Proceedings of CLAWAR 2011: the 14th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines, Paris, France, 6–8 september 2011

Adaptive Walking Gait for Locomotion on Terrain with Non-uniform Slope

Carlos Neves* and Rodrigo Ventura** and João Sequeira***

Institute for Systems and Robotics
Instituto Superior Técnico
1049-001 Lisboa, Portugal
* *E-mail: cneves@isr.ist.utl.pt*
** *Visiting Scholar, Computer Science Department, Carnegie Mellon University*
E-mail: rodrigo.ventura@isr.ist.utl.pt
*** *E-mail: jseq@isr.ist.utl.pt*

The autonomous locomotion of a legged robot presents several challenges, such as stability and sensitivity to the ground slope. Complex sensory systems, accurate physical models, and demanding computational resources are often required to maintain balance, that together with precise servo control leads to high energy consumption levels. This paper addresses the problem of controlling the locomotion of a humanoid robot, using measurements from inertial sensors to create adaptive walking gaits. These gaits are based on the interpolation of walking gait templates, weighted by the estimated ground slope. A supervised learning method was also implemented to perform a faster recognition of the ground slope when in a static pose. This system provides a simple, yet effective approach to humanoid locomotion on sloped terrain, thus freeing computational resources for other tasks. The approach was validated in simulation, using a realistic model of a Bioloid Humanoid platform.

Keywords: Humanoid robots, Legged Locomotion, Extended Kalman Filter.

1. Introduction

One of the challenges in legged locomotion of humanoid robots is to maintain a stable gait, even on a non-uniform slope terrain. Unless the walking gate is appropriately adjusted, once the ground is no longer horizontal, balance is often lost. In humans, walking makes use of our complex muscular system dynamics, such as the heel strike or the leg swing, exploiting the physics of the body, and thus reducing the energy consumption. On robots, this process is hard to mimic, although extensive work on passive walkers can be found in the literature.[1–4] Other common approaches to humanoid locomotion include robots like Honda's P series[5] and ASIMO[6] using ground reference points,[7] like the Zero Moment Point (ZMP),[8] in order to plan trajectories for the center of mass (CoM), from which gaits are adjusted using Inverse Kinematics. However, these methods require complex and precise modelling of the robot in order to be feasible in practice. The use of pre-programmed gaits for known slopes is, therefore, a simpler, alternative solution.

This paper addresses the design of an adaptive walking gait controller by combining previously developed walking gaits, according to the ground slope estimation. Stochastic filtering of inertial sensor measurements is used for this estimation, with a small computational workload. This allows robots to dedicate less computational resources on walking, enabling them to focus on other tasks.

2. Adaptive Walking

In order to create an adaptive walking gait, the controller requires an estimation of the ground slope. An Extended Kalman Filter (EKF) was implemented following the work on Refs. 9, 10 and 11, allowing the estimation of the rotation transform between the World frame and the Bioloid frame. Since this transform is equivalent to the composition of the rotation from the World frame to the ground slope, with the rotation from the robot feet to its body (Robot frame), knowing the robot joint configuration it is possible to recover the ground slope.

This method was validated using a simple balancing controller: placing the robot on an orientable platform, the controller compensates the slope of the platform by actuating on the joints in order to maintain the robot body vertical with respect to the World Frame. The robot was able to maintain its balance on slopes up to 20°, both in simulation and in the real robot.

The adaptive walking gait is based on three handcrafted template gaits:

- gait **H**, to cope with a horizontal ground;
- gait **F**, to cope with a ground plane rotated 15° around an horizontal axis perpendicular to the robot saggital plane (herein called *frontal* slope); and
- gait **L**, to cope with a ground plane rotated 10° around an horizontal axis paralel to the robot saggital plane (*lateral* slope).

Each gait contains a sequence of poses, as servo set points and associated timestamps.

