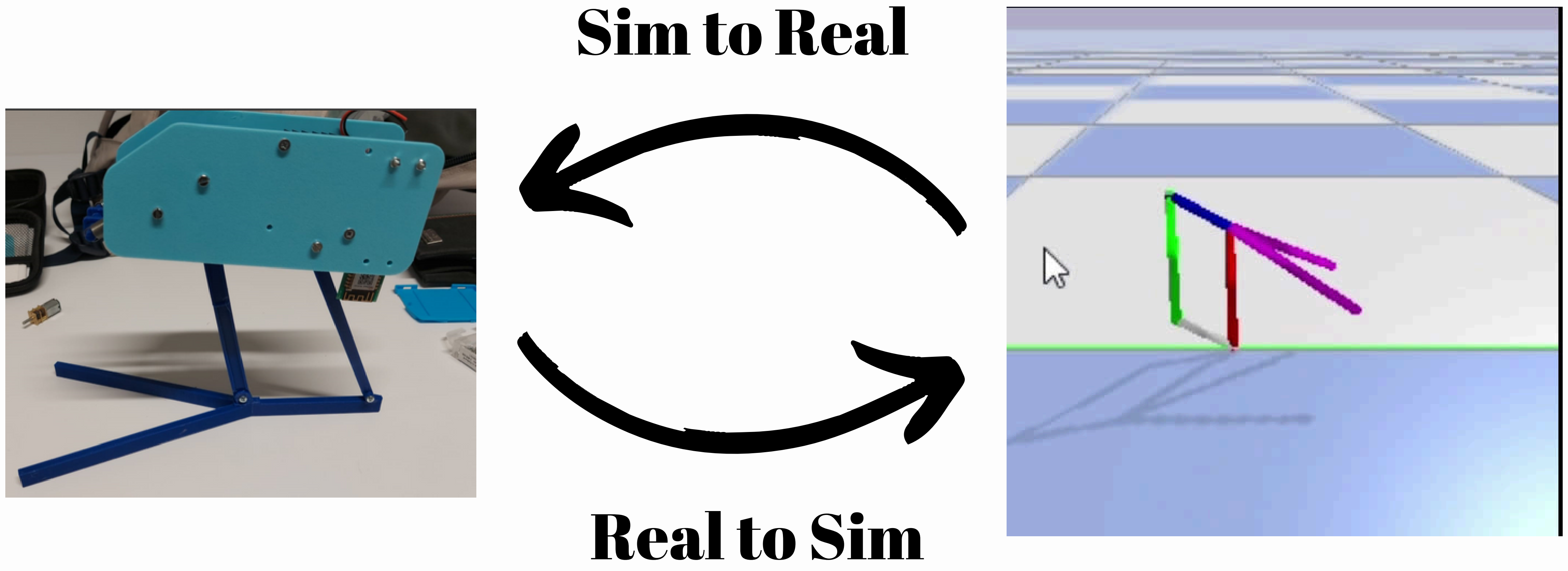
```
.0e+01, 1.5e+02, 3.0e+01
```

keep going....

Data generation – Simulate the robot

Hyperparameters	Values
Lateral Friction	0.8
Spinning Friction	0.1
Rolling Friction	0.01
Coefficient of Restitution	0.7
Joint Damping	0.001

