

Adaptive Four Legged Locomotion Control Based on Nonlinear Dynamical Systems

Giorgio Brambilla, Jonas Buchli, Auke Jan Ijspeert

Biologically Inspired Robotic Group (BIRG),
Ecole Polytechnique Fédérale de Lausanne (EPFL),
Station 14, CH-1015 Lausanne, Switzerland
giorgio@alwaysnet.net, {jonas.buchli, auke.ijspeert}@epfl.ch
<http://birg.epfl.ch/>

Abstract. Dynamical systems have been increasingly studied in the last decade for designing locomotion controllers. They offer several advantages over previous solutions like synchronization, smooth transitions under parameter variation, and robustness. In this paper, we present an adaptive locomotion controller for four-legged robots. The controller is composed of a set of coupled nonlinear dynamical systems. Using our controller the robot is capable of adapting its locomotion to the physical properties of the robot, in particular its resonant frequency. Our approach aims at developing an on-line learning system that attempts to minimize the energy necessary for the gait. We have implemented the model both in a simulated physical environment (Webots) and on a Sony Aibo robot. We present a series of experiments which demonstrate how the controller can tune its frequency to the resonant frequency of the robot, and modify it when the weight of the robot is changed.

1 Introduction

Nonlinear dynamics is ubiquitous in the physical and in the biological world. Nonlinear dynamics theory has provided us with new tools to understand complex phenomena that were difficult to explain before. It can be used to model competition in predator-prey systems, emergent behavior in collective systems, growth of biological organisms, the production of rhythmic patterns in the heart [10] and in the spinal cord for locomotion [9, 7], to name a few examples.

In this article, we explore how a nonlinear dynamical system implemented as a system of coupled oscillators can be designed (1) to control walking gaits of compliant four-legged robots, and (2) to continuously tune important parameters such as the frequency of oscillations to (possibly time-varying) properties of the body. In particular, we aim at designing adaptive controllers in which the adaptive process is embedded in the dynamical system (i.e. expressed as differential equations) rather than in a separate learning or optimization algorithm.

To produce locomotion, a controller must be capable of generating a rhythmic and coordinated behavior. Nonlinear dynamical systems such as systems of coupled oscillators present several advantages over alternative approaches (e.g. gait

tables or sine-based trajectory generators) to generate gaits for a robot. They allow to harmoniously interact with the environment, they can create limit cycle behavior, they allow smooth modulations of the trajectories, and they make the emergence of new behaviors possible. Moreover, some dynamical systems, like Adaptive Frequency Oscillators (AFOs) [5, 15] are interesting because of their adaptive properties.

The controller we propose is suitable for robots with compliant (i.e. elastic) components. It adapts the walking frequency to resonant frequencies of the robot in order to minimize the amount of energy required to move forward. Hopf oscillators and adaptive frequency oscillators are used as building blocks in the controller. Adaptation is embedded in the dynamical systems, and no external optimization is required. Moreover, adaptation is not a batch process, and the controller adapts its parameters online using proprioceptive signals.

The robot locomotion is based on two different kinds of joints: knees - passively controlled by springs, and hips - actively controlled by servos. The robot swings on the knees behaving like an inverted pendulum. A Hopf oscillator [11] controls each hip. Each of these oscillators is coupled to the other hips for interlimbs gait coordination. Moreover, each hip is coupled in phase to the relative knee. This movement coordination permits to recycle the potential energy of the knee springs to push the robot forward. Furthermore, an Adaptive Frequency Oscillator (AFO) tunes its frequency to the knee oscillations, and this frequency is used for the hip oscillators. This controller has two feedback loops: one permits phase synchronization to proprioceptor signals and the second frequency adaptation.

In the rest of the article, we describe our implementation of this system for a simulated and a real Aibo robot (Section 2). Experiments in simulation (Section 3) and in the real world (Section 4) demonstrate how the system is capable of producing efficient walking gaits that are tuned to the resonant frequency of the robot, and that are continuously adjusted to changing body properties.

2 Adaptive Four Legged Locomotion Model

In this section we describe the main elements of our model. First, we explain what properties make a good locomotion controller and our approach to building one. Second, we describe the mechanical specifications the robot shall satisfy. Third, we introduce our CPG (Central Pattern Generator), a central component in our locomotion model. Eventually, we elaborate on the adaptive equations, which evolve the walking frequency parameter depending on the physical properties of the robot.

