Online Generation of Cyclic Trajectories Synchronized with Sensor Input

Rodolphe Héliot, Bernard Espiau

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Online Generation of Cyclic Trajectories Synchronized with Sensor Input

Rodolphe Héliot — Bernard Espiau

N° 6101

Novembre 2006

Thèmes NUM et BIO

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6101--FR+ENG

Online Generation of Cyclic Trajectories Synchronized with Sensor Input

Rodolphe Héliot [*,†], Bernard Espiau [†]

Thèmes NUM et BIO — Systèmes numériques et Systèmes biologiques
Projets Bipop et Demar

Rapport de recherche n° 6101 — Novembre 2006 — 25 pages

Abstract: In this paper, we adress the issue of online trajectory generation, frame in which we focus on the following problem: how to synchronize the generated trajectory with a given cyclic sensor measurement? We model our measurement with a nonlinear oscillator, and build an observer of it; therefore, the system is adapted to its specific input. Then, we can estimate the oscillator phase, and from it generate a multidimensional trajectory at low computational cost. We finally evaluate the method and assess its robustness to error in parameter estimation, and to input changes.

Key-words: Online trajectory generation - Oscillator - Synchronization - Observer - Sensor

[*] CEA-LETI, 17 rue des Martyrs, 38054 Grenoble - France.
[†] INRIA Rhône Alpes, 655 av. de l'europe, 38334 Saint Ismier - France.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38334 Montbonnot Saint Ismier (France)
Téléphone : +33 4 76 61 52 00 — Télécopie +33 4 76 61 52 52

Generation en ligne de trajectoire synchronisée à une entrée capteur

Résumé : On s'intéresse à la question suivante : comment synchroniser en ligne une commande cyclique avec un signal capteur lui aussi cyclique ? Nous modélisons notre mesure à l'aide d'un oscillateur non-lineaire, et en construisons un observateur. Ensuite, on vient "filtrer" les mesures capteur à l'aide de l'observateur. Ce dernier étant également un oscillateur, il est possible d'estimer sa phase, et de générer une commande paramètrée temporellement par cette phase. Nous évaluons la méthode et caractérisons sa robustesse aux erreurs d'estimation des paramètres, ainsi qu'aux changements de rythme du signal d'entrée. Finallement, une application de cette méthode à la réhabilitation fonctionnelle des patients hémiplégiques est présentée.

Mots-clés : Génération de trajectoire en ligne - Oscillateur - Synchronisation - Observateur - Capteur

1 Introduction

The problem of trajectory generation is basic as well as classical in robotics. Although there exists now efficent industrial solutions to generate complex trajectories for manipulator robots, allowing for example to avoid obstacles, or to optimize energy consumption or cycle time, the problem is still a research one for some classes of mobile robots. In fact, the realisability of precomputed trajectories may become questionable when it is needed to cope with control requirements or to adapt in real time to unexpected changes. Non-holonomy of wheeled robots, and dynamical walking stability of legged robots are two examples of such difficulties.

In the case of bipedal systems, addressed in this paper, trajectory generation may be splitted in two parts: 1- path planning of the whole system: this problem is close to the one of mobile robots, except when an accurate determination of footprints is required; 2- computation of leg trajectories preserving some invariants in the posture and ensuring a certain form of stability. Focusing on this second issue, it can be seen that several classes of methods can be found in the literature (an overview of bipedal trajectory generation techniques is presented in [AAEA04]), among them: computation of optimal trajectories in various spaces, derivation of intrinsic trajectories in passive walking, or use of model predictive control... More precisely, when periodic motions are considered, a frequently used approach consists in mimicking living being's CPGs (Central Pattern Generator), under the form, either of ANNs, or of nonlinear oscillators. Originally designed for very stable systems, like 6- or more legged systems, snake or fish robots, etc... (see for example [Ijs01]), this approach has more rarely been used for biped robots [EMNC02]. However, to our knowledge, the problem of using this class of method, in conjunction with sensory inputs delivered in real-time, has not yet be addressed. This is precisely the goal of this paper. Indeed, we consider the case where we want to generate the cyclic trajectories of selected links which have to be synchronous with the output of a sensor mounted on another link. This is a kind of teleoperation, which can be, for example, very useful in the case of controlling a lef prosthesis or remotely operating a walking robot.

In this paper we will focus on the case of bipedal systems and, therefore, consider the following situation: a sensor, installed on the thigh of a human, provides at each time with an information related to the absolute angular position of the link; the question which arises is then: how to use this single information to generate the full trajectories of another system (a robot or the other leg, for example), in a perfectly synchronous way? We will see later that the problem is addressed by designing a specific conjunction of a nonlinear oscillator and of the related observer. The paper is organized as follows: after having recalled some basic facts on nonlinear oscillators required in the following, we will discuss the choice of the oscillator, and derive the associated nonlinear observer. Then, we will analyze the properties of the designed system, and give some simulation and experimental results. Finally, we will draw some plans for the near future.

