Pattern generators with sensory feedback for the control of quadruped locomotion

Ludovic Righetti and Auke Jan Ijspeert

Abstract— **Central Pattern Generators (CPGs) are becoming a popular model for the control of locomotion of legged robots. Biological CPGs are neural networks responsible for the generation of rhythmic movements, especially locomotion. In robotics, a systematic way of designing such CPGs as artificial neural networks or systems of coupled oscillators with sensory feedback inclusion is still missing.**

In this contribution, we present a way of designing CPGs with coupled oscillators in which we can independently control the ascending and descending phases of the oscillations (i.e. the swing and stance phases of the limbs). Using insights from dynamical system theory, we construct generic networks of oscillators able to generate several gaits under simple parameter changes. Then we introduce a systematic way of adding sensory feedback from touch sensors in the CPG such that the controller is strongly coupled with the mechanical system it controls.

Finally we control three different simulated robots (iCub, Aibo and Ghostdog) using the same controller to show the effectiveness of the approach. Our simulations prove the importance of independent control of swing and stance duration. The strong mutual coupling between the CPG and the robot allows for more robust locomotion, even under non precise parameters and non-flat environment.

I. INTRODUCTION

The control of locomotion of legged robots is a great challenge that has not yet been completely solved. Recently, taking inspiration from the way nature solves the problem, researchers have proposed the concept of Central Pattern Generators (CPGs) to generate control policies for the locomotion of robots [1]–[5]. In biology, a CPG is a distributed neural network, located in the spinal cord of vertebrates, able to generate complex high dimensional signals for the control of coordinated periodic movements [6]. These CPGs are activated by simple tonic signals from higher part of the brain and they are strongly coupled with the body they control and the environment, via sensory feedback.

In robotics, these CPGs are often modeled as coupled dynamical systems, mostly oscillators. The advantage of using such controllers is their stability properties (limit cycle behavior), where transient perturbations are rapidly forgotten. Another advantage is that oscillators have intrinsic properties of synchronization that allow strong coupling with the robot and its environment. Moreover, using such systems for generating the control policies reduces the dimensionality of the control problem, since only simple parameters as frequency, amplitude and coupling between the oscillators have to be chosen to generate high dimensional coordinated policies. Finally, since these methods are model free, they are well adapted to the locomotion in unknown environments. This approach has been applied for the control of both simulated [4] and real biped robots [1], [3], snake and salamander robots where one CPG was able to control both walking and swimming [5] and quadruped robots [2] to cite a few successful examples. However, a major drawback in the CPG approach is the lack of design methodologies. Despite some attempts to provide generic tools for building CPGs [7], [8], most of the time they are tailor-made for specific applications and very few design principles are available, especially for the integration of feedback in such systems.

In a recent contribution [8], we presented a simple way to independently control the duration of swing and stance phases during locomotion (i.e. controlling the duration of the ascending and descending phases of the oscillators), which allowed also to control the shape of the control signal. We used the theory of symmetric coupled cells networks [9] to design a generic coupling architecture for our CPG and showed how it could be applied to the control of a crawling humanoid robot.

In this contribution, we continue our previous attempt to provide design methodologies. We present a generic network of coupled oscillators able to generate different gaits (walk, trot, bound and pace) and we integrate sensory feedback in the CPG in a systematic manner. Several other contributions have proposed ways to integrate sensory feedback in CPGs, most of them are based on the idea of using the sensory input either to reset the phase of the oscillators when the feet touch the ground [1], or by using the entrainment properties of the oscillators [2]–[4]. Other feedback pathways have been designed in [2], however these are specific to a particular robot and oscillator model and it is not clear how to use the same controller for other robots. The novelty of this contribution compared to other works is threefold: first it proposes generic networks of oscillators to generate gaits independently of the chosen oscillators, second the oscillator model is specifically designed for locomotion and allows for independent control of swing and stance durations, third we include sensory feedback by explicitly shaping the phase space of the oscillators such that we can control precisely the behavior of our system. A great advantage of such an approach is that the resulting controller is simpler compared to [2] and can easily be used to control very different robots. We show the genericity of our approach by testing this CPG on three different simulated quadruped robots (iCub, Aibo

This work was supported by the European Commission's Cognition Unit, project no. IST-2004-004370: RobotCub (L.R.) and by a grant from the Swiss National Science Foundation (L.R. and A.I.).

