ROBotic Open-architecture Technology for Cognition, Understanding and Behavior

Project No. 004370

RobotCub
Development of a Cognitive Humanoid Cub

Instrument: Integrated Project
Thematic Priority: Cognitive Systems

D 3.8 Crawling Demo

Due date: **18/01/2010**
Submission Date: **21/12/2009**

Start date of project: **01/09/2004** Duration: **60 months**

Organisation name of lead contractor for this deliverable: **EPFL**

Responsible Person: **Sarah Degallier, Ludovic Righetti, Sebastien Gay and Auke Ijspeert**

Revision: **1.0**

Project co-funded by the European Commission within the Sixth Framework Programme (2004-2009)		
Dissemination Level		
PU	Public	**PU**
PP	Restricted to other programme participants (including the Commission Service)	
RE	Restricted to a group specified by the consortium (including the Commission Service)	
CO	Confidential, only for members of the consortium (including the Commission Service)	

Crawling Demo

Introduction

EPFL-B was is in charge of delivering a controller for the iCub for infant-like locomotion, i.e. crawling. Locomotion is an important feature in development because it allows for active exploration of the environment and thus provides autonomy. Furthermore, locomotion (and in particular legged locomotion) is a challenging field of research as it is still an open, arduous issue in robotics. Indeed, it involves the control of many degrees of freedom at the same time, interaction with a possibly unknown environment (and thus modeling issues), discrete contacts with the ground, closed kinematics and dynamics chains and balance issues, among others.

In order to fullfill this task, we have developed a low-level controller for the generation of discrete and rhythmic movements based on the concept of central pattern generators (CPGs), please refer to Deliverable 3.4 for more details. Our main focus was to implement an adaptive, closed-loop controller for crawling. Discrete movements are important in locomotion as they allows for short-term adjustments of the trajectories according to the sensory information (e.g. for visually-guided feet placement). Moreover, as the controller allows for both the generation of discrete and rhythmic movements, it can be used to generate many different behaviors, from reaching to locomotion. It has moreover been designed to be easily integrable in the *iCub* Cognitive Architecture developed by the RobotCub consortium (Vernon et al. (2007)).

Biological Background

To address the complex problem of movement generation, we base ourselves on the concept of central pattern generators - or motor primitives - that is, spinal neural networks that can generate complex, patterned signals for the controls of many muscles and that can be modulated both by simple tonic inputs and/or by sensory feedback.

In terms of control, the concept of central pattern generators is attractive notably because it drastically reduces the dimensionality of the problem: instead of a complex activation of a vast number of muscles across the body, only a couple of synergies of muscles need to be controlled.

The existence of central pattern generators in the human system is well accepted nowadays, even if identification of such a spinal network has not been possible yet. Strong evidence is indeed provided by studies on infants (Thelen (2000); Yang et al. (1998); Lamb and Yang (2000)). Stepping reflexes, just after birth, have been observed in anencephalic infants, providing evidence that circuits responsible for this behavior are located at the spinal and/or at the brain stem level. In addition, studies of disabled patients have shown that in the absence of sensory information, gross movement control is preserved, even if peripheral information is necessary for precise movement organization and control (see Jeannerod (1988) or Gandevia and Burke (1992)). Moreover, it was shown that treadmill exercises for spinal cord injured patients improved the walking pattern (Barbeau and Rossignol (1994); Dietz et al. (1995)).

A review of literature on the generation of discrete and rhythmic movements in vertebrates within this framework of motor primitives, as well as the description of existing mathematical models, has been conducted for Deliverable 3.3.

Modeling

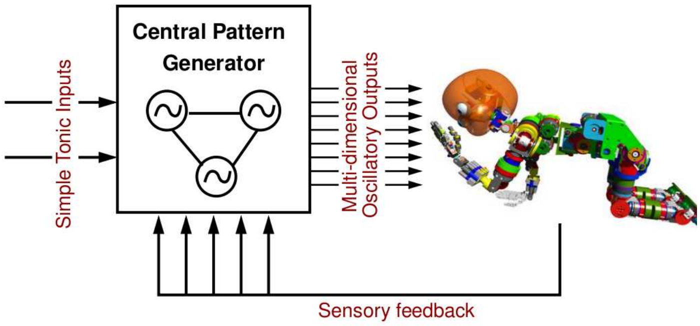


Figure 1: A schematic of the control architecture: the CPGs are modulated by both sensory feedback and high-level inputs. The inputs consist of the target of the discrete movement and the frequency and the amplitude for the rhythmic movement.

To model the concept of CPGs, we based ourselves on dynamical system theory. Indeed, one of the key feature of CPGs is that they can produce complex high-dimensional trajectories that can be modulated by simple high-level and low dimensional inputs. The main idea here is that the complexity should emerge from the dynamics of the network rather than from the command signals. Designing a dynamical system with given, elaborate dynamics is however an irksome task, as no clear methodology exists yet, except for instance adaptive frequency oscillators, a tool that we have developed to learn and reproduce any periodic signal (see Righetti et al. (2009a),Buchli et al. (2008),Righetti and Ijspeert (2008),Righetti and A.J. (2006)).

