Biped Locomotion on the HOAP-2 robot

Computer Science Master Project

Christian Lathion

10 January 2007

BIOLOGICALLY INSPIRED
ROBOTICS GROUP (BIRG)

Outline

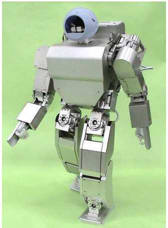
Biped Walking

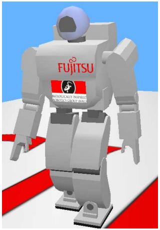
- The goal of this project is to implement biped locomotion on a humanoid robot, based on an existing controller.
- Not an easy task, even if we are used to do it naturally:
  - Nonlinear dynamics of the body (inverted pendulum).
  - Many degrees of freedom.
  - Interactions with the environment.
  - ...
- Main difficulty: achieve stability.

Biped Walking

- Several different methods have been proposed for artificial biped locomotion:
  - *Trajectory-based*: Use offline optimization and constraint satisfaction algorithms.
  - *Heuristics*: Similar technique, but uses heuristic or evolutionary algorithms.
  - *Central Pattern Generators*: Bio-inspired approach, model the nodes – located in the spinal cord – that control vertebrates locomotion.
  - ...
- But still no perfect solution.

The HOAP-2 Robot





- The controller is applied to the HOAP-2 robot:
  - *Humanoid for Open Architecture Platform*
  - Developed by Fujitsu Automation Ltd.
  - 7kg, 50cm
  - 25 degrees of freedom
  - Modelized under Webots

Frequency and phase coupling

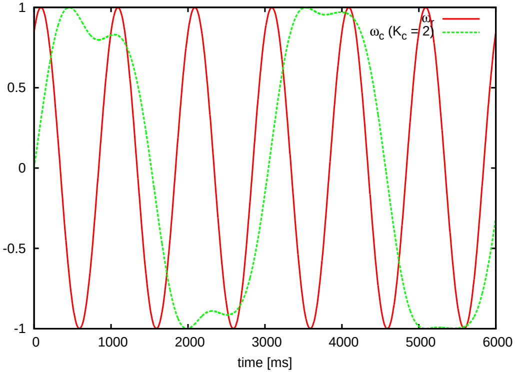
- The controller (ϕ_c) and robot (ϕ_r) phases follow a differential equations system:

$$\dot{\phi}_c = \omega_c + K_c \sin(\phi_r - \phi_c)$$
$$\dot{\phi}_r = \omega_r + K_r \sin(\phi_c - \phi_r)$$

- This synchronizes the controller dynamics with the robot.
- In practice, ϕ_r is obtained through the feet pressure sensors, as the robot natural frequency ω_r and coupling constant K_r are usually unknown.
- Controller phase equation is solved by numerical integration.

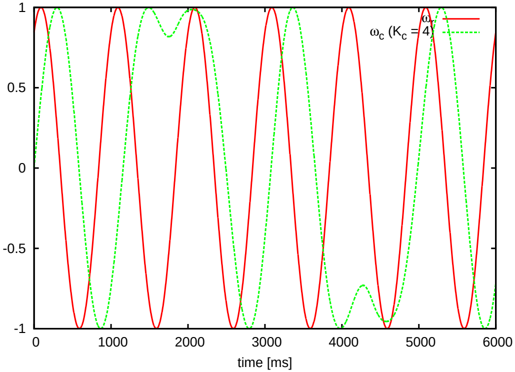
Frequency and phase coupling

- A strong coupling value is necessary to obtain the desired locking effect:
- $K_c = 2.0$



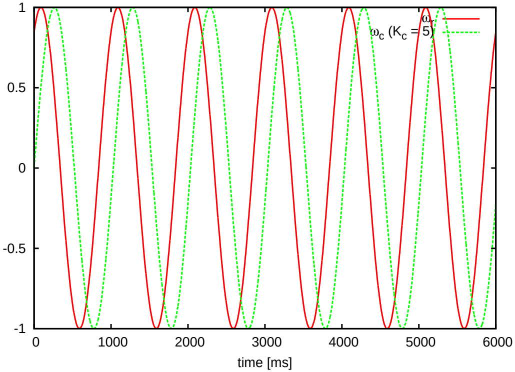
Frequency and phase coupling

- A strong coupling value is necessary to obtain the desired locking effect:
- $K_c = 4.0$