2 Framework

2.1 Oscillators: some definitions

In this section, we recall some basic facts concerning periodic solutions of ordinary differential equations (ODE). All this section is largely inspired from the book of Pikowsky et al. [PRK01]: "*Synchronization, a universal concept in nonlinear sciences*", part 7.1: phase dynamics. The concepts of phase and isochrones are defined, and will be used in the following.

Let's consider a system of autonomous ordinary differential equations:

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}), \qquad \mathbf{x} \in \Re^n \tag{1}$$

and suppose that this system has a stable periodic (with a period T_0) solution $\mathbf{x}_0(t) = \mathbf{x}_0(t + T_0)$. In the phase space (space in which all possible states are represented) this solution is an isolated closed attractive trajectory, called the limit cycle of eq. (1) (Fig. 1). A classical example of a self-oscillating system is the van der Pol equation:

$$\ddot{x} - \mu(1 - x^2)\dot{x} + \omega_0^2 x = 0 \tag{2}$$

with $\mu > 0$ and $\omega_0 > 0$.

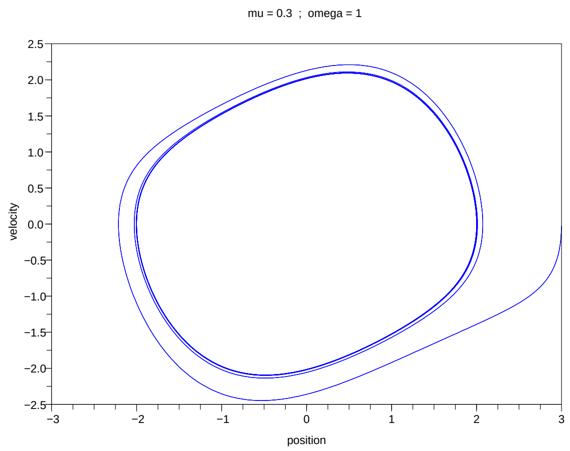

Figure 1: Limit cycle of the van der Pol oscillator. Here $\mu = 0.3$ and $\omega_0 = 1$.

Let's introduce the phase φ as a coordinate along the limit cycle, such that it grows uniformly in the direction of the motion and gains 2π during each rotation, thus obeying the equation:

$$\frac{d\varphi}{dt} = \omega_0 \tag{3}$$

where $\omega_0 = 2\pi/T_0$ is the frequency of the self-sustained oscillations.

From eq. 3 follows an important property of the phase: it is a neutrally stable variable, in the sense that a perturbation in the phase remains constant: it never grows or decays in time.

Consider now the effect of a small external periodic input on the self-sustained oscillations, described by:

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}) + \epsilon\mathbf{p}(\mathbf{x}, t) \tag{4}$$

where the input $\epsilon\mathbf{p}(\mathbf{x}, t) = \epsilon\mathbf{p}(\mathbf{x}, t+T)$ has a period T, which is in general different from T_0. The input is proportional to a small parameter ϵ, and below we consider only first-order effects in ϵ. The external perturbation drives the trajectory away from the limit cycle, but because it is small and the cycle is stable, the trajectory only slightly deviates from the original one $\mathbf{x}_0(t)$. Thus perturbations in the directions transverse to the limit cycle are small; contrary to this, the phase perturbations can be large.

A need is then to define the phase variable in such a way that it rotates uniformly according to eq. 3 not only on the cycle, but in its neighbourhood as well. To this end, we define the so-called *isochrones* in the vicinity of the limit cycle [GH90]. Observing the dynamical system stroboscopically, with the time interval being exactly the period of the limit cycle T_0, we get a mapping:

$$\mathbf{x}(t) \to \mathbf{x}(t + T_0) \equiv \Phi(\mathbf{x})$$

This construction is illustrated by the figure 2. Let us choose a point $\mathbf{P}^*$ on the cycle and consider all the points in the vicinity that are attracted to $\mathbf{P}^*$ under the action of Φ. They form a $(M-1)$-dimensional hypersurface I called an isochrone, crossing the limit cycle at $\mathbf{x}^*$. An isochrone can be drawn at each point of the limit cycle, thus we can parametrize the hypersurface according to the phase as $I(\phi)$ (see Fig. 2). We now extend the definition of the phase to the vicinity of the limit cycle, demanding that the phase is constant on each isochrone. In this way, phase can be defined in the neighbourhood of the limit cycle.