To simplify the design problem, we model complex dynamics through the combination of simple dynamical systems that serves as primitives of movements; we distinguish two abstract types of motor primitives corresponding to discrete and rhythmic movements. This provide us with a system that can generate both discrete and rhythmic movements, as well as the combination of both, given simple input signals (as illustrated on Fig.2).

Thanks to the integrative nature of dynamical systems, our architecture as several appealing properties in addition the simplicity of the control, such as:

- a low computational cost
- smooth on line modulation of trajectories against parameters changes
- the robustness against perturbations of the attractors
- synchronization between different joints
- the possibility for closed loop control

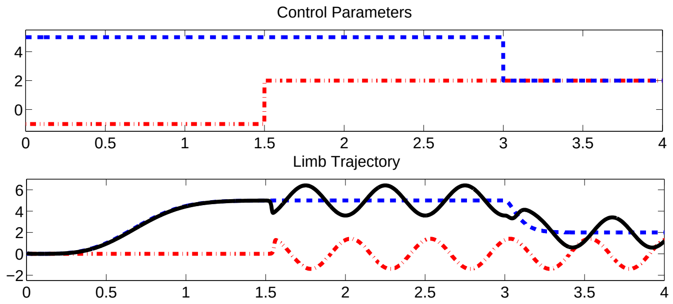


Figure 2: Upper panel. Control commands for discrete and rhythmic movements, that is the target position (in blue) and the amplitudes (in red), the frequency being not shown on the figure. Bottom Panel: The resulting discrete and rhythmic movements (resp. in blue and in red) and the trajectory embedding the two dynamics (black).

While the advantages of the three first items are obvious, the applications to crawling and drumming described below illustrate how coupling can be used to ensure a coordinated behavior across the different degrees of freedom and that by using closed loop control, the robot is able to deal with a time-varying environment.

Note that if the architecture was originally developed for crawling, it has been designed so that it can be used for many other tasks, as for instance reaching or drumming, as it will be briefly mentioned below. Moreover, it has been designed in order to be easy to use, in the sense that discrete and rhythmic movement can be generated simply by specifying the Cartesian position of the object to reach, or the amplitude and the frequency of a rhythmic movement, or the combination of both types of movements. Please refer to Deliverable 3.4 for more details or to Degallier et al. (2008).

A model of the iCub robot, as well as a yarp interface, was developed for the physics-based simulator Webots (Michel (2004)) in order to test our controller. This model is available in the webots directory on svn. We discuss briefly the application of the system to drumming as a test of the architecture and then, more extensively, crawling.

Drumming

As a testbed for the architecture, drumming with both contact and visual feedback has been implemented on the iCub. In this application, a user can freely define the score that the robot is playing on the fly, showing the robustness of the architecture and its smooth on line modulation properties (cf. Deliverable 3.4 and the movies from the different demos). Moreover, thanks to the feedback, the robot adapts its trajectory to the changing environment (displaced drums for instance).

Drumming is a challenging application as it requires coordination between the limbs, precise timing and the robust on line modulation of the parameters, without raising the question of balance, as the robot is fixed to metallic structure in our case. Drumming has been implemented on robots several times before, to study agent-object interaction (Williamson (1999)) or learning from demonstration (Ijspeert et al. (2002)) for instance. Here we focus mainly on the adaptability and on the robustness of the obtained behaviors, indeed trajectories are modulated on the fly by both the high level commands (i.e. through the on line definition of a score) and by visual and ”auditive” feedback information. Previous versions of our implementation of drumming (without the visual feedback) have been published before (Degallier et al. (2006, 2008)).

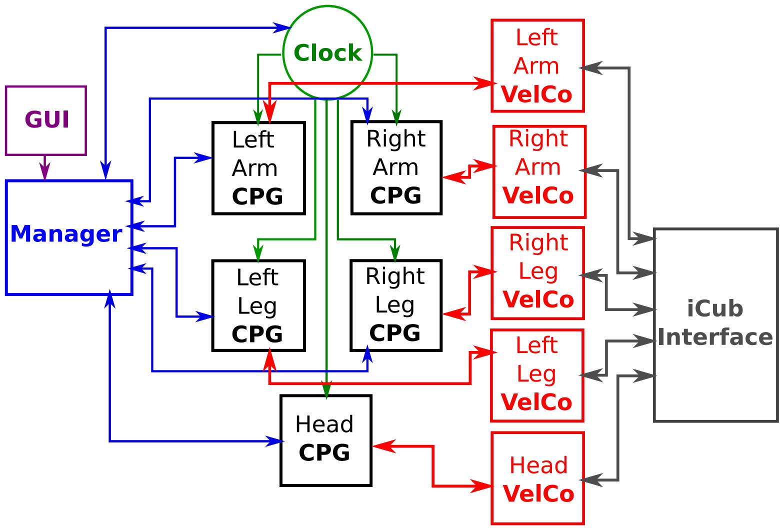


Figure 3: Implementation of the drumming behavior. Five parts are controlled, namely the head, the left arm, the right arm, the left leg and the right leg. Green arrows denote the couplings between the different parts. .