2.1 Mechanical Model

We propose a locomotion controller for four legged robots. Every limb has two joints: the upper one (hip) and the lower one (knee). Every hip is actively controlled and is composed of: a servo (actuator), and an encoder (angle sensor). Every lower joint is passively controlled and is composed of: a spring-damper

system (actuator), and an encoder (angle sensor). The controller receives information from the knee and only controls the hip angle to produce locomotion. The body behaves like an inverted pendulum. Our controller does not explicitly deal with: inertial forces, moments of inertia, static and dynamic forces, and contact ground areas. We will call this mechanical framework "elastic locomotion framework." The ideas introduced by Blickhan [2] and by Fukuoka [8] inspired this framework.

Our controller reproduces a walking gait. Walking is easy to implement in this framework, and seems to be the most energy efficient gait compared to trot and gallop [12]. For reasons of stability the terminal limbs are oriented in forward direction, as in Figure 1. In fact, if the limbs were turned backwards, the four legged robot would tumble and fall during the swing phase. A spring-damper rotational system produces in the knee torques proportional to two components: the angle (γ , i.e. the spring torque) and the the angular velocity ($\dot{\gamma}$, i.e. the viscous damping force): $T = k\gamma - d\dot{\gamma}$, where k and d are positive spring and damper coefficients.

2.2 Approach to learning

A good controller to be used in the elastic locomotion framework allows to recycle the potential energy stored in the knee spring during the step cycle, converting it into kinetic energy. Our challenge was to build and tune a controller for a general four legged robot without specific body information, that satisfies the following properties: (1a) knees and hips should move at the same frequency and with a constant phase shift; (1b) the four hips move with a fixed phase shift depending on the gait; and (1c) the knee resonant frequency depends on the weight of the robot as well as on the constant of the knee spring (k). The weight can change during the robot life, and the controller should be able to adapt to it. Our approach is based on nonlinear dynamical systems. There are two kinds of dynamical systems of interest to us: (2a) Hopf oscillators [11, 16], which are interesting because of their limit cycle behavior with the possibility of phase synchronization; and (2b) "Adaptive Frequency Oscillators" (AFOs) [5, 15], which are capable of synchronizing their frequency and phase to an external oscillating signal. In earlier contributions [6] it has been shown that such systems in a feedback loop with the mechanical system can indeed adapt to the resonant frequency of the body. Thus, our controller should satisfy constraints (1a), (1b), and (1c) using dynamical system (2a) and (2b) as building blocks.

2.3 CPG with feedback

Our controller is composed of a fully connected network of four oscillators inspired by animal CPGs [4] (see Figure 1). A continuous arrow means that the signal of the source oscillator at time t is rotated by means of a rotational matrix (R) and summed to the differential equations of the target oscillator. The phase shifts (ρ_{ji}) introduced by the rotation matrix are constant and correspond to the one specified by the walking gait. The coupling values are as in Table 1. Each connection adds a perturbation to the target oscillator that contributes to

the amplification of the signal. We have used a fully connected network because it permits to have a very stable CPG system. In such an architecture, any perturbation is quickly absorbed by the system, and, moreover, we have a reduced influence of the noise. A discontinuous arrow also adds a perturbation, but this time the source is the knee angle value, and it is rotated by a constant angle (ξ). Eqs. 1–2 describe each oscillator in the network. x, y are the state variables describing the oscillator, μ is a parameter which determines the amplitude of oscillations, k is a damping constant, ω is the intrinsic frequency of the oscillator, and a is a global coupling constant, finally to keep the expressions shorter we use $r_i^2 = x_i^2 + y_i^2$.