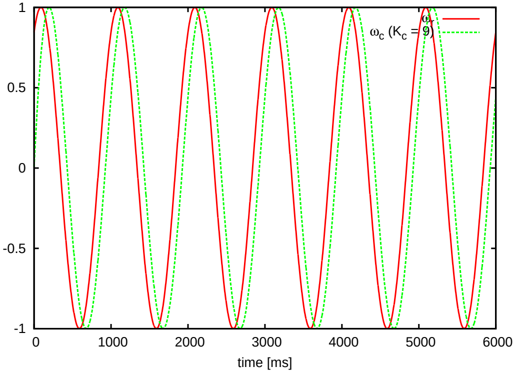
Frequency and phase coupling

- A strong coupling value is necessary to obtain the desired locking effect:
- $K_c = 5.0$



Frequency and phase coupling

- A strong coupling value is necessary to obtain the desired locking effect:
- $K_c = 9.0$



Joint Trajectories

- Trajectories are generated from the controller phase by using simple sinusoidal patterns.
- Divided in *stepping* and *biped walking* sub-movements.

$$
\begin{aligned}
\theta^d_{hip_r}(\phi_c) &= A_{hip_r} \sin\left(\phi^1_c\right) \\
\theta^d_{ankle_r}(\phi_c) &= A_{ankle_r} \sin\left(\phi^1_c - \frac{\pi}{4}\right) \\
\theta^d_{hip_p}(\phi_c) &= A_p \sin\left(\phi^1_c\right) + A_{hip_s} \sin\left(\phi^2_c\right) + \theta^{res}_{hip_p} \\
\theta^d_{knee_p}(\phi_c) &= -2A_p \sin\left(\phi^1_c\right) + \theta^{res}_{knee_p} \\
\theta^d_{ankle_p}(\phi_c) &= A_p \sin\left(\phi^1_c\right) - A_{ankle_s} \sin\left(\phi^2_c\right) + \theta^{res}_{ankle_p}
\end{aligned}
$$

Joint Trajectories

- Trajectories are generated from the controller phase by using simple sinusoidal patterns.
- Divided in *stepping* and *biped walking* sub-movements.

$$\theta^d_{hip_r}(\phi_c) = A_{hip_r} \sin\left(\phi^1_c\right)$$

$$\theta^d_{ankle_r}(\phi_c) = A_{ankle_r} \sin\left(\phi^1_c - \frac{\pi}{4}\right)$$

$$\theta^d_{hip_p}(\phi_c) = A_p \sin\left(\phi^1_c\right) + A_{hip_s} \sin\left(\phi^2_c\right) + \theta^{res}_{hip_p}$$

$$\theta^d_{knee_p}(\phi_c) = -2A_p \sin\left(\phi^1_c\right) + \theta^{res}_{knee_p}$$

$$\theta^d_{ankle_p}(\phi_c) = A_p \sin\left(\phi^1_c\right) - A_{ankle_s} \sin\left(\phi^2_c\right) + \theta^{res}_{ankle_p}$$

Joint Trajectories

- Trajectories are generated from the controller phase by using simple sinusoidal patterns.
- Divided in *stepping* and *biped walking* sub-movements.

$$
\begin{aligned}
\theta^d_{hip_r}(\phi_c) &= A_{hip_r} \sin\left(\phi^1_c\right) \\
\theta^d_{ankle_r}(\phi_c) &= A_{ankle_r} \sin\left(\phi^1_c - \frac{\pi}{4}\right) \\
\theta^d_{hip_p}(\phi_c) &= A_p \sin\left(\phi^1_c\right) + A_{hip_s} \sin\left(\phi^2_c\right) + \theta^{res}_{hip_p} \\
\theta^d_{knee_p}(\phi_c) &= -2A_p \sin\left(\phi^1_c\right) + \theta^{res}_{knee_p} \\
\theta^d_{ankle_p}(\phi_c) &= A_p \sin\left(\phi^1_c\right) - A_{ankle_s} \sin\left(\phi^2_c\right) + \theta^{res}_{ankle_p}
\end{aligned}
$$