2.2 CPG for trajectory generation

Recently, the concept of Central Pattern Generator (CPG) hase been used in robotics for online trajectory generation [Wil98, RA06, Tag00, EMNC02]. The CPG concepts comes from biology [Gri85, CRG88]: it is a small neural network, located at the spinal level, able to generate rhythmic commands for the muscles. It can be divided into two parts: a rhythm generator, and an patterning mechanism [PPS98]. CPGs receive inputs from higher parts of the central nervous system, and also from peripheral afferents; thus, its functioning results from an interaction between central commands and local reflexes (see fig 3).

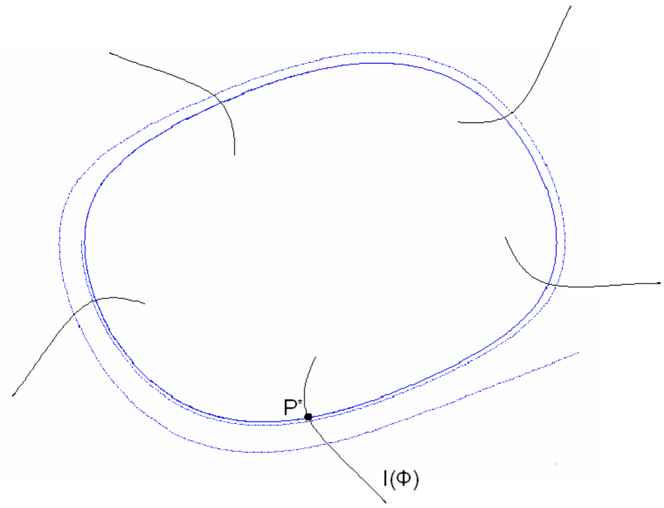

Figure 2: Isochrones in the vicinity of the limit cycle

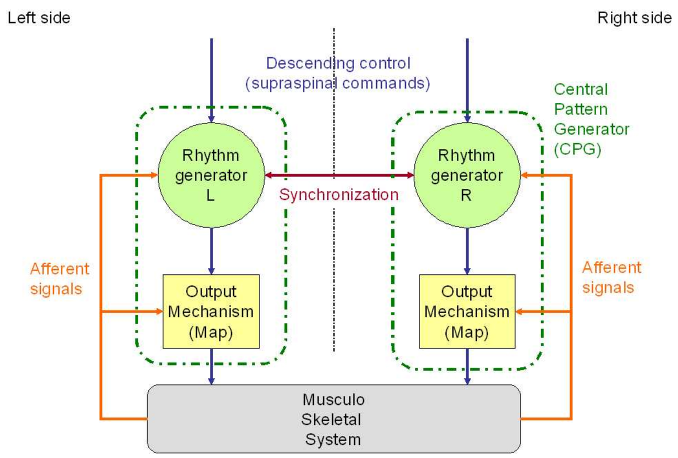

Figure 3: CPG architecture for movement control in vertebrates

The CPG can be modeled either with a simulated neural network (for example using the Fitzhugh-Nagumo model [Mat85]) or a non-linear oscillator. In both cases, the idea is to encode the desired trajectory in a stable limit cycle.

In robotics, a CPG-bases command structure has several advantages for the design of cyclic trajectories: the system is stable against small perturbations, thanks to the intrinsic stability of the limit cycle; one can easily modulate the amplitude or the period of the trajectory; it is well suited for feedback integration [DZZZ06]. Finally, a multidimensional output can be generated for the same low computational price, which is helpful when dealing with robots with numerous Degrees Of Freedom (DOFs), having to exhibit multiple synchronized perodic motions, like walking machines. The Fig. 4 control scheme gives an example of a multi-DOFs command structure.

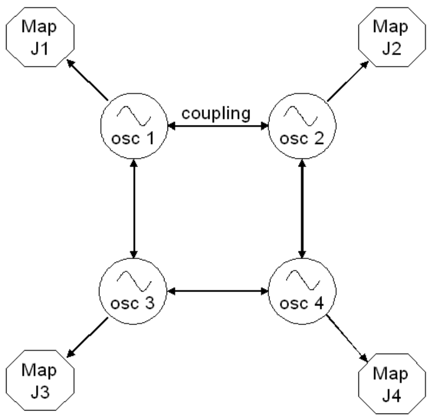

Figure 4: This schematics gives an example of a joints command structure. Four coupled oscillators (osc 1 .. osc 4) produce rhythm for the output mechanisms of the four joints J1 .. J4

In this paper, we will focus on the input integration problem in the CPG concept: " How can we build a rhythm generator (oscillator) such as we can be *sure* that it will synchronize with a given input ?"