Implementation The set up for drumming is depicted on Fig.3: the robot is fixed to a metallic structure by the hips and plays on an electronic drum set. The four limbs together with the head are controlled. We control actively four joints for each limb and three for the head. The sticks are grasped by the hands which remain fixed afterwards.

Each dof is controlled by the discrete and rhythmic pattern generators (please refer to Deliverable 3.4 or to Degallier et al. (2008)). The dofs of each limb are unilaterally coupled to a clock. Indeed, after a Hopf bifurcation, one can observe a phase resetting of the oscillators; the clock can be seen as a metronome that ensures that the limbs stay in synchronization with the absolute tempo despite those phase resettings.

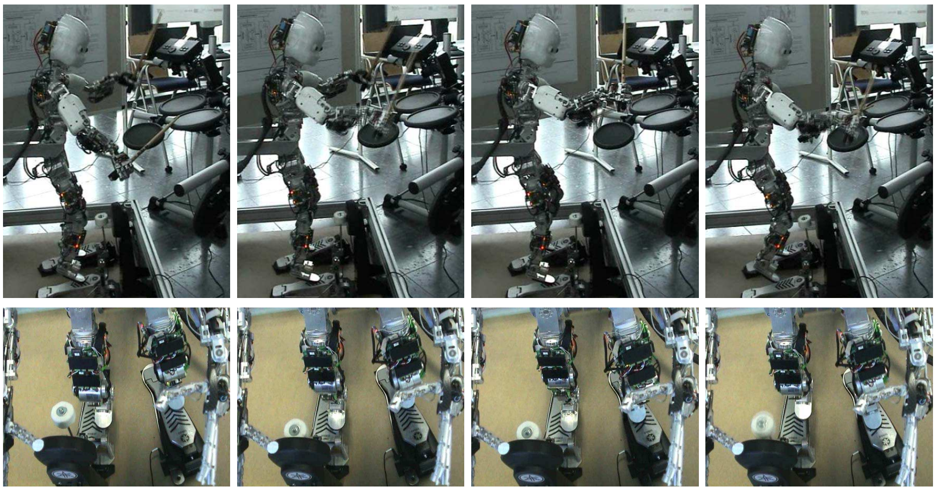
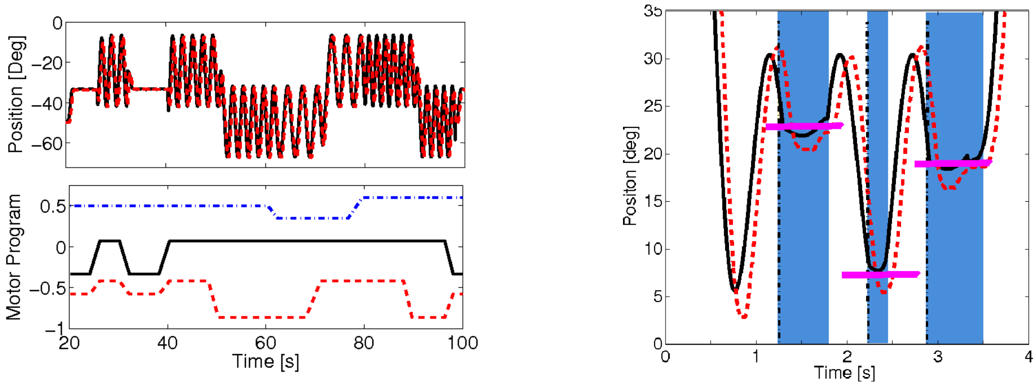


Figure 4: Snapshots of the iCub drumming at the conference CogSys 2008. Top: Side view of the complete robot. Bottom: Downward view of the legs hitting the pedals.

On line score definition The parameters defining a score are the target position g and the amplitude m (on/off) for each dof, the phase shift k_{ij} for each limb (relatively to the leg that plays the bass drum) and the frequency ω (which is the same for each joint). All those parameters can be modified online, at any time, through a graphical user interface (GUI). The manager is then responsible to send those commands at the right timing (i.e. in accordance with the rhythm) to the generator.

Visual Feedback. To get the target position corresponding to the different drums, we use a vision tracker based on ARToolKit and the inverse kinematics module iKin developed by U.Pattacini from IIT. The target angles for the dofs are constantly updated according to the visual informaiton.



(a) Amplitude/Target

(b) Feedback

Figure 5: Drumming trajectories. 5(a) **Up: Generator.** Trajectories generated by the generator for one arm obtained with iCub when drumming. Plain lines are desired trajectories and dotted lines are the actual trajectories. **Bottom: Manager.** Corresponding parameters sent by the manager to the generator: the amplitude (plain line), the frequency(dash-dot line) and the target position in radians (dotted line). 5(b) **Feedback.** Typical trajectories obtained with the feedback enabled; here the robot is tricked, i.e. it is playing without touching any drums, but a user hits the drum at t≈1.3, 2.2 and 2.8 (vertical dash-dot lines) to stop the arm..