School of Communication and Computer Science, Ecole Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland ludovic.righetti@a3.epfl.ch, auke.ijspeert@epfl.ch

and Ghostdog), using several gaits, on different terrains.

II. Assumptions and necessary features for a locomotion controller

We first define a set of desirable features for our locomotion controller and we make some assumptions on the form of the controller, in order to constrain our design space.

A. Features

A controller for quadruped locomotion should be able to generate several different gaits into a single controller (via appropriate simple parameters change). Indeed it is quite obvious that different gaits are suitable for different speeds. At slow speeds, one would prefer statically stable gaits such as a walk and at high speeds dynamically stable gaits such as bound or gallop. From the control point of view, the system should generate smooth control policies with the possibility to modulate smoothly these policies in amplitude and frequency by simple parameter change. We would also like the system to be stable against perturbation in order to allow integration of sensory feedback. Suitable feedback pathways should make the locomotion more robust in unknown environment. Finally the architecture of the CPG should be simple and generic enough to be applied to different kinds of robots and to allow integration of higher controls (e.g. precise feet placement) or more stereotyped reflexes (e.g. moving the foot higher when hitting an object in the middle of swing).

In summary, the required features are

- Generation of several gaits
- Stability of the controller to allow feedback integration
- Smooth policies modulated by simple parameter change
- Robust locomotion by feedback integration
- Genericity of the architecture (applied to several robots)
- Framework that allows more complex behavior

B. Assumptions

It is well known in biology that the speed of locomotion in quadruped animals is controlled by the duration of the stance phase (there exists a linear relation between inverse of stance duration and speed). On the other hand the duration of the swing is almost constant for any gait and has no relation with the speed of locomotion, certainly for stability issues. We assume that it might be the same for robots, so we need to independently control the durations of swing and stance in our CPG. We also know that force sensing under the feet has a very important role in the behavior of biological CPGs [6]. Indeed it strongly modulates the onset of the swing and stance phases. In our system one limb should stay in swing phase as long as the foot does not touch the ground, if the foot touches the ground sooner than expected, then the controller should switch to stance phase. We expect the opposite behavior to happen for the stance phase. From a control point of view, we assume that the CPG is a network of coupled oscillators that encode in their phase space the locomotion policy of each limb. One advantage of using oscillators is that the coordination of the limbs comes as an intrinsic property of the system and moreover they are stable against perturbations (limit cycle behavior). Using such an approach is interesting for sensory feedback integration, because we can view the CPG as a system that is controlled by the sensory information, i.e. this information, depending on the state of the CPG, will change the form of its phase space. We use the term "controlled by sensory information" because of the strong coupling that can completely change the internal dynamics of the CPG.

As a summary we make the following assumptions:

- Independent control of swing and stance duration
- CPGs encode the control policies in phase space
- Strong coupling with the mechanical system (the CPG controls and is "controlled" by the mechanics)
- Sensory feedback modulates the onset of swing and stance phases

III. Locomotion controller design

We present the central pattern generator (CPG) that generates the control policies for locomotion, including the properties and assumptions described in the previous section.

A. Feedforward trajectory generator

1) Oscillator model: We need an oscillator in which we can independently control the swing and stance durations (i.e. the ascending and descending phases of the oscillation). To do so we use a modified Hopf oscillator that has a phase-dependent frequency. It is based on the oscillator we designed in [8], but normalized such that the limit cycle has an invariant shape for different frequencies. Its equation is

$$\dot{x} = \alpha(\mu - r^2)x - \omega y \quad (1)$$

$$\dot{y} = \beta(\mu - r^2)y + \omega x \quad (2)$$

$$\omega = \frac{\omega_{stance}}{\mathrm{e}^{-by}+1} + \frac{\omega_{swing}}{\mathrm{e}^{by}+1} \quad (3)$$