max height 10 cm
max distance 25 cm
energy 0.8 J



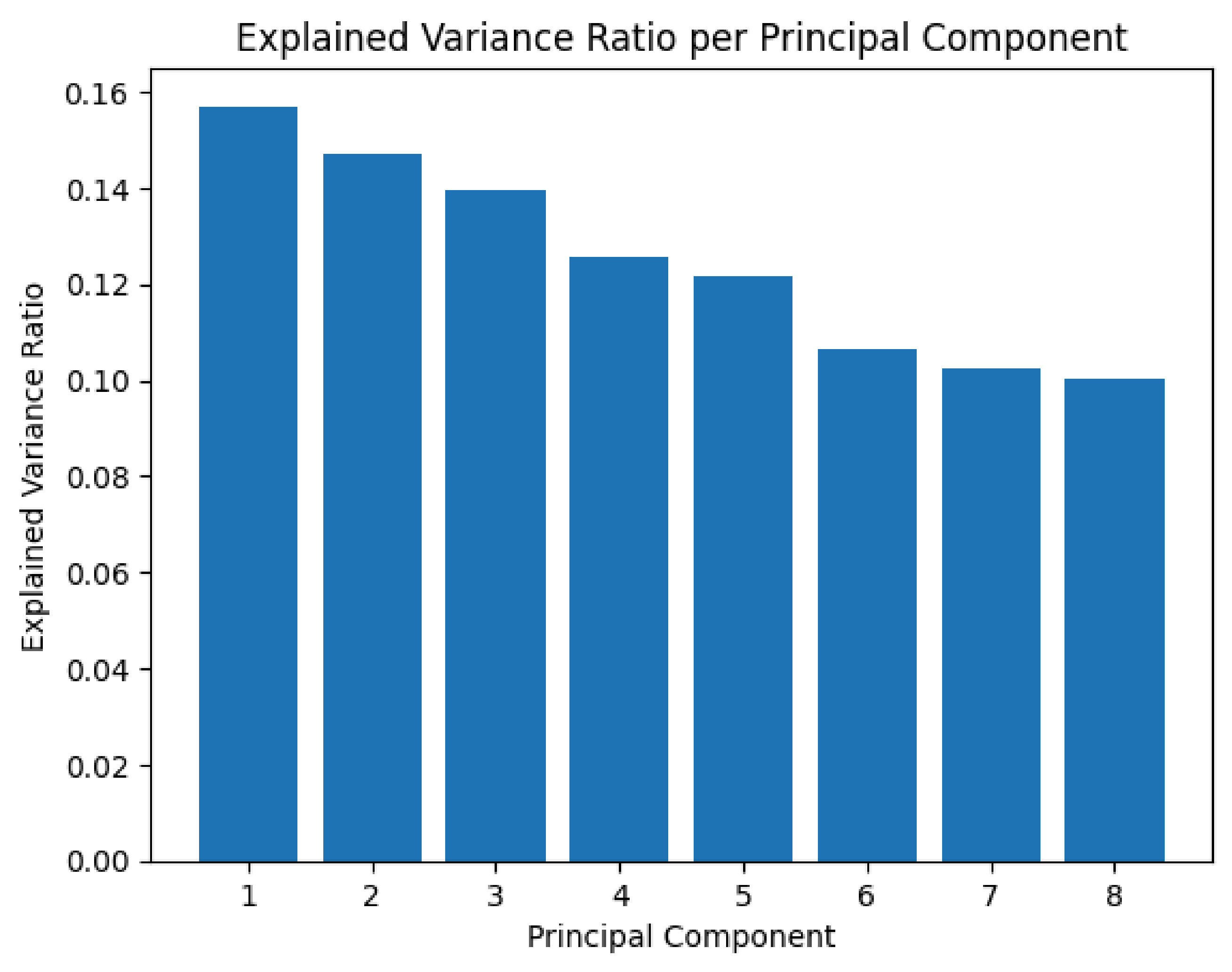
Sim to Real

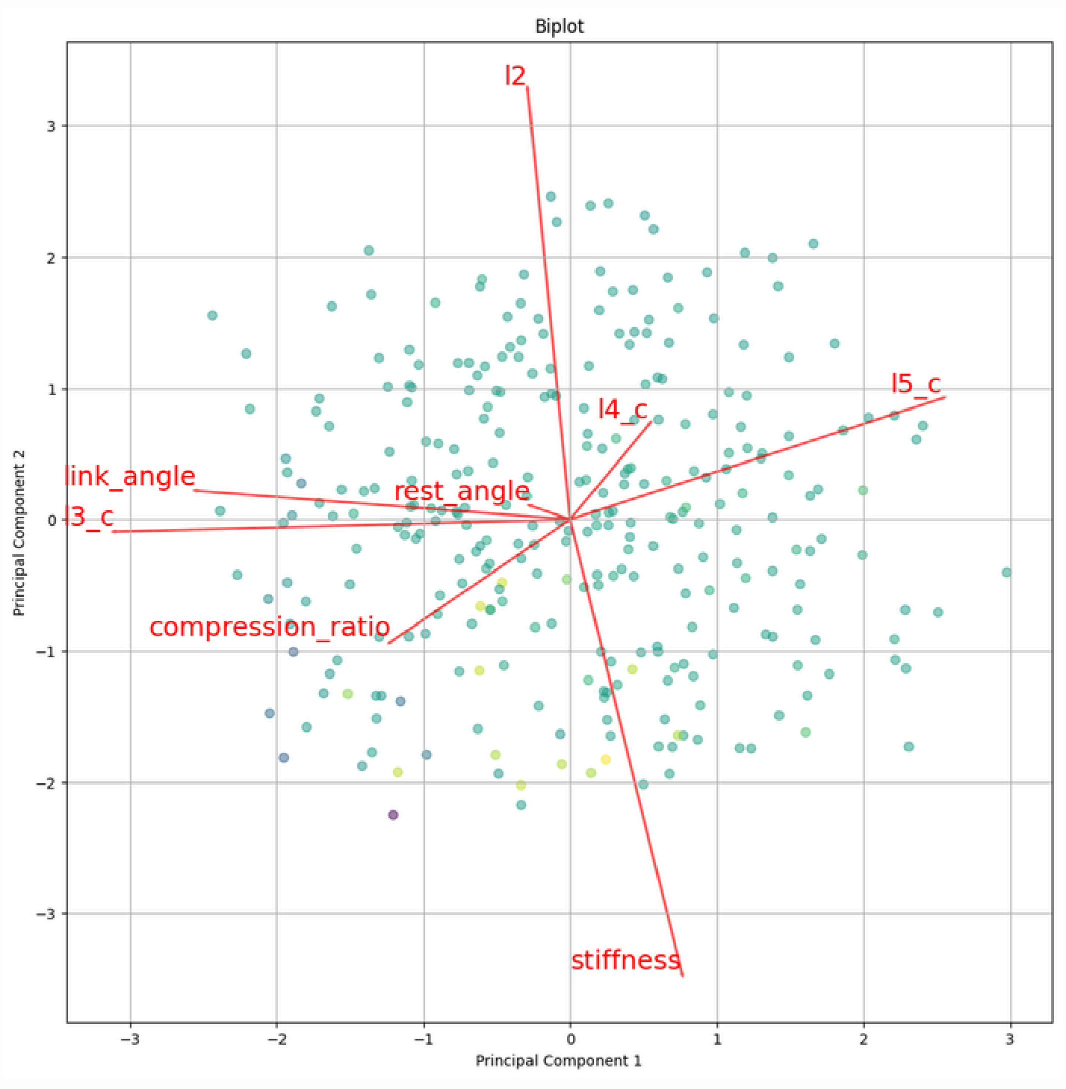
Real to Sim

- Parameters of simulation optimized with the first prototype
- Instability issues are solved with constraint limits
- Reconfigurable in a loop
- Runtime per simulation ~0.3s

Data driven optimisation – PCA

- No principal components that dominates the performance change
- Eigenvector with the smallest eigenvalue explains 10% of variance in performance



