Velocity control of a 2D dynamic walking robot

G. van Oort
Impact institute/University of Twente
g.vanoort@ewi.utwente.nl

S. Stramigioli
Impact institute/University of Twente
s.stramigioli@utwente.nl

1 Introduction

In this abstract we introduce velocity control for our 2D dynamic walking robot Dribbel [1] (figure 1) and show that, by ‘closing the loop’, this automatically leads to increased robustness. Takuma and Hosoda [2] also have used active gait adaptation by using step-by-step feedback, but their research was largely based on experimental results, whereas this research is more theoretical.

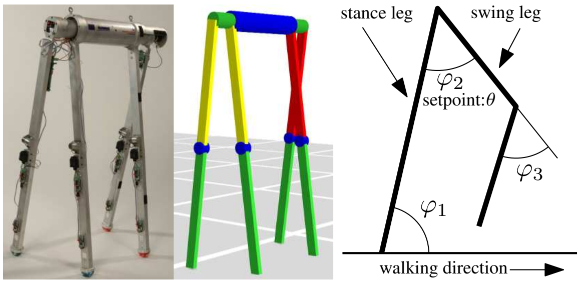

Figure 1: The dynamic walker Dribbel (left), a simulation model (middle) and the degrees of freedom (right). The hip is actuated; the knees are unactuated.

2 The walking algorithm in Dribbel

The control algorithm used in Dribbel and in the corresponding 20-sim simulations is very simple. As soon as the swing foot hits the ground, the (former) stance leg is swung forward. For this movement a very non-stiff P-controller is used in the hip. The setpoint θ can be varied.

For each θ within a certain range, a limit cycle can be found, where the walker has a certain velocity (figure 2). Each limit cycle has a basin of attraction (denoted as BOA(θ)): the set of all states $x = (\varphi_1, \varphi_2, \varphi_3, \dot{\varphi}_1, \dot{\varphi}_2, \dot{\varphi}_3)^T$ for which the walker will converge back to the limit cycle (i.e. it does not fall).

3 Changing velocity

Changing the velocity of the walker can simply be done by changing the setpoint θ to a new value θ_{new}. Given a certain state x of the walker, it is interesting to know which values for θ_{new} can instantaneously be given without making the walker fall. Theoretically, this can be formulated as follows:

$$\text{find the set } \Theta(x) = \{\theta_{new} \in [0,\pi] | x \in \text{BOA}(\theta_{new})\} \quad (1)$$

such that, if the walker is in state x, choosing any $\theta_{new} \in \Theta(x)$ will *not* make the walker fall (assuming no more disturbances, of course). Similarly, a $\Theta(x^*)$ can be found for a reduced state $x^* = (\varphi_2, \dot{\varphi}_1)^T$ at the end of the step.

The algorithm for changing velocity now works as follows. If the desired θ_{new} is in $\Theta(x^*)$, directly change the setpoint to the desired value. If not, select a temporary $\theta_{temp} \in \Theta(x^*)$ which is closest to θ_{new} and wait a few steps before changing the setpoint again.

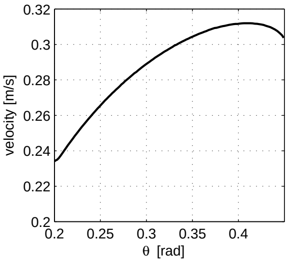

Figure 2: The limit-cycle velocity of the walker for different values of θ. This graph was obtained by simulation.

4 Improving robustness

A nice feature of this strategy is that it automatically increases robustness. Assume that a disturbance puts the walker in a state x^*_{dist}. If $x^*_{dist} \in \text{BOA}(\theta)$, it is no problem; the walker will automatically converge back to its limit cycle again. If $x^*_{dist} \notin \text{BOA}(\theta)$, then, using the same strategy, a temporary $\theta_{temp} \in \Theta(x^*_{dist})$ can be chosen which is close to the original θ but does not make the walker fall.

5 Implementation in Dribbel

A variant of the algorithm described above has already been implemented in Dribbel. As $\dot{\varphi}_1$ cannot be measured in Dribbel (it lacks angular sensors in the feet), the set Θ is approximated by a range $[\theta_{min}, \theta_{max}]$ which is dependent only on the walking velocity measured during the previous step $v_{\text{last-step}} = (2\ell \sin\frac{\varphi_2}{2})/t_{step}$ (note that the step time t_{step} is related to the angular velocity of the stance leg $\dot{\varphi}_1$). Tests showed that with this algorithm it is indeed possible to change the velocity without making the robot fall. Investigating how the robustness is affected is planned for the near future.

Acknowledgements

This research topic was initiated by Ester Guidi, who also implemented the velocity control algorith in Dribbel. Thanks also to Edwin Dertien for his help on the research.

References

[1] E. C. Dertien. “Realisation of an energy-efficient walking robot”, Master’s thesis no. 022CE2005, University of Twente, June 2005.

[2] T. Takuma, K. Hosoda. “Controlling the walking period of a pneumatic muscle walker”, Int. J. Robotic Research, Vol. 25, No. 9, p. 861-866, 2006.