nature portfolio

https://doi.org/10.1038/s42256-021-00320-3

Supplementary information

Real-world embodied AI through a morphologically adaptive quadruped robot

In the format provided by the authors and unedited

Supplementary Materials

Supplementary Methods

Experiment 1: Adapting in controlled indoor environments. Stop when sensing a change in terrain, reconfigure the morphology, then start walking again.

```
Start with concrete-specialized morphology (femur 50mm, tibia 20mm)
repeat
    Take one step/leg forward, measuring terrain characteristics
    if Non-optimal terrain for current morphology detected then
        Stop walking
        Reconfigure to optimal morphology for detected terrain
    end
until 16 steps/leg has been walked in total;
```

The robot is initially positioned so that it will take 8 steps on the concrete, before stepping onto the gravel for the last 8 steps.

Experiment 2: Adapting in realistic outdoor environments - Adapt morphology and measure terrain and performance without stopping.

```
Start with initial morphology (femur 0, tibia 0)
Walk for three steps/leg, measuring terrain characteristics
repeat
    Generate predicted map for current terrain from model
    if best predicted neighbor COT > current COT then
        Start changing morphology to best performing neighbor
    end
    repeat
        Take one step/leg
    until new leg lengths are achieved;
    Walk for three steps/leg, measuring terrain and energy efficiency
    Add measured terrain characteristics and COT to data set
    Regenerate model with newly experienced data point
until 96 morphologies tested;
```

The robot is initially positioned on the grass, before walking onto road, then back on grass. It is manually led onto the next terrain type after 32 morphologies have been tested on each terrain section.

Algorithm 1: Sensing terrain roughness with the Intel Realsense RGBD camera.

```
Find 3d plane best approximating the point cloud
Inliers = all points within a 35mm distance of the plane

Square error = 0
for all inliers do
|  Square error += squared distance between current inlier and plane
end

Mean square error = Square error / number of inliers

if More than 30000 inliers: then
|  Roughness = Mean square error
else
|  Roughness = 0
end
```

Standard methods from the PCL library are used for plane extraction and segmentation. The RGBD camera used had problems with flat surfaces without discerning visual features, so a roughness of 0 was assumed when a very low number of points were returned by the sensor. The functionality for roughness sensing is mainly implemented in the pointCloudPlaneFitter.cpp file in the terrain_characterizer repository [60].

Algorithm 2: Sensing terrain hardness with the Optoforce leg sensors.

```
for both front legs do
    for all three axes do
        if current force measurement > 100N then
            discard measurement
        end
    end

    filtered x-axis force = median(last 5 x-axis measurements)
    filtered y-axis force = median(last 5 y-axis measurements)

    if current z-axis force measurement < 0 then
        current z-axis force measurement = 0
    end

    filtered z-axis force = median(last 5 z-axis measurements)
end

hardness = 0
for both front legs do
    for all three axes do
        hardness += max filtered absolute force over last 6 seconds
    end
end
```

The force sensors exhibit some high frequency noise, and a median filter of size 5 has been used to reduce it. Erroneous values over 100N are also removed due to being above the nominal capacity of the sensors. Due to the construction of the sensors, it sometimes reports negative forces in the z-direction. These are also filtered away, as they do not reflect actual forces. The full datasheet for the sensors has been uploaded to the dyret-documentation repository [51].

Supplementary Tables

Supplementary Table 1: Parameters for the gait controller. * These parameters are linearly scaled as morphology changes, see Supplementary Table 2 for details.

frequency	0.2
lift duration	0.15
p0_x	0.0
p0_y	50.0
p1_x	0.0
p1_y	-80.0
p2_x	0.0
p2_y*	50.0
p2_z*	50.0
p3_x	0.0
p3_y*	-15.0
p3_z*	100.0
p4_x	0.0
p4_y*	-80.0
p4_z*	50.0
wagPhase	0.05
wagAmplitude_x	25.0
wagAmplitude_y	75.0

Supplementary Table 2: Morphologies and spline scaling for all leg length combinations used in our experiments. Optimal and best trade-off morphologies were found while generating our baseline data set.