where $r = \sqrt{x^2+y^2}$, ω is the frequency of oscillations in $\mathrm{rad}\cdot\mathrm{s}^{-1}$, $\sqrt{\mu}$ is the amplitude of oscillations. ω_{swing} and ω_{stance} are the frequencies of the swing and stance phases respectively. α and β are positive constants that control the speed of convergence to the limit cycle. Using different speeds of convergence for the x and y variables allows to control more precisely the generation of the control policy. For example in the following we use $\mu = 1$, $\alpha = 5$ and $\beta = 50$, so the convergence is faster on the y axis than on the x axis. We thus have strong convergence properties, while having a limited derivative for the control policy generated by x. We motivate this choice of oscillator because the Hopf oscillator has a harmonic limit cycle that is structurally stable and the shape of the limit cycle is independent of the choice of the frequency parameters. Moreover we can explicitly and independently control the frequencies of the ascending and descending phases of the oscillation as well as their amplitude with ω_{swing}, ω_{stance} and μ.

The control policy is the x variable and represents joint angle of the most proximal joint in the sagittal plane of a limb in our experiments. However it can be noted that x could easily represent any other quantity (position of the hand,

torque, etc...). The policy of the other joints is a function of x and is discussed in Sec. IV.

2) Network structure: We now couple the oscillators in order to generate the desired gaits. Here we extend our previous work [8] to have a network able to generate walk, trot, pace and bound. To define the generic coupling architecture, we use results from the theory of symmetric coupled cells [10]. This theory gives conditions on the existence of symmetric periodic solutions in networks of coupled dynamical systems. The interest of such a theory is that the design of the network structure can be made independent of the internal dynamics of the cells. The design of such a network relies only on algebraic arguments, which makes the design of the network easier and scalable to more complex networks (e.g. for hexapod locomotion). Moreover it is then possible to calculate the other possible periodic solutions of the network and make sure that only the desired ones are stable.

The symmetries present in a network of coupled cells induce the existence of periodic solutions possessing the same symmetries. A symmetry in the network is defined by a permutation of the cells that preserves the coupling architecture (we assume here that all the cells have the same internal dynamics). Given a network, we can define a coupling matrix A whose entries describe the coupling between the cells. Then we say that a linear transformation γ is a symmetry of the network if $\gamma A = A$ (i.e. the network structure is invariant under the transformation)[1].

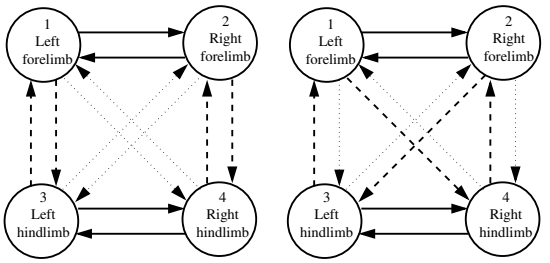
The spatial symmetries of a periodic solution $x(t)$ of a system of differential equations are the symmetries γ such that $x(t) = \gamma x(t)$, for all t. The spatio-temporal symmetries σ of a periodic solution $x(t)$ are the symmetries that preserve the orbit of the solution, i.e. the symmetries that induce a phase shift, ψ, in the solution such that $\sigma x(t) = x(t + \psi)$.

The main result we use is the H/K Theorem from [9],

Theorem 1: H/K Theorem Let Γ be the symmetry group of a coupled cell network in which all cells are coupled and the internal dynamics of each cell is at least two-dimensional. Let $K \subset H \subset \Gamma$ be a pair of subgroups. Then there exist periodic solutions to some coupled cell systems with spatio-temporal symmetries H and spatial symmetries K if and only if H/K is cyclic and K is an isotropy subgroup. Moreover, the system can be chosen so that the periodic solution is asymptotically stable.