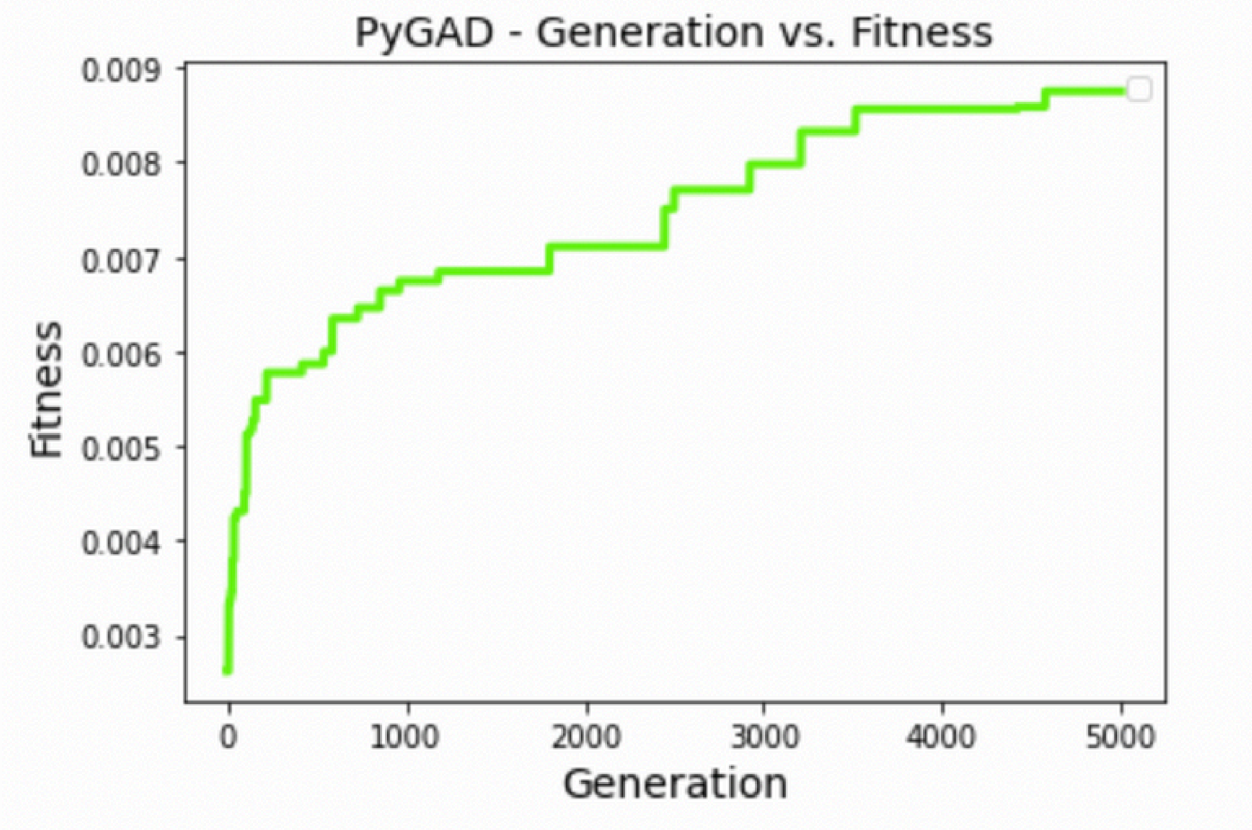


- The biplot shows 'Stiffness' and 'I2' strongly influence PC2, while 'Compression Ratio' and 'Link Angle' impact PC1 negatively.
- Observations cluster based on these variables, revealing distinct patterns in the dataset