$$\dot{x}_i = (\mu - kr_i^2)x_i + \omega y_i + a \sum_{\forall j \in I \wedge j \neq i} R_x(\rho_{ji}, x_j(t), y_j(t)) + cR_x(\xi, s_i(t), 0) \quad (1)$$

$$\dot{y}_i = (\mu - kr_i^2)y_i - \omega x_i + a \sum_{\forall j \in I \wedge j \neq i} R_y(\rho_{ji}, x_j(t), y_j(t)) + cR_y(\xi, s_i(t), 0) \quad (2)$$

$$\begin{bmatrix} R_x(\alpha, x, y) \\ R_y(\alpha, x, y) \end{bmatrix} = \begin{bmatrix} x \cos(\alpha) - y \sin(\alpha) \\ x \sin(\alpha) + y \cos(\alpha) \end{bmatrix}$$

$$I = \{LF, RF, LH, RH\}$$

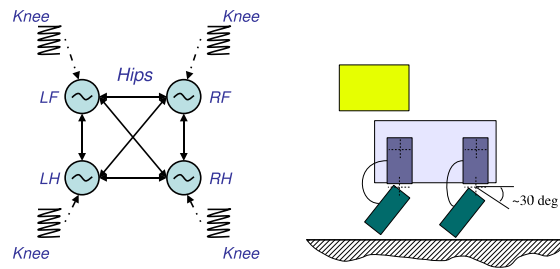


Fig. 1. (Left) Feedback system. In the figure each oscillator corresponds to one of the four hips. The arrows identify the phase coupling. **(Right)** Aibo limbs orientation in the absence of external forces, we have chosen an angle of 30 deg for the knee, and an angle of 0 deg for the hips.

2.4 Adaptive CPG

We have defined a mechanical framework and a walking controller in the previous section. Now we want to reduce the number of parameters to make the CPG frequency adaptive. As explained in Section 2.2, we aim at making the walking frequency adaptive to the resonant properties of the body. As input we use the knee angles. Hence, we compute the signal of the knee angle for every leg. The signal is periodic (but not sinusoidal) and has a similar shape in all legs. The four legs create similar oscillatory signals shifted by a constant phase difference defined by the hip oscillator connections. However the signals slightly differ from the specified phase and shape of the signal. We have to find a means to extract

Table 1. Default parameter values, inter-limbs coupling phase difference, upper limb joints moving range on Aibo platform

<i>parameter</i>	<i>value</i>	ρ	<i>LF</i>	<i>RF</i>	<i>LH</i>	<i>RH</i>
μ	0 (bifurcation point)	<i>LF</i>	0	$-\pi$	$-\frac{3}{2}\pi$	$-\frac{\pi}{2}$
k	0.15	<i>RF</i>	π	0	$-\frac{\pi}{2}$	$\frac{\pi}{2}$
ω	8 [rad/s]	<i>LH</i>	$\frac{3}{2}\pi$	$\frac{\pi}{2}$	0	π
a	1	<i>RH</i>	$\frac{\pi}{2}$	$-\frac{\pi}{2}$	$-\pi$	0
c	5					
ξ	0.2 [rad]					

	<i>MIN ANGLE</i>	<i>MAX ANGLE</i>
Front limbs	0.0 [rad]	0.6 [rad]
Hind limbs	-0.1 [rad]	0.3 [rad]

the frequency information of these signals despite these differences. This can be achieved with an adaptive frequency oscillator (AFO) [5, 15]:

$$\begin{aligned}
 \dot{x} &= (\mu - kr^2)x + \omega_l y + c \sum_{\forall j \in I} R_x(\beta_j, s_j(t), 0) \\
 \dot{y} &= (\mu - kr^2)y - \omega_l x \\
 \dot{\omega}_l &= c\eta \frac{y}{r} \sum_{\forall j \in I} R_x(\beta_j, s_j(t), 0)
 \end{aligned} \tag{3}$$

The AFO (Eq. 3) has three state variables. Compared to a Hopf oscillator, it has one additional state variable used for frequency adaptation and one additional parameter used for learning. The state variable x will synchronize to the input signal. Again, the variable ω_l stands for the frequency [in rad/s]. Due to the additional differential equation the frequency will adapt to one of the frequencies of the input signal (see [15] for further discussion). The parameter η represents the learning rate: a too high learning rate influences the stability of variable ω_l , and a too low learning rate does not permit the adaptation process. In our experiment, we have varied η between 1 and 10.