Joint Trajectories

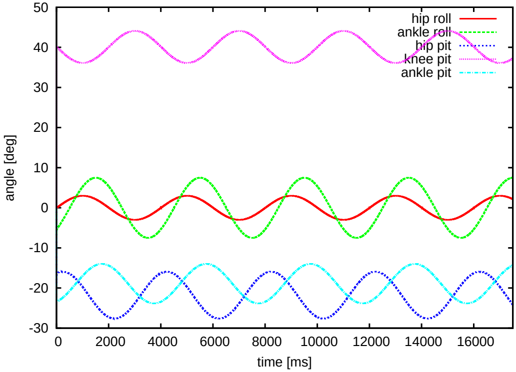
- Limb movements are synchronized by using four different phases:
  - π phase difference for right/left limb movement.
  - $\frac{\pi}{2}$ difference between stepping and walking.
  - $\alpha_i = \left[0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\right]$

$$\dot{\phi}_c^i = \omega_c + K_c \sin\left(\phi_r(\chi) - \phi_c^i + \alpha_i\right)$$

- θ_i^{res} angles define the *rest posture* of the robot joints.
- An additional phase difference of $-\frac{\pi}{4}$ was introduced in the ankle joint equation.
  - Without it, over-oscillations occured, leading to the robot fall.
  - As a side-effect, oscillations of the foot are present during the stance phase.

Joint Trajectories

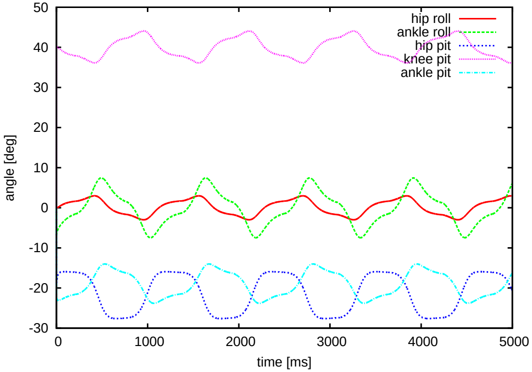
- Coupling changes the shape of the joint trajectories.
  - Simple sinusoidal trajectories are not sufficient to generate the walking pattern.
- The resulting frequency also rises from $\frac{\pi}{2}$ *to* $\simeq \frac{3\pi}{2}$.



(Uncoupled)

Joint Trajectories

- Coupling changes the shape of the joint trajectories.
  - Simple sinusoidal trajectories are not sufficient to generate the walking pattern.
- The resulting frequency also rises from $\frac{\pi}{2}$ *to* $\simeq \frac{3\pi}{2}$.



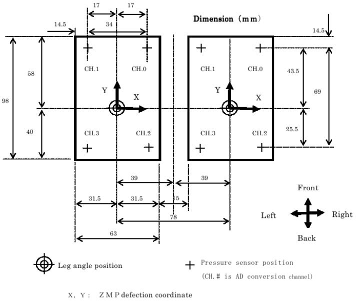
(Coupled)

Feet Pressure Sensors

- Unique sensory input of the controller.
- Four sensors located under each foot.
- Robot phase ϕ_r is detected from the position and velocity of the *center of pressure* x:

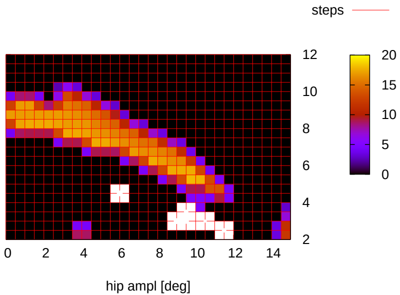
$$\phi_r(\chi) = -\arctan\left(\frac{\dot{x}}{x}\right)$$

- Sensors can also approximate the contact angle with the ground.



Parameters Evaluation

- Small parameter set → an exhaustive search is possible.

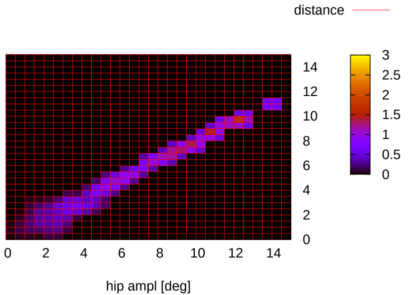


(Stepping)

- Gives a precise view of the working parameters combinations and the resulting performance.
- A_{hip_r} and A_{ankle_r} are linked by a $\simeq$ linear relationship.
  - → The number of parameters can be reduced.