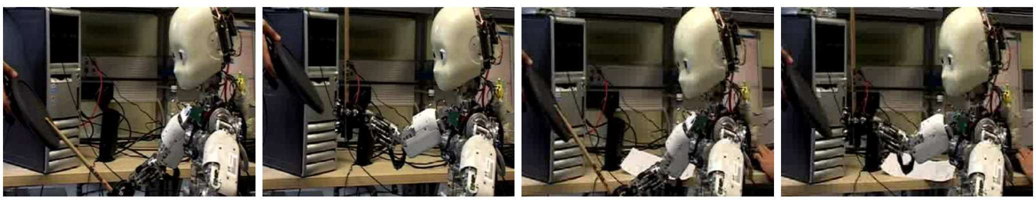


Figure 6: Snapshots of the robot drumming with feedback. The robot adapts its movement to the moving drum.

Auditive Feedback. In order to couple the movements of the robot to the environment, an acoustic feedback was added. Each time a drum is hit, a message is sent to the manager which identifies the corresponding limb and sends a command to the generator to stop the movement in the current position (see Fig.6). Mathematically, an attractor with a high gain is activated to stop the movement in its current position (in Eq. 1) while the dynamics is slowed down (in Eq. 2), i.e. we have the following equations

$$\dot{x} = a(m_i - r_i^2)(x_i - y_i) - \omega s_i + \alpha_{\mathbf{x}}(\hat{\mathbf{x}}_\mathbf{i} - \mathbf{x}_\mathbf{i}); \quad (1)$$

$$\dot{s} = \frac{a(m_i - r_i^2))s_i + \omega(x_i - y_i)}{\mathbf{1} + \alpha_{\mathbf{y}}(\hat{\mathbf{x}}_\mathbf{i} - \mathbf{x}_\mathbf{i})^\mathbf{2}} \quad (2)$$

where $\hat{x}_i$ is the current desired position of joint i when the feedback is received.

Conclusion. Thanks to this application, we were able to test the suitability of the architecture for

- the coordinated control of the whole body (19 degrees of freedom were controlled at the same time);
- the on line modulation of the parameters (instead of playing a predefined score, a user could change on the fly the behavior of the robot);
- the integration of contact information; and
- the integration of visual feedback.

Crawling

As it was mentioned before, the principal contribution of crawling during development is that it allows for the active exploration of the environment. Thus, we defined a target final target scenario for crawling that can be described as follows:

> *The robot is seated in a room containing target objects and obstacles. Being attracted by a target object, the robot goes on all fours, start crawling towards the object while avoiding the obstacles. When close enough from the object, the robot stops, reach for it and then sits.*

This complex behavior involves three different simpler behaviors, namely:

- crawling;
- going on all fours / sitting; and
- simple reaching.

and high-level planning for steering the robot between the obstacles.

Implementation. For this application, both arms and legs are controlled as well as the head and the torso. For each arm and leg, we actively control 4 dofs, that are the shoulders pitch, roll and yaw and the elbows for the arms and the hips pitch, roll and yaw and the knee for the legs; the three degrees of freedom of the head and torso are also controlled. We thus actively control 22 dofs. The remaining dofs are set in particular position at the beginning of the task and remain fixed at that position afterwards.

During crawling, the trajectories of the hip and shoulder pitch joints are rhythmical (with fixed offset), while these of the other controlled joints are purely discrete. When the robot has to stop or reach, the oscillations are switched off and all the produced trajectories are discrete.

Crawling in infants. Our goal was to develop a controller that reproduces the key kinematic features of crawling in infants (rather than the exact trajectories). Indeed the mass distribution and the compliance of the robot being different to ones of an infant, we have chosen not to focus on kinematic details as they may be consequences of these features.

In order to do so, a preliminary, extensive study of the kinematics of crawling infants was performed in collaboration with the University of Uppsala (Righetti et al. (2009b)). The key results were that

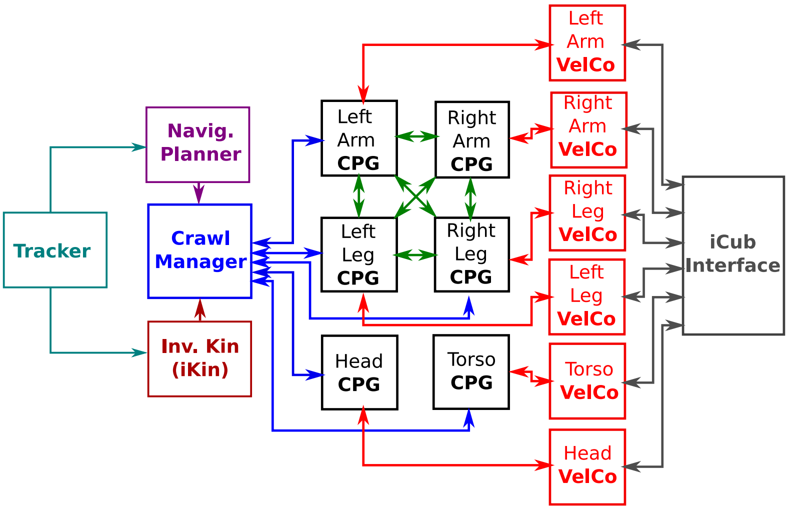
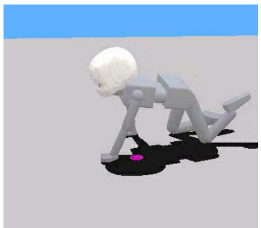
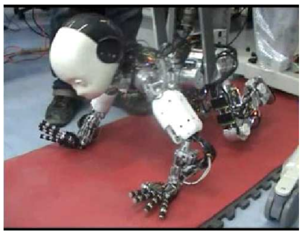