Knowing the symmetries of a desired gait, we can now construct a network that can generically support this gait by calculating the constraints on the coupling matrix that the symmetry imposes. The trot gait is a gait in which the diagonal legs move in synchrony and half a period out of phase with the contralateral legs. Thus, using permutation notation, if we number the limbs as in Fig 1, we have one spatial symmetry $((14)(23), 0)$ (the 0 means no phase shift after the permutation) and 2 additional spatio-temporal symmetries $((12)(34), \frac{1}{2})$ and $((13)(24), \frac{1}{2})$. The symmetry group generated by these symmetries is isomorphic to $\mathbb{Z}_2 \times \mathbb{Z}_2$. It can easily be calculated that the pace and bound gaits have the same group of symmetries and that generically in a system of 4 coupled cells having the same symmetry, all the 3 gaits will co-exist (since the subgroups of the network are $\{I, ((12)(34))\}$, $\{I, ((13)(24))\}$ and $\{I, ((14)(23))\}$). It has indeed already been shown that these solutions are conjugate in a 4 cell network [10]. However, since the stability of these conjugate solutions can be determined by an appropriate choice of coupling parameters, we can make sure that only the desired pattern is stable. From a control point of view, this is satisfactory, since unstable solutions cannot be reached. For the walking gait, the generator of the group is $((1423), \frac{1}{4})$ and the symmetry group is isomorphic to $\mathbb{Z}_4$. Figure 1 shows the minimal structure of networks for supporting walk, pace, trot and bound. The hypotheses of the H/K Theorem (H/K cyclic and K isotropy subgroup) can easily be checked to be true. Then it is also possible to calculate the other possible periodic solutions by calculating the subgroups of the symmetry group of the network.

[1]Rigorous mathematical definitions and technical hypotheses such as Γ-equivariance of the ODEs can be found in [9]



(a) Trot Pace Bound Net. (b) Walk Net.

Fig. 1. Generic coupling structure for the 4-cell networks. Same coupling between cells is shown by arrows of the same type, this also shows how the network is constrained by the required gaits.

We now couple the oscillators according to generic coupling matrices such that the desired gaits are stable. Their stability was verified by numerical simulations. The equations for the CPG are now

$$\dot{x_i} = \alpha(\mu - r_i^2)x_i - \omega_i y_i \quad (4)$$

$$\dot{y_i} = \beta(\mu - r_i^2)y_i + \omega_i x_i + \sum k_{ij} y_j \quad (5)$$

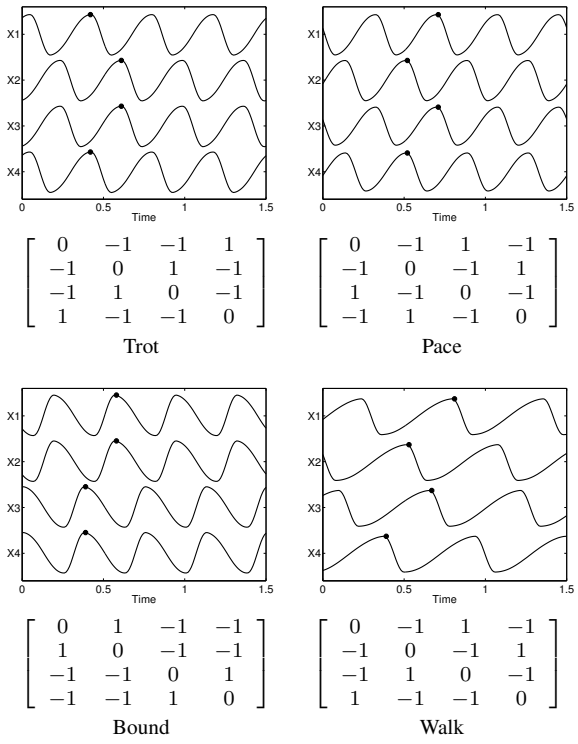
the k_{ij} being defined by the coupling matrices. The other parameters are defined as before. Figure 2 shows the coupling matrices and the resulting gaits. Note that the type of gait is determined by the coupling matrix and is not affected by the swing-stance durations.

B. Feedback modulation

In this section we show how to integrate sensory feedback from the touch sensors on the extremities of the limbs in order to modulate the onset of swing and stance phases. The idea is to explicitly change the phase space of the oscillator according to their current state and the sensor values. First we rewrite the oscillator equations, where we add a control input u as

$$\dot{x_i} = \alpha(\mu - r_i^2)x_i - \omega_i y_i \quad (6)$$

$$\dot{y_i} = \beta(\mu - r_i^2)y_i + \omega_i x_i + \sum k_{ij} y_j + u_i \quad (7)$$



$$\begin{bmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & 1 & -1 \\ -1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 & 1 & -1 \\ -1 & 0 & -1 & 1 \\ 1 & -1 & 0 & -1 \\ -1 & 1 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 & -1 & -1 \\ 1 & 0 & -1 & -1 \\ -1 & -1 & 0 & 1 \\ -1 & -1 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 & 1 & -1 \\ -1 & 0 & -1 & 1 \\ -1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 \end{bmatrix}$$

Fig. 2. Coupling matrices and examples of gait generation for the 4 gaits. $\omega_{stance} = 2\omega_{swing}$ for the trot and pace gaits. $\omega_{stance} = 4\omega_{swing}$ for the walking gait and $\omega_{swing} = 2\omega_{stance}$ for the bound.