The resulting gait is obtained by interpolating walking gait **H** with either **F** or **L**, depending on the orientation of the current slope. The interpolation is linear, weighted by the amount of slope detected, and is performed in a per-servo and per-timestamp fashion.[a] The position θ_i of servo i for an estimated slope of γ is given by Eq. (1), where m and b are obtained with Eqs. (2) and (3). θ_{ig} and γ_g refer to the servo position and slope of a gait where g is the gait being used.

$$\theta_i = m \cdot \gamma + b \tag{1}$$

$$m = \frac{\theta_{ig} - \theta_{i\mathbf{H}}}{\gamma_g - \gamma_{\mathbf{H}}} \tag{2}$$

$$b = \theta_{ig} - m \cdot \gamma_S \tag{3}$$

In the case of gait **L**, a non-linear correction $\boldsymbol{\rho}$, explicited in Eq. (4), is to be added to the servos of the leg that needs to be bend (either servos 11, 13 and 15 or 12, 14 and 16) to ensure the robot maintains its balance. θ_{foot} is the position of servo 17, if the right leg is bent, or 18, if the left leg is bent. The servos localization can be seen in Figure 1.

$$\boldsymbol{\rho} = -11.159 \cdot |\theta_{foot}|^2 + 3.855 \cdot |\theta_{foot}| \tag{4}$$

The walking algorithm can be summarized as:

```
estimate initial slope
generate initial walking gait files
execute motion until left single support pose
FOR step from 0 to total_steps:
  execute motion until right single support pose
  estimate slope
  IF abs(estimated_slope - old_slope) > threshold THEN
    generate walking gait file for current slope
  END IF
  execute motion until left single support pose
  estimate slope
  IF abs(estimated_slope - old_slope) > threshold THEN
    generate walking gait file for current slope
  END IF
END FOR
execute motion until double support pose
finish
```

[a]Note that the motion files have to be properly synchronized in time.

The walking gait controller estimates the slope of the ground and generates the corresponding walking gait at each step. Two methods are used to estimate the slope of the ground: (1) the *balancing method*, using the balancing controller described above, and (2) using a *learnt table*, described next. The first method runs the balancing controller until the robot body is aligned with the Frontal plane, everytime the robot is on a static single support pose. The second method uses a lookup table mapping a set of pair (*slope, gait*), to the corresponding positions of the Center of Mass (CoM), as estimated by the EKF. This table is initially learnt using a set of known slopes. Everytime the static single support pose is achieved, the robot uses this table to estimate, by interpolation, the current slope, given the current gait and the estimated CoM position. This results in a correction of the gait. Even if this estimation is not accurate once, the re-estimation of the slope at the next time the single suuport pose is achieved leads the algorithm to converge to the appropriate gait (assuming that there is enough data so that the vertical stance is a fixed point of the table).
For instance, in Table 1 is a line taken from the estimation table, corresponding to using a a gait for 10.0° frontal slope. When the gait causes the robot to fall, the corresponding cell is filled with '*'. In this case, the robot falls down when using this gait for slopes of 0.0°, 2.0°, 4.0° or above 12°. If after taking a step using this gait the estimation results in $-1.36cm < \hat{x}_{CoM} < 0.49cm$, then a linear interpolation is made to estimate the current slope. If $\hat{x}_{CoM} \approx -0.58cm$ the slope remains unchanged at 10.0°.

	0.0°	2.0°	4.0°	8.0°	10.0°	12.0°
10.0°	*	*	*	0.0049m	-0.0058m	-0.0136

The advantage of the learnt table method with respect to the balancing one is that a table lookup is faster than running the balancing controller.
The estimation of the slope is performed by:

```
read servo measurements and update EKF
IF current_gait exists in table
  find lowest recorded CoM value higher than estimated CoM
  find highest recorded CoM value lower than estimated CoM
  use the (current_gait, slope) pairs to obtain slope
    estimation by linear interpolation
ELSE
```

```
  WHILE robot not aligned with Frontal plane:
    correct servo positions
  END WHILE
  write new data in the learnt table
END IF
```

3. Results

The simulations presented here were obtained using the Webots™ software, where a realistic model of the Bioloid Humanoid robot developed by the authors was used. The similarities between the Bioloid robot and the model can be seen in Figure 1.