Data driven optimisation – Genetic algorithm

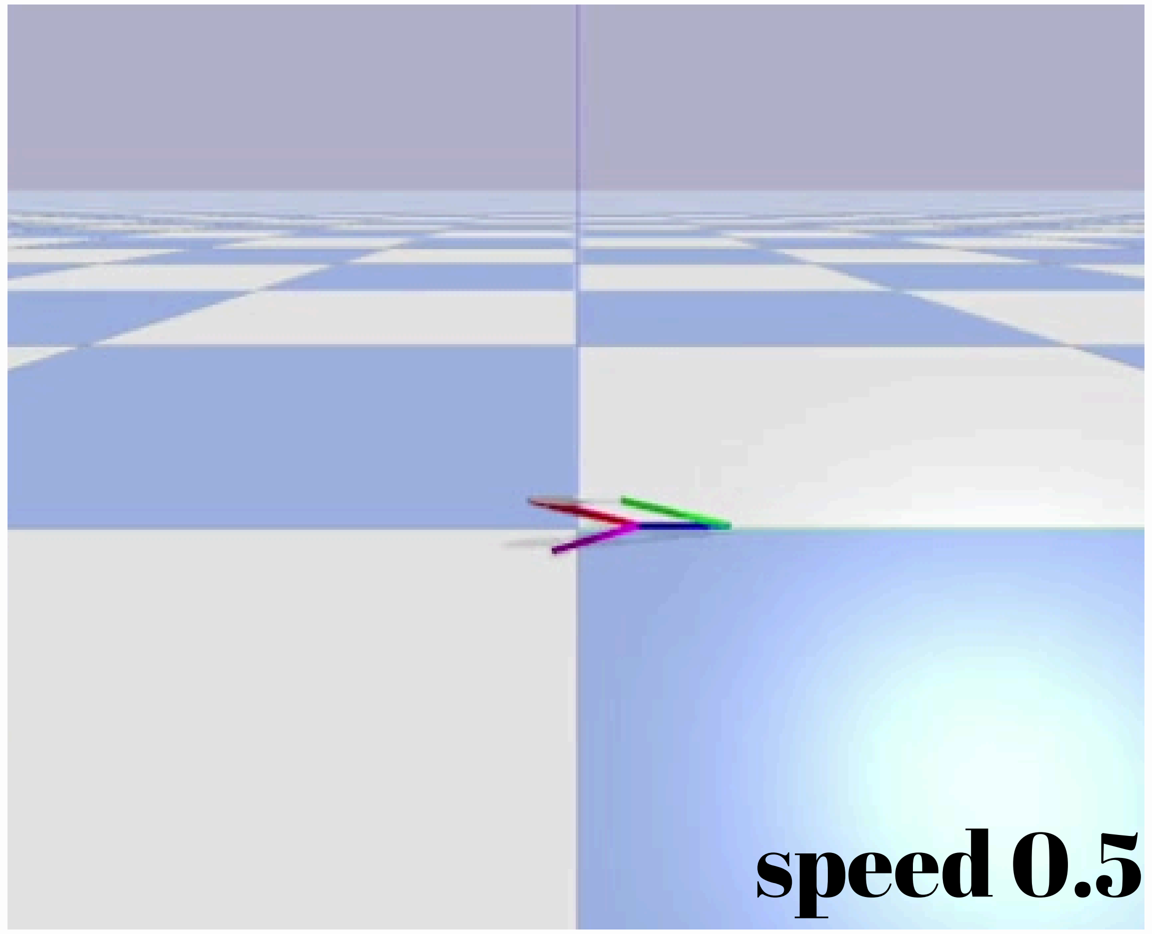
algorithm

- several iterations
- 8 parameters
- up to 5000 generations
- 10 solutions per generations
- crossover and mutations