We decide to add the feedback on the y_i variables rather than x_i for 2 reasons. First this is the variable that defines if we are in stance ($y_i > 0$) or swing phase ($y_i < 0$) as is shown in Figure 3. Second, since we use the x_i variables as the policy for the trajectories of the limbs, adding the control on the y_i variables assures that the x_i variables will always produce a smooth output.

1) Stopping before transition: We want the oscillator to stop in two cases, during swing to stance transition when the limb is not yet in contact with the ground and during stance to swing transition when the limb still supports significant body weight (i.e. to avoid premature lifting of the limbs).

The stopping behavior is obtained by temporarily activating the following control signal: $u_i = -\omega_i x_i - \sum k_{ij} y_j$. This choice is motivated by the fact that we want the oscillator to stop at the transition, i.e. when $y = 0$ we want $\dot{x} = \dot{y} = 0$. Thus we get $\alpha(\mu - x_i^2 - y_i^2)x_i - \omega_i y_i = 0$ which implies that the fixed point is at $x_i = \pm\sqrt{\mu}$ (the oscillator stops when x_i reaches its maximum). Note that it is possible to change the stopping position by changing μ. For the second equation, we get $\omega_i x_i + \sum k_{ij} y_j + u_i = 0$ which gives us the control $u_i = -\omega_i x_i - \sum k_{ij} y_j$. This control in fact erases the coupling from the other oscillators and removes the oscillating term of the y_i variables.

Linearizing the equation with the new control, we can deduce the stability of the fixed points (at $x = \pm\sqrt{\mu}$, $y = 0$). The Jacobian matrix at these points is

$$Df = \begin{bmatrix} -2\alpha\mu & -\omega_i \\ 0 & 0 \end{bmatrix} \tag{8}$$

its eigenvalues are 0 and $-2\alpha\mu$ with corresponding eigenvectors $(\frac{-\omega_i}{2\alpha\mu}, 1)$ and $(1, 0)$. Thus the stable manifold around the fixed points is tangent to the x axis and the flow cannot cross the x axis and converges automatically to the fixed points. It converges to $-\sqrt{\mu}$ when $y > 0$ and to $\sqrt{\mu}$ when $y < 0$, thus it stops before transition, but once the Ox line crossed it cannot come back.

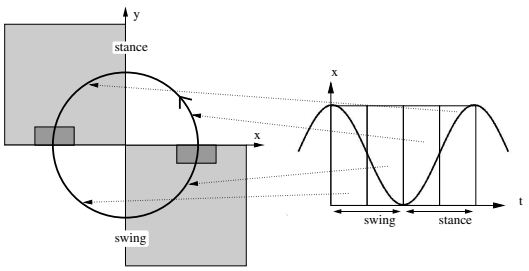


Fig. 3. Phase space of an oscillator (left fig.) with its activation zone for the feedback (light gray for transition and dark gray for the stop controls). Correspondence with the x variable of the oscillator shown on the right.

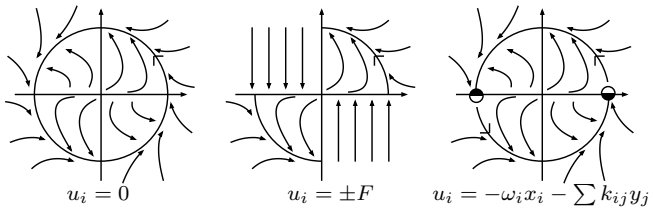


Fig. 4. Schematic oscillator phase plot for the different types of feedback.