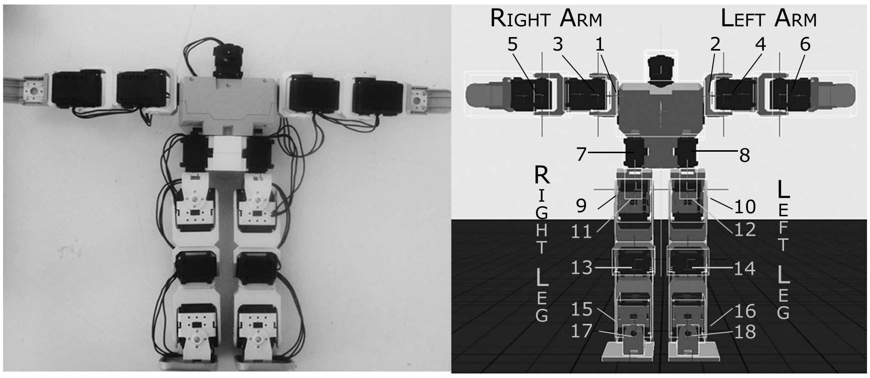
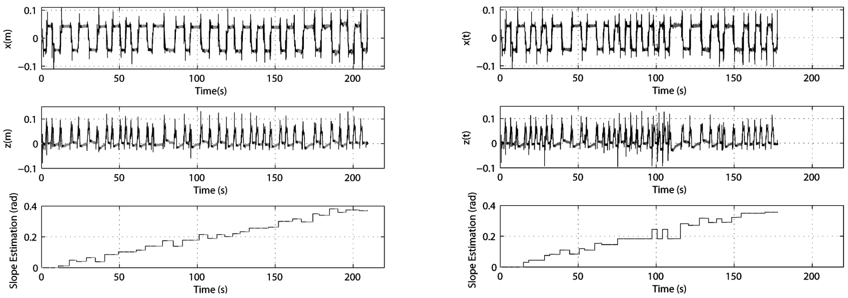


Figure 1. The real Bioloid, in the left image, and the simulated Bioloid, with servo numbering, in the right image

The method of interpolating walking gaits was tested on platforms with either a frontal or a lateral constant slope, with good results on slopes up to 20°. The gaits generated for platforms with rotations around horizontal axes different than these two are very unstable and a better mapping of the corrections that need to be done to the servos is necessary.

The plots in Figure 2 show the results for the adaptive walking gait, in terms of the CoM position (X_{CoM} and Z_{CoM})[b], as the robot travels the test environment depicted in Figure 3, where the slope of the ground increases progressively, from 0° to 20°, in discrete steps of 2°. Even when the CoM estimate is not accurate, but still close to the real value, the generated

[b]The X and Z axis correspond to the horizontal axels, perpendicular and paralel with the robot saggital plane, respectively.



(a) balancing method

(b) learnt table method

Figure 2. Simulation results using the two presented methods: (1) balancing, and (2) learnt table. Top - X_{CoM}; Middle - Z_{CoM}; Bottom - Estimated slope,

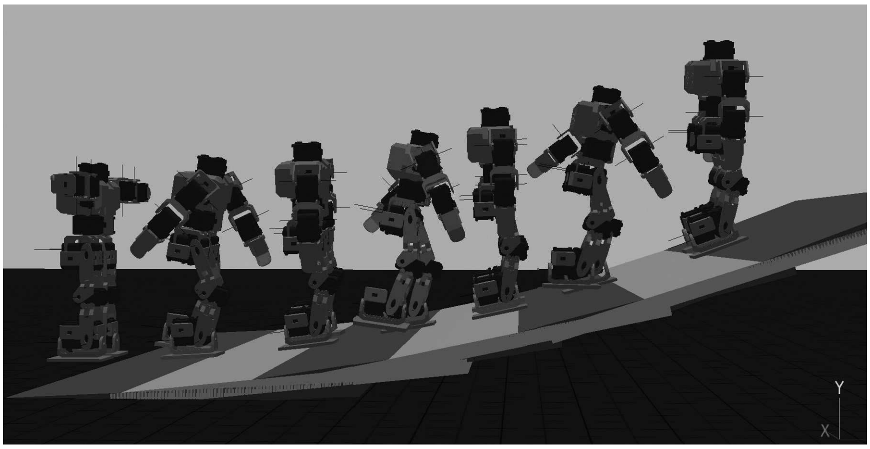
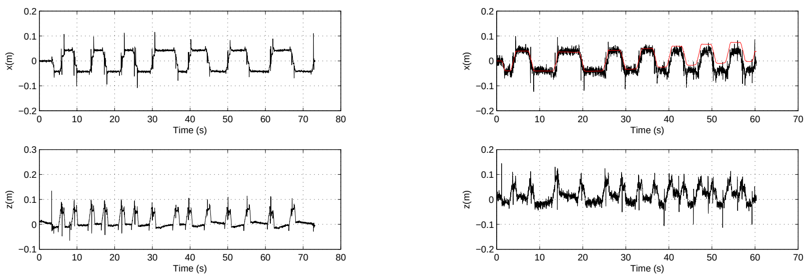


Figure 3. Snapshots taken of the simulation running.

gait allows the robot to maintain balance. The results in Figure 2(a) was obtained using the balancing method, while Figure 2(b) was obtained using the learnt table method. As expected, using the learnt table method allowed the robot to walk faster, taking 11% less time to walk the same distance than using the balancing method.