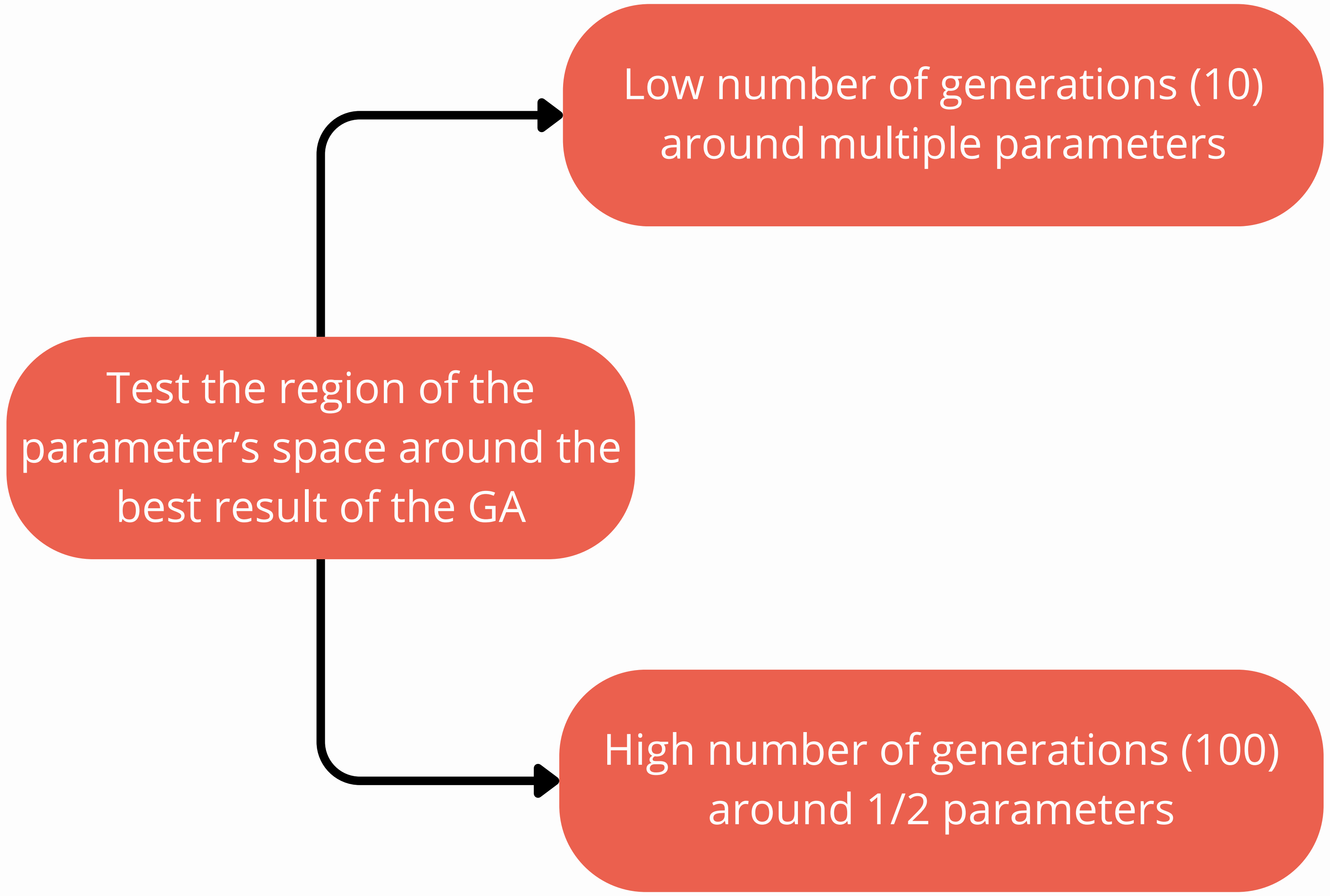
Best solution

max height 38.6 cm
max distance 89.1 cm
energy 0.73 J



Data driven optimisation – Bayesian optimisation

Principle

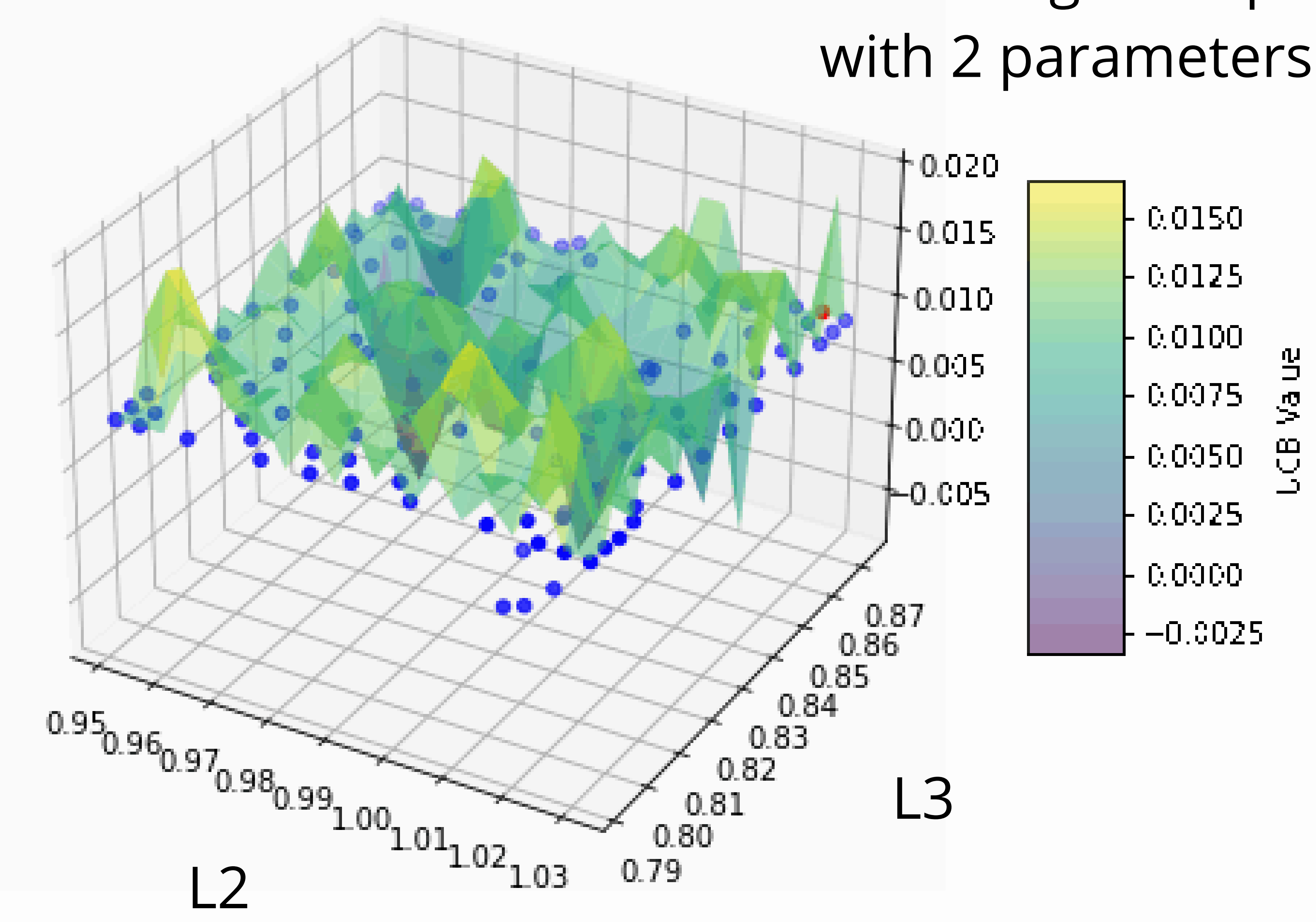


Test the region of the parameter's space around the best result of the GA

Low number of generations (10) around multiple parameters

High number of generations (100) around 1/2 parameters