2) Fast transitions: We want a fast transition from one phase to the other in two cases: during stance when the weight under the foot becomes low and during swing when the foot touches the ground. To accelerate the transition, what we need to do is to make the y variable go to 0, thus the control for this case becomes $u = F$, the sign of F depending on the direction of transition. Using this control, we can know the time delay before transition that will occur in the system. Indeed, during transition (we consider the transition much faster than the oscillation), we will have $x \simeq cte$ and thus the speed of convergence to $y = 0$ is given by $\dot{y} = \alpha(\mu - x^2)y - \alpha y^2 + \omega_i x + F$ and since F is bigger than the other terms, we have approximately $\dot{y} \simeq F$. Thus the delay before transition will be $\frac{y(t_{transition})}{F}$ seconds, where $t_{transition}$ denotes the time when the transition starts.

The interest of using such a control instead of directly resetting y to 0 is that we can explicitly control the delay before the transition actually occurs. Of course, resetting after some delay would be equivalent, but should be made explicitly outside the equations. In the following we use $F = 300$, which corresponds to a transition of $\simeq 3.3$ ms.

The activation of the feedback for an oscillator depends on its phase and on the pressure sensors information of the corresponding foot as is shown in Figure 3. The phase space of the oscillator in function of the feedback is shown in Figure 4. In summary the feedback goes as

$$u_i = \begin{cases} -\text{sign}(y_i)F & \text{fast transitions} \\ -\omega_i x_i - \sum k_{ij} y_j & \text{stop transition} \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

IV. EXPERIMENTS

We performed experiments[2] on three different simulated quadruped robots having very different mechanic properties (iCub, Aibo and Ghostdog). We used Webots [11], a software for the physic simulation of robots based on ODE, a library for the simulation of rigid body dynamics.

A. Experimental setup

For all the robots, the CPG generates the joints angles of the hip and/or shoulder joints in the sagittal plane. The trajectories for the knees (or elbows) are functions of the x_i variables of the CPG and are defined such that they bend during swing and do not move during stance, to keep the movement as simple as possible.

We used the simulated iCub robot, which is an infant-like humanoid robot currently under construction for the RobotCub project [12]. The robot has the size of a 2-year old infant. It locomotes on the ground on its hand and knees, thus 2 DOFs in the sagittal plane are available for the fore limbs (shoulder and elbow) while there is only one for the hind limbs (hip). For this robot, we also control the stiffness of the elbow joint to allow a smooth landing of the hands (i.e. we reduce the gains of the PID controller of the elbow when the hand touches the ground). This is a very naive approach but it is suitable for our current experiments. We will use more advanced force control techniques for the elbow in future experiments. For this robot we tested a trot gait as real infants do. The second tests were done on a simulation of the Aibo robot. Here walk and trot were tested. Contrary to the iCub robot, we can use the knees of the hind limbs. The final robot is called Ghostdog, which is a simulated robot available in the Webots software. It is a dog-like robot, having motors in the hips and knees. In addition there is a spring in series with the knee joint, which gives to the robot some passive dynamics, which is interesting to test dynamically stable gaits such as bound. We tested walk and bound on this robot.

B. Flat ground experiments

The first set of experiments was done on flat terrain. We systematically tested the locomotion of the robots for different values of ω_{swing} and ω_{stance}. We used the speed of locomotion as a performance criterion. We tested the robots with and without feedback, we also tested them without inter-oscillator coupling but with feedback to test the significance of a controller with only reflexes and no explicit coordination between the limbs. In Figure 5 we show for the iCub the speed of locomotion for the different experiments as a function of the stance and swing parameters. The results for the other robots at slow speeds (walk and trot) are qualitatively the same.

We first observe that for the three experiments, the duration of the swing has almost no influence in the speed of locomotion (vertical stripes in the figures), but it seems that

[2]Refer to the attached video to see typical results of the different experiments described in the article.

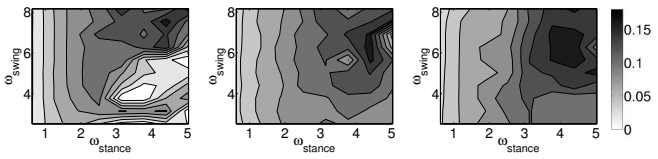


Fig. 5. Speed characterization of the iCub robot for different values of ω_{stance} and ω_{swing}. Left graph: without feedback, middle one: with feedback and without coupling and right one: with both feedback and coupling enabled. The right bar shows the color legend for the speed ($\mathrm{m} \cdot \mathrm{s}^{-1}$), maximum speed is approx. equivalent to 0.6 body length $\cdot\, \mathrm{s}^{-1}$

without feedback, a wrong choice of swing duration implies a strong degradation in performance. On the other hand, we find a clear correlation between speed of locomotion and stance duration, as in mammals. This result is quite important because it proves that the assumption we made in Section II made sense and also shows that ω_{stance} can be used as a control parameter for the speed of locomotion.