Robustness of these methods to noisy measurements was evaluated by artificially adding noise to the inertial sensor measurements. This trial is of extreme importance, since the goal is to implement this controller on a Bioloid Humanoid and there will be noise sources not accounted for in the simulations (the plastic frame is not rigid, the ground will not be completely smooth, etc.). In this case, the robot is placed on the floor, with no slope, and walks up to a platform with 3° of slope. The plots in Figure

4(a) show its progress when using normal noise values, and it is clear the transition from the plain floor to the platform at the 31^{st} second, when it takes him longer to balance. As in the previous chapter, it is important to know how the controller handles higher noise rates. In this case, the value of $\sigma^2_{max} = 20\sigma^2_{real}$ was the highest possible to ensure the robot did not stumble or lose its walking direction. The estimation results are shown in Figure 4(b), where it can be seen that are much more unstable than on the previous plots. It is also of note that the estimations tend to stray from the real positions of $\boldsymbol{x}_{CoM}$ after the first 40 seconds of walking.



(a) normal noise variances (b) $\sigma^2_{max} = 20\sigma^2_{real}$

Figure 4. $\boldsymbol{x}_{EKF}$(t) when the robot adjusts to a new slope with normal and increased noise variances

4. Conclusions

The walking gait controller proposed in this paper enables the Bioloid Humanoid robot to have an adaptive walking gait in the presence of non-horizontal terrains, using only inertial sensory data. The simulation results are promising, showing the effectiveness of the proposed methods. The generation of walking gaits by linearly interpolating previously designed gaits is easily applicable to other humanoid robots, since there is no need of a kinematic model of the robot. Future work includes the creation of gaits to handle ground slopes around an arbitrary axis.

Acknowledgments

This work was supported by the FCT (ISR/IST plurianual funding) through the PIDDAC Program funds. We also want to thank Ricardo Oliveira, Limor Schweitzer and the Robosavvy Forum users, who gave invaluable help with the Bioloid platform.

Bibliography

1. S. H. Collins, A bipedal walking robot with efficient and human-like gait, in *Proceedings of IEEE International Conference on Robotics and Automation*, 2005.
2. M. Wisse and A. L. Schwab, First steps in passive dynamic walking, in *Proceedings of the 7th International Conference CLAWAR 2004*, 2004.
3. M. Wisse, C. G. Atkenson and D. K. Kloimwieder, Swing leg retraction helps biped walking stability, in *Proceedings of 2005 5th IEEE-RAS International Conference on Humanoid Robots*, 2005.
4. M. Wisse, *International Journal of Humanoid Robotics* **2**, 459 (2004).
5. K. Hirai, M. Hirose, Y. Haikawa and T. Takenaka, The development of Honda humanoid robot, in *Procedings of the 1998 II-1-1 International Conference on Robotics and Automation*, 1998.
6. H. Masato and T. Tooru, *Honda R&D Tech Rev.* **13**, 1 (2001).
7. M. B. Popovic, A. Goswani and H. Herr, Ground reference points in legged locomotion: Definitions, biological trajectories and control implications, in *International Journal of Robotics Research*, (12)2005.
8. M. Vukobratocić, B. Borovac and D. Šurdilocić, *USC Neuroscience* (2004).
9. E. Lefferts, F. Markley and M. Shuster, *Journal of Guidance, Control and Dynamics* , 417 (1982).
10. N. Trawny and S. I. Roumeliotis, *Indirect Kalman Filter for 3D Attitude Estimation - A Tutorial for Quaternion Algebra*, tech. rep., Department of Computer Science&Engineering - University of Minnesota (March 2005).
11. A. I. Mourikis, N. Trawny, S. I. Roumeliotis, D. M. Helmick and L. Matthies, *The International Journal of Robotics Research* , 737 (2007).