As a first step, we have tested an intermediate solution where the controller "learns" the main knee oscillation frequency ("**Open Loop**" solution). Our system rotates ($\beta = \{0, -\pi, -\frac{3}{2}\pi, -\frac{\pi}{2}\}$) and sums the knee signals (s_j) so that the walking frequency becomes the most powerful frequency component¹. This signal is used to perturb a frequency adaptive oscillator (Eq. 3). As a result, the oscillator smoothly learns the input frequency and maintains it, reaching a steady state. After a few seconds of transition the dynamical system adjusts its frequency and phase to the one coming from the perturbation signal. We will discuss results showing this properties in Section 3.1.

Eventually, we have rendered the controller adaptive ("**Closed Loop**" solution), as one can see in Figure 2. We have used the value of the state variable

¹ If we sum the signals without rotating them, as the shapes of the signals are comparable and shifted among each other by about 90 deg, they are going to annihilate each other (see Figures 3 and 5). In the following section we propose the experimental results obtained with both "un-rotated" ($\beta = \{0, 0, 0, 0\}$) and "rotated" ($\beta = \{0, -\pi, -\frac{3}{2}\pi, -\frac{\pi}{2}\}$) solutions, to support the choice of the "rotated" solution.

ω_l instead of the parameter ω in the CPG of the walking controller (explained in Section 2.3) making the system adaptive. These ideas permit a smooth adaptation of the walking frequency to the frequencies of the knee. Furthermore, the dynamic formulation of the adaptation by means of dynamical systems allows to adapt the frequency in the case of changes of the body properties or changes in the environment (e.g. ground friction) during the robot life. The online adaptation process also introduces a new feedback loop in the dynamical system. This feedback loop is nonlinear and it is not clear from the outset that it will work. It can however be expected from previous results in simulation [5, 6] and has recently been treated analytically [3]. Even more, the presented experimental results show how well and stable this solution works.

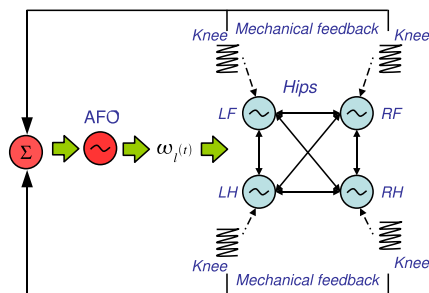


Fig. 2. Closed loop adaptive walking feedback system. In the figure the CPG presented in Section 2.3 is integrated with the AFO and the CPG becomes adaptive.

3 Experimental Results (Simulation)

Webots [13] is an integrated environment for robot simulation, and the physics is simulated using the ODE Library [14]. The robot platform chosen is a Sony Aibo². Aibo has advantages and drawbacks. The advantages are that one can find a detailed model of Aibo in the Webots environment, and can test the controller on a real Aibo robot. However, Aibo is not an optimal platform for our "Elastic Locomotion Framework" because the robot legs do not have a real spring-damper system. Two different strategies have been applied to solve this problem: in the simulation, a knee is controlled by a simulated spring, and, hence, it is the most accurate model of a spring using the Webots simulator. In the real Aibo robot springs are simulated using a PID and an active spring law control to simulate the spring behavior.

All differential equations in our controller are numerically integrated using the Runge-Kutta method with a fixed time integration step. The simulator integrates all the equations at every iteration, using a time step of 0.008 [s] of the virtual simulation time.

² Sony AIBO by Sony Corporation. "Aibo" is a registered trademark of Sony Corporation.

3.1 Open Loop Results

First, we have successfully tested our CPG network, using the default parameters shown in Table 1. Then, we have applied our adaptive component, Equation 3, and plotted the learning curve of the knee signal starting from a range of initial frequency values. In Figure 3, we plot an example of frequency learning using as a perturbing signal the un-rotated (left) and the rotated (right) sum of the knee angles. Comparing the graphs in Figure 3 with the Fourier spectrum (on the right), one can see that the AFO converges to the higher power frequency component and reaches a steady state. Furthermore, rotation provides a wider basin of attraction.