The second observation is that without feedback, an incorrect swing parameter could lead to an impressive decrease of performance, while when the feedback is enabled (with or without coupling between the oscillators), the performance does not degrade. This observation shows that the feedback structure makes the controller more robust to parameter choice. This makes sense since the feedback regulates the timing of each phases, correcting thus bad parameters choice.

It must also be noted that it is only when both the feedback and the coupling are enabled that the system makes its best performance. It shows that trying to explicitly keep a correct coordination between the limbs of the robot is important and that a reflex-based controller that would only react to the sensory feedback might not be as efficient as our CPG. For the bound gait of the Ghostdog, the results are a bit different. Indeed, we did not manage to find a correct locomotion speed without the feedback. However, when we enable the feedback, then the robot moves at high speed (ranging between 0.2 to 0.6 $\mathrm{m} \cdot \mathrm{s}^{-1}$ or equivalently from 0.36 to 1.08 body length $\cdot\, \mathrm{s}^{-1}$) for a large range of parameters. This shows that sensory information combined with the CPG can successfully stabilize a gait that is not intrinsically stable.

Another observation is that the feedback structure strongly couples the CPG with the robot it controls. In Figure 6, we show typical trajectories generated with and without feedback by the CPG during walking for the Aibo, the strong entrainment of the system is clear, it makes the controller closer to the intrinsic dynamics of the robot, even if the parameters are not perfectly chosen. This entrainment is even more obvious on the phase space plot, where it is clear that the stance phase is strongly shortened. For certain parameter values, typically faster stance, the walking gait even change into a trot gait by the feedback coupling, showing that the system is able to adapt its gait to the specificity of the robot it controls. Figure 6 shows also typical sensor values from the foot of the Aibo. We clearly notice that the swing and stance phases become more regular when the feedback is enabled. This finally shows the mutual entrainment of both the CPG and the robot, due to the strong coupling. We observe the same qualitative results for all the robots, this coupling being

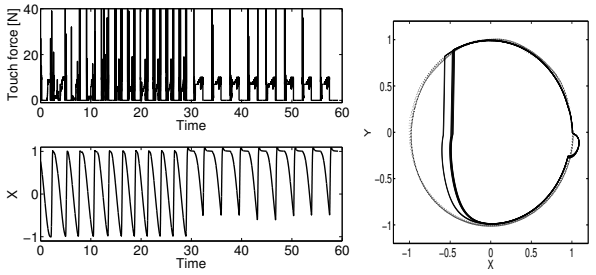


Fig. 6. On the left graphs we show the typical fore left leg foot sensor values and CPG output for the Aibo during walking. From t= 0 to t= 30s there is no feedback in the system. After t= 30 feedback is enabled. The right graph shows the corresponding state space of the oscillator controlling the left fore leg (dotted line without feedback and plain line with feedback).

even more important during bound of the Ghostdog (see attached video).

C. Uneven terrain experiments

Finally we performed experiments with the three robots on non-flat terrains. The robots went up and down terrains with a slope (up to 10 degrees of slope for the iCub and Ghostdog, 5 degrees for the Aibo). In this case again, the feedback was able to stabilize the robots and adapt the phase timing according to the terrain. We also made the robots go down stairs of up to 10cm high (for the iCUb and Ghostdog). The feedback improved the locomotion in this case too, however it happened that the robots fell. One major reason of these falls is that sometimes the robots put their feet on the edges of the stairs, making them slip and fall. Since our feedback relies only on sensory information from the feet pressure sensors and only modulates the swing and stance phases, it cannot cope with such events. Stereotyped reflexes and a higher level balance controller should be developed to cope with such obstacles. Adding vision to the robot would also help planning the robot trajectory and avoid such problems.