Parameters Evaluation

- Small parameter set → an exhaustive search is possible.



(Walking)

- Gives a precise view of the working parameters combinations and the resulting performance.
- A_{hips} and A_{ankle_s} are linked by a linear relationship.
  - → The number of parameters can be reduced.

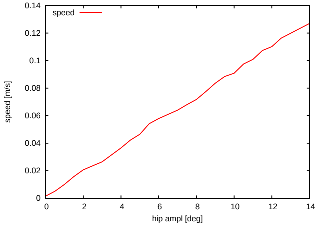
Speed Control

- One of the main limitations of the proposed coupling scheme.
  - The gait frequency cannot be controlled, but depends on the robot dynamics and coupling constant K_c:

$$\omega^* = \frac{K_r \omega_c + K_c \omega_r}{K_c + K_r}$$

  - Target frequency ω_c has almost no influence.
- On HOAP-2, the resulting frequency is $\omega^* \simeq \frac{3\pi}{2}$, which is a bit too fast to be realistic.
- Unfortunately, K_c cannot be decreased without breaking the walking movement.

Speed Control



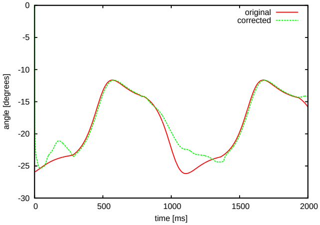
- Consequently, we can only control the step length.
  - Sufficient to control the locomotion speed (linearly).
  - But cannot produce a truly realistic gait.

Stabilization

- The proposed controller doesn't provide a real stabilization mechanism.
- This can be an issue in case of perturbances.
- A first and simple way of minimizing the robot oscillations is to add arm balancing.
    - Opposed to the leg displacement.
    - Also helps to keep a straight displacement direction.

Stabilization

- Foot placement is crucial to obtain a stable gait.
- Adding supplementary control on the ankle joints can give a better contact with the ground.



- We used a PID controller, but results were not as good as expected.
  - Tendency to "break" the joints coordination.
  - The whole controller should be adapted.

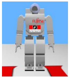
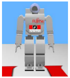
Performance

- Top achieved speed: $\sim 0.15 m/s$ (0.54 km/h).
- Perfect stability when unperturbed ($>$ 5 min continuous walk).
- Quite realistic gait (straight posture) even if accelerated.
- But sagittal plane oscillations and undesired direction changes appear as speed increases.
- 

Resistance to perturbations

- Many perturbations can arise out of the perfect simulated world:
  - External forces, continuous (e.g. wind) or discrete (e.g. a shock).
  - Ground irregularities (e.g. bumps, slippy floor).
  - Slope, obstacles.
  - ...
- Adapting to a perturbation can require a complex set of actions (reflexes, posture change etc.).
- The controller should return smoothly into a stable walk cycle.
- While this is natural for a human, robots usually don't perform very well against perturbations.

Resistance to perturbations

- Robustness can only be roughly approximated.
  - Impossible to model all possible perturbations under Webots.
  - The moment when the perturbation occurs is also very important.
- Able to climb small slopes with minor modifications:
  - Shorter step length.
  - More leg lifting.
  - 
- Able to walk on different surfaces.
  - 
- External forces are more problematic.

Pros and Cons of the proposed Controller

- √ Simple and intuitive.
- √ Easy to add new features and capabilities.
- √ Adaptive to its environment.
- √ Generic controller (successful on Sony Qrio, HOAP-2 and custom human-sized robot).

- X Not as flexible as other methods (e.g. CPGs).
  - Only sinusoidal trajectories can be modeled.
  - Global joints synchronization.
- X Doesn't provide a formal design methodology.
  - No automated optimisation tools.
- X Hand tuning is mandatory.

Possible improvements

- Stabilize the gait (should allow a higher displacement speed).
- Direction control (partially implemented).
- Clean start and stop.
- Climbing stairs.
- . . .

Questions?