Realtime Contact Estimation in the Absence of Force Sensors

Anushree Sabnis

20 January, 2022

Abstract

The following work presents a way to estimate real-time foot contact in a quadruped robot without any external force sensors.

1 Method Description

1.1 Foot Acceleration

The foot velocity is estimated using the data from the joint velocities. The Jacobian matrix , which is a function of the joint positions , relates the end-effector (foot) velocity to the joint velocities as shown in Equation. Equation 1 equation enables us to obtain Cartesian end-effector velocity in Task Space using Joint Space data.

$$\dot{x}_{m*1} = J(q)_{m*n} \cdot \dot{q}_{n*1} \tag{1}$$

Where $\dot{x}_{3*1}$ is the end-effector velocity, $J(q)_{3*3}$ is the Jacobian , $\dot{q}_{3*1}$ are the joint velocities.

To estimate the touchdown and lift-up events during a Quadruped's gait, acceleration of the foot in Z-direction ($\ddot{x}$) is taken into consideration.

1.2 Foot Contact Events

To estimate a foot contact event, the Stance phase of the Gait cycle is further subdivided into a **Contact transition** event and a **Contact Retention** event as shown in 1, as taken from [1]

The Contact Transition Events are characterized by peaks in the acceleration data as seen in figure 2

The Contact Retention events are to be detected using the estimated foot force data.

1.3 Realtime Peak Detection

A peak detection algorithm was required which was able to smoothen the Cartesian Acceleration data as well as reliably detect Peaks at Contact Transition events

A moving average (μ) of acceleration data is calculated :

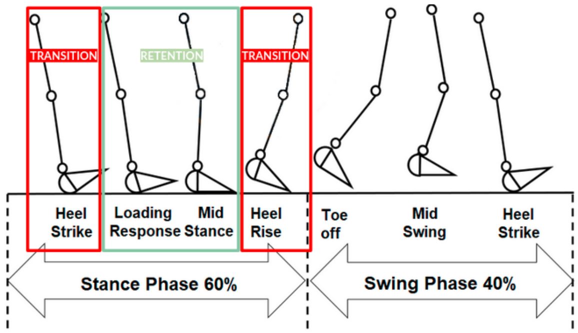

Figure 1: Foot Contact Event with Contact Transition and Retention.

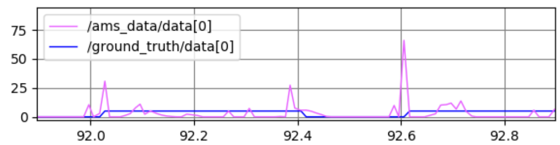

Figure 2: Acceleration Data of Front-Left Leg.

$$\mu_i = (1-\alpha)\cdot\mu_{i-1} + \alpha\cdot(x_n - \mu_{i-1}) \tag{2}$$

The Standard Deviation of this moving average is calculated using a recursive algorithm as mentioned in [2]

$$\begin{aligned} \sigma_i &= (1-\alpha)\cdot(\sigma_{i-1} + \alpha\cdot(\ddot{x}_i - \mu_i)^2) \\ S_i^2 &= \sqrt[2]{\sigma_i} \end{aligned} \tag{3}$$

Peak is detected when the difference between the current data-point and moving average exceeds the standard deviation by a certain threshold, as stated in [3]

$$|\ddot{x} - \mu_i| \geqslant h\cdot S_i \tag{4}$$

2 Simulation Tests

This algorithm has so far been tested in a ROS-Gazebo environment for a simulated quadruped robot. The values of the parameters α and $threshold(h)$ are noted for the following tests

Video logs of the tests can be found here

Test	Alpha (α)	threshold (h)	Type-1 error %
Baseline	0.08	1.7	0.83
4 kg External Load	0.16	1.7	1.46
5 kg External Load	0.16	1.7	0.74
Increase Stance Duration by $\pi/4$	0.08	1.7	2.84
Decrease Stance Duration by $\pi/4$	0.08	1.7	2.29
Trot on 1 degree slope	0.16	2.5	0.21
Trot on 2 degree slope	0.14	2.25	0.35
Trot on 5 degree slope	0.14	2.15	0.64
Reduced frequency by 0.5	0.14	2.15	1.53

2.1 Error Calculation

The metric for testing tuned values of α and $threshold(h)$ is the Type-I error. This is a false-positive contact event i.e. A contact event is detected in the absence of actual contact. The total error is calculated as the proportion of of false-positives over all datapoints.

References

[1] P. Aqueveque, E. Germany Morrison, R. Osorio, and F. Pastene, “Gait segmentation method using a plantar pressure measurement system with custom-made capacitive sensors,” *Sensors*, vol. 20, p. 656, Jan. 2020. DOI: 10.3390/s20030656.

[2] E. W. Weisstein, “Sample variance computation. from mathworld–a wolfram web resource. https://mathworld.wolfram.com/samplevariancecomputation.html,”

[3] G. Palshikar, “Simple algorithms for peak detection in time-series,” Jan. 2009.