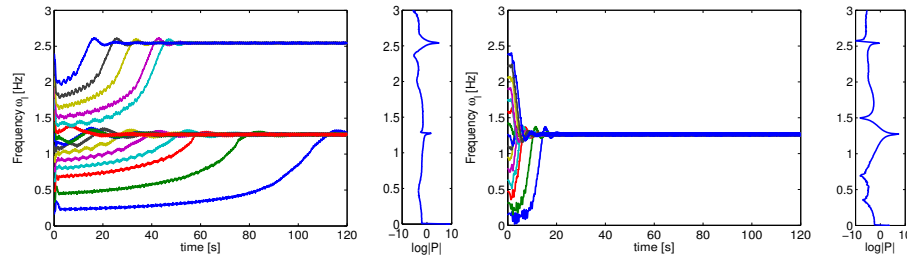


Fig. 3. In the first figure, a set of open-loop experiments (over the same signal) shows the adaptivity properties of an AFO dynamical system ($\eta = 1$). Here the perturbing signal is un-rotated in the left figure and rotated in the right one (Webots). The small graphs are the power spectral density of the un-rotated and rotated signals.

3.2 Closed Loop Results

Second, we have substituted the parameter ω of the hip oscillators with the AFO variable ω_l . Consequently, the CPG parameter ω (walking frequency) has become adaptive and equal to ω_l . This new feature introduces new feedback in the walking controller, as described in Section 2.4. In this system, eventually, knee angles are used for tuning the walking phase and frequency. Moreover, the Hopf oscillators as well as the AFO work at the bifurcation point ($\mu = 0$), which means that in the absence of a stimulus they will not oscillate. Hence, in order to create a locomotion process, at the begin the spring system must be stressed enough in order to create a chain reaction of oscillating stimuli, and must be strong enough to maintain the oscillation without stopping it. To make this possible, it is necessary that one chooses spring-damping parameters that allow the signal to oscillate a few times (in experiments 3 or 4) before being damped out. In other words, the damping parameter d of the knee must be high enough to avoid instability in the dynamics of the spring-mass system, but small enough to permit oscillations. It is expected that every instance implementing the "elastic locomotion framework" has this basic behavior.

We present two experiments: the first is a set of adaptation processes, and the second shows the relation between four different experiments with the same initial conditions but different robot weights.

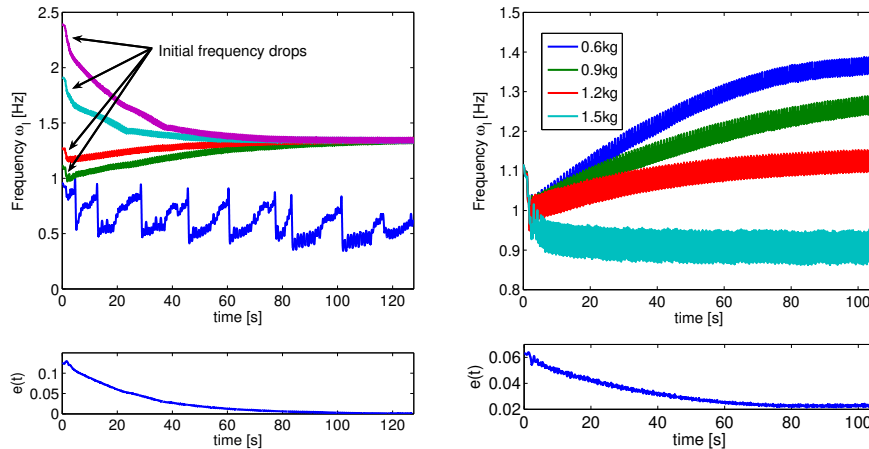


Fig. 4. (Left) closed loop frequency adaptation with different initial frequency but same initial condition ($\eta = 1$). The controller converges if the initial frequency is close to the asymptotic one (Webots). (Right) closed loop frequency adaptation with different robot weights but same initial conditions ($\eta = 1$) (Webots). The two bottom small plot show the average error (see text for definition) between the solution of our controller and the spring-mass oscillating law $\sqrt{\frac{k}{m}}$.