Figure 7: Implementation of the crawling behavior. Six parts are controlled, namely the head, the torso, the left arm, the right arm, the left leg and the right leg. Green arrows denote the couplings between the different parts.

(a) Real baby



(b) Simulated iCub



(c) iCub robot

Figure 8: Crawling was first studied is in real infants, then a model was established and tested in simulation to be later integrated on the real robot.

- Crawling infants use a walking trot gait (duty factor $\geq$ 50%) and the hands swing first, which is theoretically the most stable gait ;
- Swing duration is almost constant for every speed and locomotion speed is linearly correlated to stance duration, as observed in other mammals.

It is noteworthy that the crawling gait share common properties with gaits observed in other quadruped mammals albeit the great difference in limb geometry, which emphasizes similarities in neural control during locomotion among mammals.

Mathematical Model. To obtain a trot gait, the hip pitch and shoulder pitch joints of each arms and legs are coupled together using the following formulation:

For generating goal directed movement towards a target angle g, we use the following set of equations:

$$\dot{h}_i = d(p - h_i) \tag{3}$$
$$\dot{y}_i = h_i^4 v_i \tag{4}$$
$$\dot{v}_i = p^4 \frac{-b^2}{4} (y_i - g_i) - b\, v_i \tag{5}$$

here and in the following i denotes a particular dof. The system is critically damped so that the output y_i of Eqs 4 and 5 converges asymptotically and monotically to a goal g_i with a speed of convergence controlled by b, whereas the speed v_i converges to zero. p and d are chosen so to ensure a bell-shaped velocity profile; h_i converges to p and is reset to zero at the end of each movement.

Cyclic movements are produced through the following set of equations:

$$\dot{x}_i = a\left(m_i - r_i^2\right) x_i - \omega_i z_i \tag{6}$$
$$\dot{z}_i = a\left(m_i - r_i^2\right) z_i + \omega_i x_i \tag{7}$$
$$\tag{8}$$

where $r_i = \sqrt{x_i^2 + z_i^2}$. When $m_i > 0$, Eqs. 6 and 7 describe an Hopf oscillator whose solution x_i is a sine of amplitude $\sqrt{m_i}$ and frequency ω_i. A Hopf bifurcation occurs when $m_i < 0$ leading to a system with a globally attractive fixed point at (0,0).

Both discrete and rhythmic dynamics are combined by embedding the discrete output y_i as an offset of the rhythmic output x_i, that is

$$\dot{x}_i = a\left(m_i - r_i^2\right)(\mathbf{x_i} - \mathbf{y_i}) - \omega_i z_i \tag{9}$$
$$\dot{z}_i = a\left(m_i - r_i^2\right) z_i + \omega_i (\mathbf{x_i} - \mathbf{y_i}) \tag{10}$$
$$\tag{11}$$

where $r_i = \sqrt{(\mathbf{x_i} - \mathbf{y_i})^2 + z_i^2}$. We call such system (that is Eqs.3-5 and Eqs.9-10) a motor primitive.

The trot gait is obtained by coupling the motor primitives together in the following way

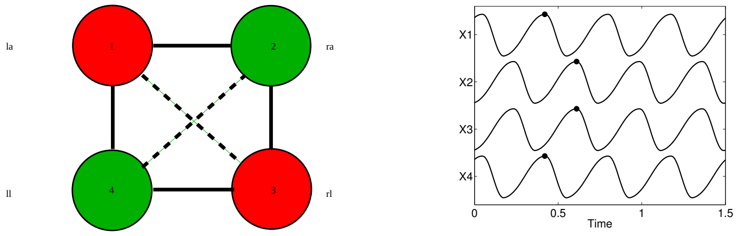
$$\dot{x}_i = a\left(m_i - r_i^2\right)(x_i - y_i) - \omega_i z_i \tag{12}$$
$$\dot{z}_i = a\left(m_i - r_i^2\right) z_i + \omega_i (x_i - y_i) + \sum k_{ij}^y z_j \tag{13}$$

with the k_{ij}^y's as defined in Tab.1.

	left arm	right arm	left leg	right leg
left arm	0	-1	-1	1
right arm	-1	0	1	-1
left leg	-1	1	0	-1
right leg	1	-1	-1	0

Table 1: Parameters k_{ij}^y's needed in Eq.13 to obtain a trot gait.