V. CONCLUSION

In this contribution we have presented a way of designing CPGs and including feedback in them. We use an oscillator designed for locomotion control, in which we can independently control the duration of swing and stance phases. We showed how insight from dynamical system theory could be used to design the network architecture of the CPG in a simple and scalable manner.

Our major contribution was the integration of sensory feedback in the CPG in a manner such that we control explicitly the behavior of the system and we can change its properties using simple parameter changes. This CPG is able to modulate online the trajectory generation for the control of locomotion. We based our design on the assumption that the CPG and the mechanical system were strongly coupled in both ways. Thus we could think of the CPG as being "controlled" by the sensory feedback, changing the state space of the oscillators accordingly.

Finally we showed the generality of our method by applying the same controller to three very different robots. The resulting locomotion is generally faster and more stable with the feedback structure. The controller is tightly coupled with the intrinsic dynamics of the robots. Our simulations also showed that the swing duration was important for stability and performance while the stance duration could be used as a parameter to control the speed of locomotion. This finding is very interesting because it is closely related to observations made in mammal locomotion and it supports the design assumptions we made in Section II.

However there are still many challenges to design a complete locomotion controller, posture and stability problems are not yet completely solved. Nevertheless we believe that our framework is suitable to include more complex behavior and control. For example it could be easily integrated in a more general controller for precise hand placement during locomotion [13]. Moreover we could use advanced adaptive control techniques to fine tune the frequency parameters of the oscillators, using for example methods from [14].

Future work will include integration of more complex sensory feedback and stereotyped reflexes. We will also study how to integrate more advanced force control techniques to control the individual joints of the robot, using the CPG as a policy generator.

REFERENCES

[1] S. Aoi and K. Tsuchiya, "Locomotion control of a biped robot using nonlinear oscillators," *Autonomous Robots*, vol. 19, pp. 219–232, 2005.

[2] Y. Fukuoka, H. Kimura, and A. Cohen, "Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts," *The International Journal of Robotics Research*, vol. 3–4, pp. 187–202, 2003.

[3] J. Morimoto, G. Endo, J. Nakanishi, D. Bentivegna, and C. Atkeson, "Modulation of simple sinusoidal patterns by a coupled oscillator model for biped walking," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006.

[4] G. Taga, "Emergence of bipedal locomotion through entrainment among the neuro-musculo-skeletal system and the environment," *Physica D: Nonlinear Phenomena*, vol. 75, no. 1–3, pp. 190–208, 1994.

[5] A. Ijspeert, A. Crespi, D. Ryczko, and J.-M. Cabelguen, "From swimming to walking with a salamander robot driven by a spinal cord model," *Science*, vol. 315, no. 5817, pp. 1416–1420, 2007.

[6] S. Frigon and S. Rossignol, "Experiments and models of sensorimotor interactions during locomotion," *Biological Cybernetics*, vol. 95, no. 6, pp. 607–627, 2006.

[7] A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA2002)*, 2002, pp. 1398–1403.

[8] L. Righetti and A. Ijspeert, "Design methodologies for central pattern generators: an application to crawling humanoids," in *Proceedings of Robotics: Science and Systems*, 2006, pp. 191–198.

[9] M. Golubitsky and I. Stewart, "Nonlinear dynamics of network: the groupoid formalism," *Bulletin of the American Mathematical Society*, vol. 43, pp. 305–364, 2006.

[10] M. Golubitsky, I. Stewart, P.-L. Buono, and J. Collins, "A modular network for legged locomotion," *Physica D*, vol. 115, pp. 56–72, 1998.

[11] O. Michel, "Webots: Professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, 2004.

[12] G. Sandini, G. Metta, and D. Vernon, "Robotcub: an open framework for research in embodied cognition," 2004, paper presented at the IEEE-RAS/RJS International Conference on Humanoid Robotics.

[13] S. Degallier, L. Righetti, and A. Ijspeert, "Hand placement during quadruped locomotion in a humanoid robot: A dynamical system approach," in *IEEE-RAS International Conference on Intelligent Robots and Systems (IROS07)*, 2007, accepted.

[14] J. Buchli, F. Iida, and A. Ijspeert, "Finding resonance: Adaptive frequency oscillators for dynamic legged locomotion," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006, pp. 3903–3909.