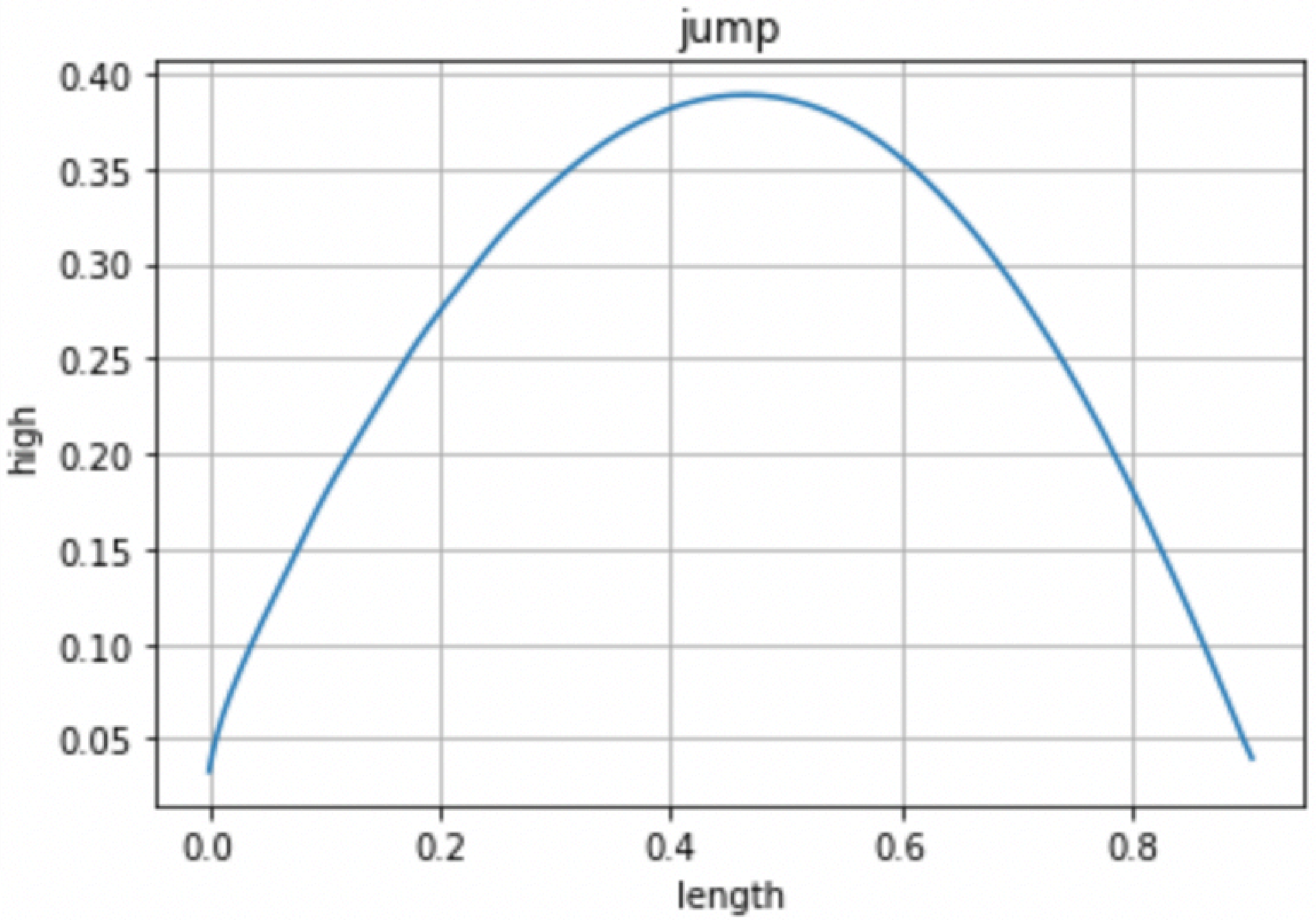
fine tuning example with 2 parameters

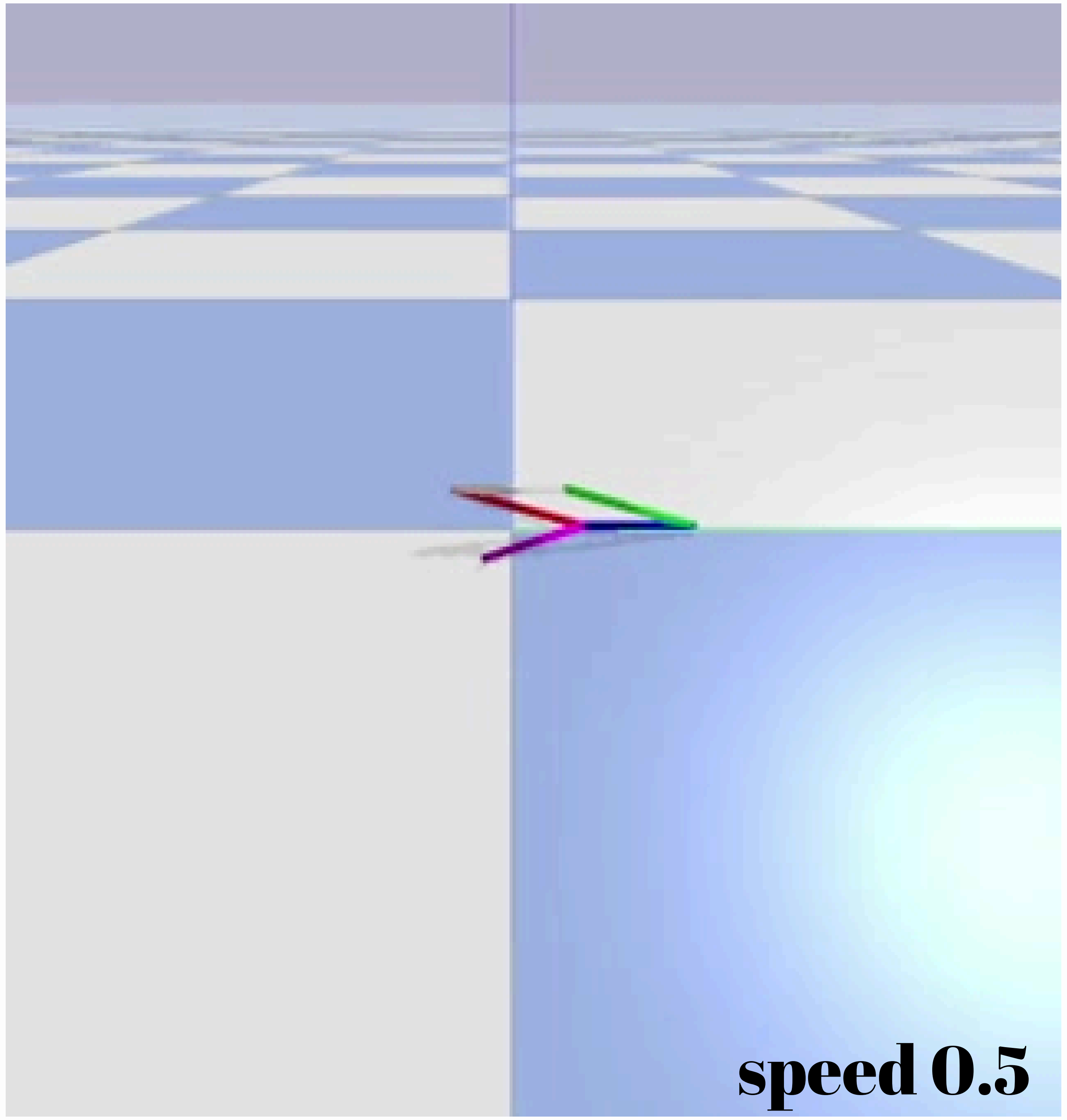


Data driven optimisation – Best result (simulation)

Best solution

max height 38.9 cm
max distance 90.5 cm
energy 0.7191 J





EPFL

Conclusion

...

We optimised a jumping robot to **maximize the jump distance according to it size**
The robot have been simulated using Pybullet
The physical prototype is comparable to the simulated prototype

first result before optimisation	**max height 10 cm max distance 25 cm energy 0.8 J**
With genetic algorithm	**max height 38.6 cm max distance 89.1 cm energy 0.73 J**
With bayesian optimisation	**max height 38.9 cm max distance 90.5 cm energy 0.7191 J**

next steps

Produce some of the best results find with Bayesian

More in-depth analysis of the best results parameters