ID	Femur (mm)	Tibia (mm)	Total	Scaling	Comment
0	0	0	0	100%	Optimal on gravel
1	0	20	20	103%	
2	0	40	40	106%	
3	0	60	60	109%	
4	0	80	80	112%	
5	12.5	0	12.5	103%	
6	12.5	20	32.5	106%	
7	12.5	40	52.5	109%	
8	12.5	60	72.5	113%	
9	12.5	80	92.5	116%	
10	25	0	25	106%	
11	25	20	45	109%	
12	25	40	65	113%	
13	25	60	85	116%	
14	25	80	105	119%	
15	37.5	0	37.5	109%	
16	37.5	20	57.5	113%	Best trade-off for all surfaces
17	37.5	40	77.5	116%	
18	37.5	60	97.5	119%	
19	37.5	80	117.	122%	
20	50	0	50	113%	Optimal on sand
21	50	20	70	116%	Optimal on concrete
22	50	40	90	119%	
23	50	60	110	122%	
24	50	80	130	125%	

Supplementary Table 3: Terrain characteristics of the three materials present in the indoor terrain boxes, shown with median and interquartile range.

	Roughness (mm^2)			Hardness (N)		
	Median	IQR	Max	Median	IQR	Max
Concrete	5.1	7.6	14.0	135.5	13.3	155.2
Sand	25.2	18.5	63.9	61.3	8.2	82.8
Gravel	35.7	12.8	58.1	85.7	24.9	119.7

Supplementary Table 4: Values and ranges for all sensors used on the robot. * This is the nominal range, and the sensor can return values outside of this, depending on calibration.

Sensor	Measurement	Rate (Hz)	Unit	Value
Servo (x 12)	Current	100	A	[-9.2, 9.2]
Optoforce leg sensors (x 4)	Force x-direction	100	N	*[-40, 40]
	Force y-direction	100	N	*[-40, 40]
	Force z-direction	100	N	*[0, 80]
Realsense depth camera	3D point cloud	6	mm	
Motion Capture rig	Position, x-direction	100	m	[0, 8]
	Position, y-direction	100	m	[0, 8]
	Position, z-direction	100	m	[0, 6]
RTK GPS	Position, x-direction	8	m	
	Position, y-direction	8	m	

Supplementary Table 5: Regression parameters for models trained on the indoor data set. Linear regression from the Scikit Learn library was used to generate and deploy the models. Femur and tibia are measured in mm, roughness in mm^2, and hardness in N.

Femur	Tibia	independent term	roughness	hardness	$roughness^2$	roughness * hardness	$hardness^2$
0	0	23.111	-0.292	0.124	0.011	-0.001	0
0	20	101.816	-5.604	-0.392	0.085	0.023	0
0	40	-34.391	2.236	0.903	-0.028	-0.012	-0.03
0	60	-34.656	2.999	0.513	-0.049	-0.005	-0.001
0	80	42.523	-0.835	-0.218	0.022	0.003	0.001
12.5	0	-93.561	4.471	1.636	-0.065	-0.017	-0.006
12.5	20	51.254	-1.030	-0.495	0.015	0.008	0.002
12.5	40	64.163	-0.607	-1.038	-0.009	0.018	0.005
12.5	60	-47.784	1.681	1.251	-0.019	-0.007	-0.006
12.5	80	-160.639	3.238	2.799	-0.017	-0.021	-0.011
25	0	-62.823	2.991	1.040	-0.034	-0.012	-0.003
25	20	11.706	-0.204	0.296	-0.009	0.009	-0.002
25	40	-98.220	4.566	1.426	-0.045	-0.021	-0.004
25	60	126.911	-2.581	-1.117	0.008	0.021	0.002
25	80	77.427	-1.355	-0.713	0.008	0.010	0.002
37.5	0	-27.108	1.801	0.626	-0.023	-0.006	-0.002
37.5	20	1.300	1.180	0.223	-0.011	-0.004	-0.001
37.5	40	29.676	0.633	-0.405	-0.009	0.002	0.002
37.5	60	-18.316	0.874	0.625	-0.009	-0.001	-0.002
37.5	80	-32.512	1.157	0.914	-0.008	-0.006	-0.004
50	0	23.989	-0.111	-0.065	-0.008	0.009	0
50	20	-34.658	0.788	1.175	-0.005	-0.004	-0.006
50	40	43.179	-0.147	-0.449	-0.003	0.005	0.002
50	60	-55.32	1.509	1.382	-0.014	-0.007	-0.006
50	80	60.060	-1.578	0-022	0-009	0.011	-0.003