The first plot (Figure 4 left) shows a series of experiments with the same initial conditions (environment, robot) but different initial frequency $\omega_0 = \{0.95, 1.11, 1.27, 1.90, 2.38\}$, in order to demonstrate how the final frequency purely comes from the adaptive process. Furthermore, in the plot the variable ω_l converges to the same asymptotic value. In the plot, the first experiment ($\omega_0 = 0.95 Hz$) shows that an initial condition too far from the one at the steady state will never converge. Moreover, one can see how in every experiment in the first few seconds the frequency drops because while the CPG synchronizes the movements of the four legs, the AFO is perturbed by the signal coming from the knees, and this signal is not yet stabilized. After these few seconds the variable ω_l converges. Moreover, the AFO initial conditions give to the dynamical system an initial moving input. As the oscillators parameter μ is equal to zero, without an initial input the system remains inactive. Hence, the initial conditions must bring the AFO to oscillate enough in order to maintain the initial oscillation of the overall system. The two small graphs at the bottom of Figure 4 show the reduction of the difference between the walking frequency and the frequency of the spring-mass oscillating law $\sqrt{\frac{k}{m}}$. Where the error is defined as $e(t) = \sigma(S(t)); S(t) = \sum_i \sum_j a_{ij}; a_{ij} = \frac{\omega_i(t)}{\omega_j(t)} - \sqrt{\frac{m_j}{m_i}}$ where m_i or m_j stands for the mass, ω_i or ω_j stands for the frequency of experiment i or j , and σ stands for standard deviation.

The second plot (Figure 4 right) shows four experiments using the same controller as in the previous experiment (Figure 4 left), with the following initial

conditions: (1) the robot has an initial frequency of 1.1 Hz. (2) In each of the four experiments the robot has a different weight (a) 0.6kg, (b) 0.9kg, (c) 1.2kg, (d) 1.5kg. The plot shows how a higher weight leads to a lower walking frequency. This behavior respects the spring-mass law where the resonant frequency can be calculated as $\sqrt{\frac{k}{m}}$. These results lead to two conclusions: the robot behaves like a spring-mass system, and it adapts its walking frequency to the resonant frequency of the knee. This second experiment shows an interesting property of our system: online adaptation to the physical properties of the robot. In other words, the robot is able to adapt its walking following the weight change during life (for ex. payload change).

Integrating the torque on the angle of the four hips and summing the four values, we have computed the energy consumed by the robot. Then, to obtain the efficiency, we have divided the energy over the distance covered by the robot. We have proved in simulation that this adaptation permits to save energy (in comparison to non-adaptive CPG) when loading a payload of 0.6 Kg of about 15% (data not shown). Our adaptive system does not maximize the walking speed but seems to find a more efficient walking frequency. The frequency found seems to optimize the conversion of the spring potential energy into kinetic energy to propel the robot forward. Our adaptive system can help make the walking locomotion more efficient. This is in line with earlier findings in simulations[6].

4 Experimental Results (Real World)

In this section, we show the experimental results on the Aibo robot. As outlined in Section 3, the Aibo is not the optimal platform to implement the "elastic locomotion framework," since it has activated knee joints, it does not have passive springs in the knee joints. In order to simulate the spring law in the knee joints, we have added controllers for the those joints such that the to a large extent the knees behave like a spring, and thus the robot corresponds to the requirements as stated in the "elastic locomotion framework". As in the case of the simulation, the equations are numerically integrated with a Runge-Kutta algorithm with a fixed time step of 0.008 [s]. The design of Aibo controller also takes care of further implementation problems occurring when simulating a spring law such as encoder resolution and accuracy, mechanical gear backlash, digital system delay, system identification and other problems (cf. [1] for more details). In the following two sections we have repeated the two simulation experiments on real world Aibo.

4.1 Open Loop Results

As in the simulation, we have successfully tested our CPG network, using the default parameters shown in Table 1. Then, we have applied our adaptive component, Eq 3, and plotted the results in Figure 5. When the AFO is perturbed using the rotated sum of the knee signals it quickly converges to a steady state, it also happened when the initial conditions are not close to the steady state. The

rotated solution, using Aibo real world robot signal, seems to have maintained the convergence properties shown in the simulation results (Figure 3).