We have seen that the duration of the stance is usually longer than the duration of the stance in infant crawling. Moreover, the speed of locomotion is controlled through the duration of the stance



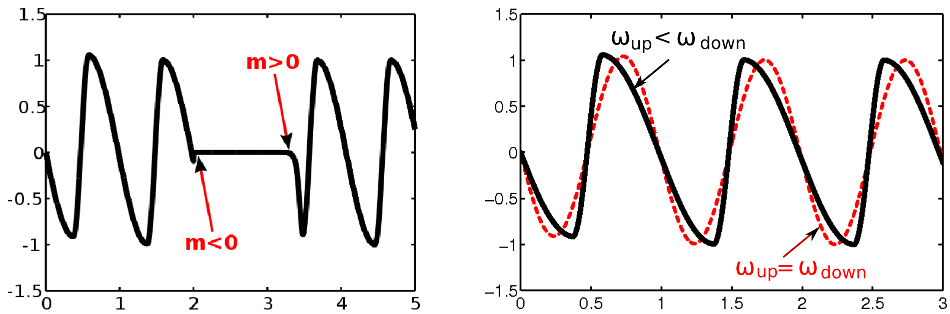
(a) Coupling between the limbs (b) Trajectoires for the trot

Figure 9: In the trot gait, diagonal pairs of arm and legs move forward at the same time.

only, the swing duration being relatively constant. To reproduce these features, we thus express the frequency ω_i as a function of two variables ω_{swing} and ω_{stance}:

$$\omega_i = \frac{\omega_{swing}}{e^{-fz_i} + 1} + \frac{\omega_{stance}}{e^{fz_i} + 1} \tag{14}$$

where f is a parameter controlling the duration of the switch between the two phases.



(a) Bifurcation (b) Swing/stance

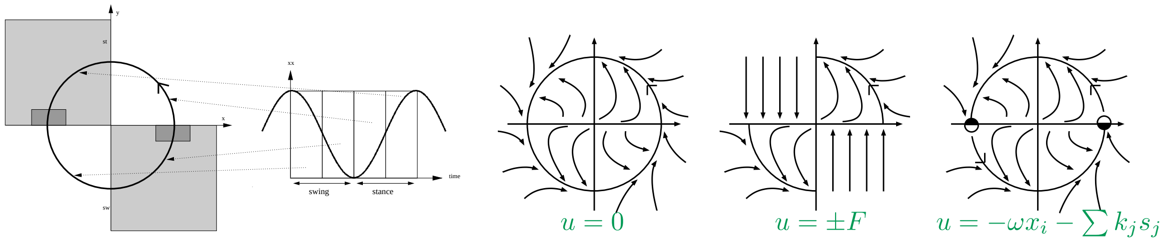
Figure 10: Rhythmic system. (a) The oscillations can be turned on and off through the parameter m controlling the amplitude. (b) Moreover, thanks to Eq.14, the duration of the stance and the swing can be specified independently.

During crawling, the joints other that the hips/shoulder pitch are controlled in the following way. The shoulder roll, the elbow and the hip roll are fixed during the stance and move proportionally to the speed of the shoulder pitch joint during swing, i.e.

$$g_i = w_i z_j \tag{15}$$

where j= shoulder pitch if i = shoulder roll or elbow and j= hip pitch if i = hip roll, where the w_i are chosen so to ensure that the knees and the hands are lifted enough during the swing to avoid collision with the ground. In other circumstances, the joints remain in the initial position ”on all fours” that was described above.

There are two control parameters: the duration of the stance (that controls the duty factor) and the amplitude of the oscillations of the hip pitch (that controls the step length), both of which influence the speed of locomotion. Note that the specification of the hip pitch amplitude impose those of the shoulder pitch for coherence reason (indeed the distance covered by the knees and the hands should be the same for the robot to go straight); the shoulder pitch amplitude is deduced using the forward kinematics module iKin developed by U. Pattacini-



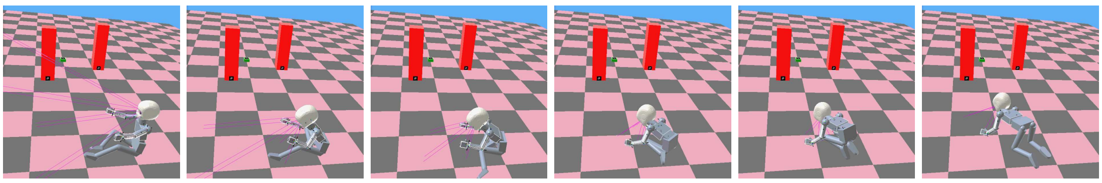
(a) Feedback activation zone (b) Oscillator phase spaces

Figure 11: Phase space of an oscillator with its activation zone for the feedback (light gray for switch and dark gray for stop controls) and the correspondence with the x variable of the oscillator is shown on the left figure. Right figure shows the schematic phase plot of the oscillator for the different types of feedback.