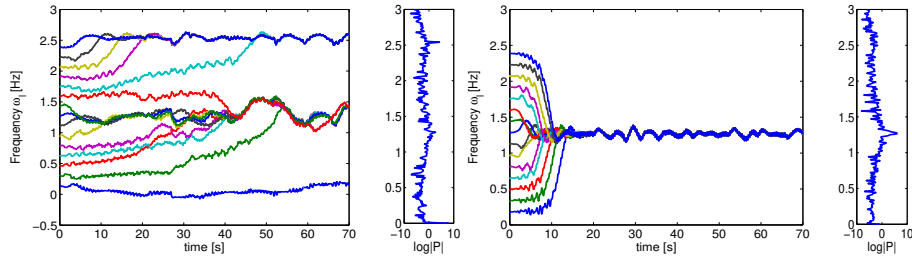


Fig. 5. This experiment is equal to the ones in Figure 3, but on a real AIBO robot. Here the perturbing signal is the un-rotated (left) sum of knees and the rotated one (right). The plot shows how the noise introduced by a real environment produces unstable solutions, in case of un-rotated signal, and very stable solutions in the case of rotated signal ($\eta = 1$). The small graphs are the power spectral density of the un-rotated and rotated signals.

4.2 Closed Loop Results

In the first experiment the initial frequency is $\omega_0 = 0$. The plot, in Figure 6, shows how in this case using a higher learning parameter $\eta = 10$ the robot can learn to walk from scratch. In this case, a (randomly applied) hand-made stress on the knee joints provides the initial input to walking. Aibo quickly reaches a steady state frequency.

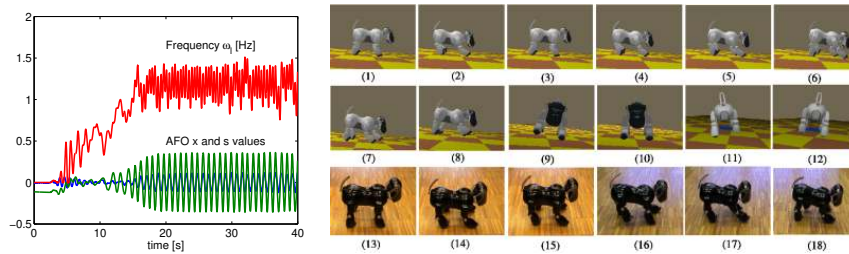


Fig. 6. The figure on the left shows how AIBO learns the walking frequency from scratch ($\omega_0 = 0$) because of the learning parameter $\eta = 10$. On the right there are a series of snapshot during the initial step of Aibo in simulation (from 1 to 12) and in the real world (from 13 to 18). (cf. movies [1])

The second experiment, as described in Section 3.2, involves the parameter adaptation in case of different weights. In this case, we have simulated a real payload change application. The robot starts its life (weight 1.68 [Kg]), (1) adapts its walk, then (2) once a steady state has been reached, (3) one can load

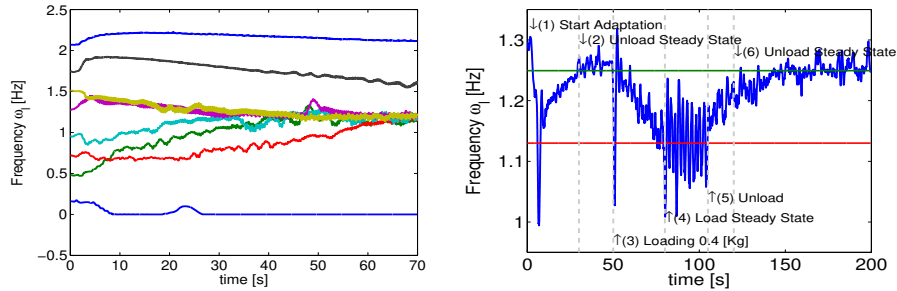


Fig. 7. (left) This figure is the same experiment of Figure 4 (left) but in this case using Aibo real robot. In this case AIBO uses a learning parameter $\eta = 3$. **(right)** This plot shows the variable ω_l and demonstrates the adaptability of our controller in a simple load transport application ($\eta = 10$, plotted is the running average of the frequency: $\omega_{l,p}(t) = \frac{1}{50} \sum_{i=0}^{50} (\omega_l(t-i))$ where $dt = 0.008[s]$, further details are given in the text).

a payload weight (0.4 [Kg]) on the robot saddle, (4) the controller reaches a new steady state adapting the frequency to the new weight, (5) in the end we unload the payload, and (6) the robot returns to a frequency close to the initial one, see Figure 7. This experiment resumes well the interesting autonomous adaptation provided by our controller.