In addition, a phase dependent sensory feedback is included in the rhythmic PG to make the crawling locomotion more robust and adaptive to the environment, as we did previously in Righetti and Ijspeert (2008). Information from the touch sensors located on the hands and knees of the robot is used to modulate the onset of the swing and stance phases, as mammals do Frigon and Rossignol (2006). The transition from stance to swing phases is delayed as long as the other limbs cannot support the body weight and is triggered sooner when the limb leaves unexpectedly the ground. Analogous policies are used for the swing to stance transition. More precisely, the term u_i of Eq 7 is defined as

$$u_i = \begin{cases} -\mathrm{sign}(y_i)F & \text{fast transitions} \\ -\omega x_i - \sum k_{ij} y_j & \text{stop transition} \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

where F ($= 200$ in our case) controls the speed of the transition. Fig. 11 shows the activation of the feedback depending on the phase of the limb and the resulting modification of the phase space of the oscillator.



(a) Transition from sitting to crawling

Figure 12: Snapshots of the transition from sitting to crawling.

Transitions between crawling and sitting. These transitions were implemented as purely discrete tasks, more precisely as sequences of target positions for the whole body. These sequences are inspired from the observation of theses transition in infants, as illustrated on Fig.12. It is unfortunately not possible to implement these behaviors on the real robot due to restrictive joint limits.

Reaching . When the robot is close enough to the target object, it stops and reaches for it. The Cartesian position of the object is transformed into joint angles through the inverse kinematics module IKin developed by U. Pattacini from IIT.

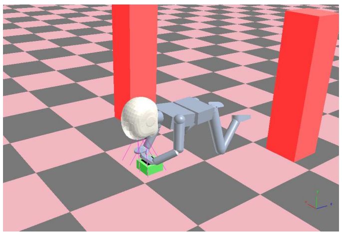
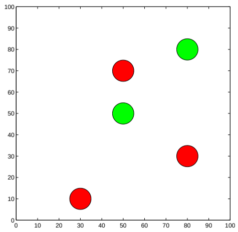
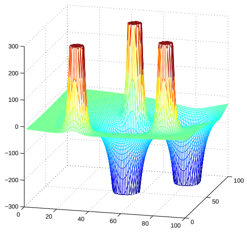


Figure 13: Snapshot of the robot reaching a mark on the ground.

Steering A high level planner based on force fields has been developed to illustrate how the crawling controller can be used in a simple navigation task. A representation map of the different positions of the obstacles and targets, acquired through a vision multi-object tracking module based on ARToolKit, is turned into a surface where obstacles and targets are represented by respectively positive and negative peaks. The trajectory is then given by the gradient of the surface.

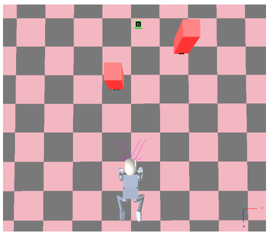


(a) Representative Map

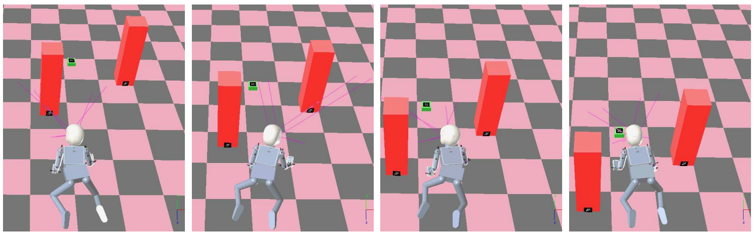


(b) Potential Fields

Figure 14: A map representing the obstacle and targets is created (a) and transformed in to a surface through force fields (b).



(a) World from top

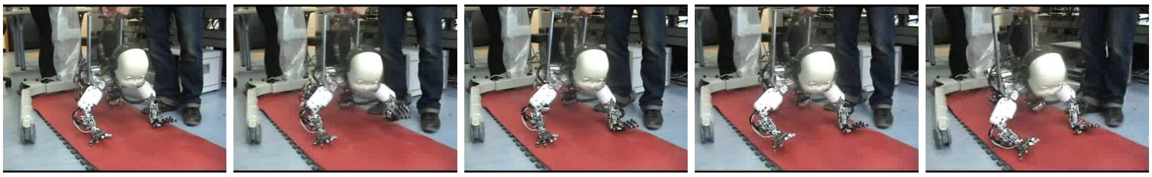


(b) Steering

Figure 15: Snapshots of the robot avoiding two obstacles.

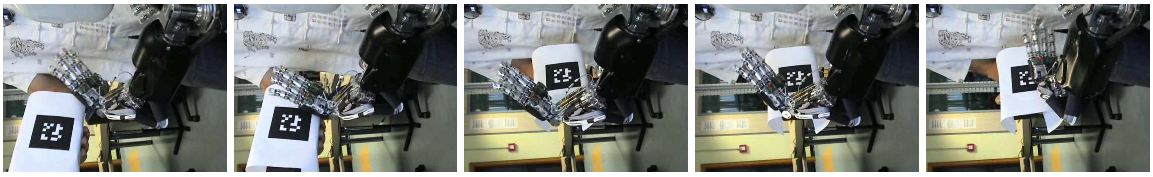
The robot torso roll joint as well as the relative amplitudes of the right and left limbs are used to control the direction of locomotion of the robot. This implementations show how easy it is to potentially combine our low-level architecture with high-level representation of movements, and thus further integration with the iCub cognitive architecture.