5 Conclusion

In biology, adaptation and memory are major properties of living systems. The main ideas presented in this article are to develop a walking controller where learning is embedded in the dynamics and not an offline optimization process.

The simulated and the real robot have different properties, but thanks to the feedback loop introduced, the controller adapts well its behavior. This demonstrates that the controller is flexible and not designed to work on a specific mechanical system. Moreover, since the robot is autonomous, the controller permits to adapt the walk using proprioceptive signals. Bio-mechanics suggests that the gait of dogs and other quadrupeds can be compared to our elastic locomotion framework. Moreover, the building blocks as well as the entire controller may possibly find implementations in neuro-biological models. We have not dealt with problems such as direction modulation, discontinuous terrain management, and other classical locomotion issues. But we believe that our model is open and flexible enough to be adapted to address these tasks.

We have tested the controller both in computer simulations and on a real robot with successful results in both cases. Furthermore, the frequency adaptation was shown to be useful to provide an efficient gait capable of recycling the energy. Hence, the controller was proved useful for quadruped transportation system applications.

Acknowledgments

G.B. wishes to especially thank his friend, Fabrizio Patuzzo, for his encouragement, support, and review during the entire work. This work was made possible thanks to a Young Professorship Award to Auke Ijspeert from the Swiss National Science Foundation.

References

1. Movies and more technical details of the implementation are available online at <http://birg.epfl.ch/page57636.html>.
2. R. Blickhan. The spring-mass model for running and hopping. *J. Biomechanics*, 22(11–12):1217–1227, 1989.
3. J. Buchli, F. Iida, and A.J. Ijspeert. Finding resonance: Adaptive frequency oscillators for dynamic legged locomotion. Submitted.
4. J. Buchli and A.J. Ijspeert. Distributed central pattern generator model for robotics application based on phase sensitivity analysis. In *Proceedings BioADIT 2004*, volume 3141 of *Lecture Notes in Computer Science*, pages 333–349. Springer Verlag Berlin Heidelberg, 2004.
5. J. Buchli and A.J. Ijspeert. A simple, adaptive locomotion toy-system. In S. Schaal, A.J. Ijspeert, A. Billard, S. Vijayakumar, J. Hallam, and J.A. Meyer, editors, *From Animals to Animats 8. Proceedings of the Eighth International Conference on the Simulation of Adaptive Behavior (SAB'04)*, pages 153–162. MIT Press, 2004.
6. J. Buchli, L. Righetti, and A.J. Ijspeert. A dynamical systems approach to learning: a frequency-adaptive hopper robot. In *Proceedings ECAL 2005*, Lecture Notes in Artificial Intelligence, pages 210–220. Springer Verlag, 2005.
7. A.H. Cohen and D.L. Boothe. Sensorimotor interactions during locomotion: principles derived from biological systems. *Autonomous Robots*, 7(3):239–245, 1999.
8. Y. Fukuoka, H. Kimura, and A.H. Cohen. Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts. *The International Journal of Robotics Research*, 3–4:187–202, 2003.
9. S. Grillner. Control of locomotion in bipeds, tetrapods and fish. In V.B. Brooks, editor, *Handbook of Physiology, The Nervous System, 2, Motor Control*, pages 1179–1236. American Physiology Society, Bethesda, 1981.
10. J. Honerkamp. The heart as a system of coupled nonlinear oscillators. *J Math Biol*, 18(1):69–88, 1983.
11. E. Hopf. Abzweigung einer periodischen Lösung von einer stationären Lösung eines Differentialsystems. *Ber. Math.-Phys., Sächs. Akad. d. Wissenschaften, Leipzig*, pages 1–22, 1942.
12. T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9:62–82, 1990.
13. O. Michel. Webots: Professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):39–42, 2004.
14. Russell Smith & others. *Open Dynamics Engine*. Available online at <http://ode.org>.
15. L. Righetti, J. Buchli, and A.J. Ijspeert. Dynamic hebbian learning in adaptive frequency oscillators. *Physica D*, 216(2):269–281, 2006.
16. S. Strogatz. *Nonlinear Dynamics and Chaos. With applications to Physics, Biology, Chemistry, and Engineering*. Addison Wesley Publishing Company, 1994.