Implementation on the iCub Crawling in open loop was successfully implemented on the robot, although some low-level control issues appeared afterwards, issues that are still under debugging. Yet, all these behaviors have been tested on the robot in the air (i.e. without contact with the ground).



(a) Crawling in open loop

(b) Steering



(c) Reaching

Figure 16: Current implementation on the real robot. (a) Crawling in open loop (b) Turning on the left (c) Reaching for a mark

Conclusion

We propose here a promising approach to locomotion (and to movement in general) based on the biologically inspired concept of CPGs, that are spinal neural networks that can generate complex outputs given simple, non-patterned inputs. Thanks to these low-level motor primitives, the architecture that was developed allows for a extremely simple high-level control of the tasks, in the sense that the only parameters that need to be provided to the CPGs are the goals of the tasks (rather than full trajectories for instance). Indeed, to control crawling, only the speed of locomotion (and possibly the amplitudes of the arms) or the angle of rotation for turning need to be provided to the CPG. The CPG will then produce and modulate the corresponding gait in real time. Simply specifying the Cartesian position of an interesting object results in a reaching movement, and in the displacement of the robot towards this object when it is too far. Such implementation makes it easy to use for generating discrete or rhythmic movements for anyone focusing more on high-level planning.

In addition, our approach to locomotion emphasizes adaptivity, in the sense that instead of using a purely predictive model, we extensively use feedback information to modulate the behavior of the robot according to its time-evolving environment (as for instance tilted terrain during crawling, moving targets during reaching or moving drums while drumming).

Bibliography

H Barbeau and S Rossignol. Enhancement of locomotor recovery following spinal cord injury. *Curr Opin Neurol*, 7(6):517–24, December 1994. ISSN 1350-7540.

J. Buchli, L. Righetti, and A.J. Ijspeert. Frequency analysis with coupled nonlinear oscillators. *Physica D: Nonlinear Phenomena*, 237:1705–1718, August 2008.

S. Degallier, C. P. Santos, L. Righetti, and A. Ijspeert. Movement generation using dynamical systems: a humanoid robot performing a drumming task. In *IEEE-RAS Inter. Conf. on Humanoid Robots*, pages 512–517, 2006.

S. Degallier, L. Righetti, L. Natale, F. Nori, G. Metta, and A.J. Ijspeert. A modular bio-inspired architecture for movement generation for the infant-like robot icub. In *Proceedings of the second IEEE RAS / EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob*, 2008.

V Dietz, G Colombo, L Jensen, and L Baumgartner. Locomotor capacity of spinal cord in paraplegic patients. *Ann Neurol*, 37(5):574–82, May 1995. ISSN 0364-5134.

S. Frigon and S. Rossignol. Experiments and models of sensorimotor interactions during locomotion. *Biological Cybernetics*, 95(6):607–627, 2006.

S.C. Gandevia and D. Burke. Does the nervous system depend on kinesthesic information to control natural limb movements? *Behav. Brain Sci.*, 15:614–632, 1992.

A.J. Ijspeert, J. Nakanishi, and S. Schaal. Learning rhythmic movements by demonstration using nonlinear oscillators. In *Proceedings of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS2002)*, pages 958–963, 2002.

M. Jeannerod. *The neural and the behavioural organization of goal directed movements*. Oxford Science, Oxford, 1988.

T. Lamb and J.F. Yang. Could different directions of infant stepping be controlled by the same locomotor pattern generator. *J. Neurophysiol*, 83:2814–2824, 2000.

O. Michel. Webots tm: Professional mobile robot simulation. *International Journal of Advanced Robotic System*, 1:39–42, 2004.

L. Righetti and Ijspeert A.J. Programmable central pattern generators: an application to biped locomotion control. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006.

L. Righetti and A.J. Ijspeert. Pattern generators with sensory feedback for the control of quadruped locomotion. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA 2008)*, pages 819–824, May 2008.

L. Righetti, J. Buchli, and Ijspeert A.J. Adaptive frequency oscillators and applications. In *The Open Cybernetics and Systemics Journal*, 2009a.

L. Righetti, A. Nylén, K. Rosander, and A.J. Ijspeert. Are crawling human infants different from other quadruped mammals? In *Submitted*, 2009b.

E. Thelen. Motor development as foundation and future of developmental psychology. *Int. Journ. of Behavioral Development*, 24(2):385–397, 2000.

D. Vernon, G. Metta, and G. Sandini. The icub cognitive architecture: Interactive development in a humanoid robot. *IEEE International Conference on Development and Learning*, 2007.

M. Williamson. *Robot Arm Control Exploiting Natural Dynamics*. PhD thesis, MIT Department of Electrical Engineering and Computer Science, 1999.

J. F. Yang, M.J. Stephens, and R. Vishram. Infant stepping: a method to study the sensory control of human walking. *J. Physiol (Lond.)*, 507:927–937, 1998.