2.3 Synchronization and observation

The synchronization problem has been of great interest recently [CP91, PRK01] and a link has been made between synchronization and observation theories, mostly by Nijmeijer and his group [NM97, BFNP97]. The observer theory, coming from the control theory, has been introduced in the early seventies by Luenberger [Lue71] in the linear case; in the non-linear case, some partial results exist [Isi95].
The idea of observation is to estimate the state variables of a system, only given the inputs and the outputs of the system. Let's consider the system

$$\Sigma : \begin{cases} \dot{x} = f(x) + g(u) \\ y = h\,(x) \end{cases} \tag{5}$$

and build a copy of Σ with *output injection* (Fig. 5):

$$\Sigma' : \begin{cases} \dot{\hat{x}} = f(\hat{x}) + g(u) + K(\hat{y} - y) \\ \hat{y} = h(\hat{x}) \end{cases} \tag{6}$$

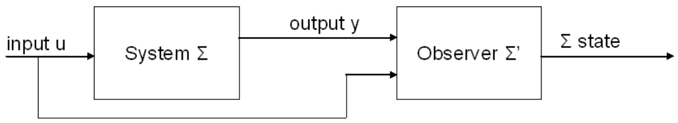

Figure 5: Observer principle

In the linear case, if the original system Σ is observable (see [Kai80] for a complete description of the observability conditions), and if gain K is correctly set, then the observer state will converge towards the original system state. When the output error $(\hat{y} - y)$ is canceled, the observer state exactly matches Σ's state: the observer is synchronized with the observed system.

In the non-linear case, there is no general result concerning the observer existence. However, it is sometimes possible to build a non-linear observer, when the error dynamics is feedback linearizable. To achieve this, the system has to belong to the *Lur'e* class [LP44], in which the non-linearity is a function of the output only:

$$\Sigma : \begin{cases} \dot{x} = Ax + f(y,t) + Bu \\ y = Cx \end{cases} \quad x \in \Re^n, u \in \Re^m, y \in \Re^p \tag{7}$$

The observer is then given by:

$$\Sigma' : \begin{cases} \dot{\hat{x}} = A\hat{x} + f(y,t) + Bu + K(\hat{y} - y) \\ \hat{y} = C\hat{x} \end{cases} \tag{8}$$

and the error dynamics can be linearized:

$$\begin{array}{ll} & e = \hat{x} - x, \\ with & \begin{aligned} \dot{e} &= \dot{\hat{x}} - \dot{x} \\ &= A\hat{x} + f(y,t) + Bu + K(\hat{y} - y) - Ax - f(y,t) - Bu \\ &= (A + KC)e \end{aligned} \end{array} \tag{9}$$

2.4 Synthesis : our approach

Our aim is to build an oscillator which will synchronize with a given cyclic sensory input. If we want to use the observer theory, we need to have a model of the system. However, sometimes, there is no model available. This leads us to propose the following method, into two parts (Fig. 6):

1. build a system as a phenomenological model, which simulates the sensor measurements
2. build an observer of this system, in which are injected the real sensor measurements

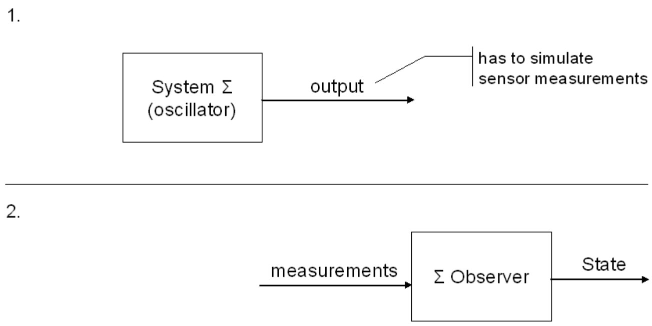

Figure 6: Schematics of the method

To simulate a cyclical sensor measurement, one could choose a linear system as model, and provide it with a cyclical input u (for example, a sinusoidal input); in that case the linear system is shaping the input so as its output simulates the given measurement. A problem arises then: there is a need for providing with a cyclical input which has to be synchronous with the measurement. This problem has been explored [RBI05], but the proposed solutions adapt to frequency changes too slowly for our application.
For this reason, we chose a non-linear oscillator as phenomenological model for our sensor measurement: it can **autonomously** (without input) generate a **cyclical** signal.

3 Methods

We previously presented our approach in a general way. In this section, we will focus on one application of this method: human gait observation using a sensor placed on a leg.

3.1 Which oscillator and why ?

3.1.1 What could be the right form of an oscillator?

To some extent, and under the assumption of rigidity, a bipedal walking system can be modelled as a tree-structured n-link mechanical system free in space. Its dynamics can therefore be described through a Lagrange equation:

$$M(q)\ddot{q} + N(q,\dot{q}) + G(q) = -B(\dot{q}) + \Gamma + \lambda^T C(q) \tag{10}$$

where q is the set of joint coordinates, $q \in \Re^n \otimes SE(3)$, M is a sdp. mass matrix, N gathers coriolis and centrifugal forces, B is the friction term, G the gravity vector, Γ the actuation input, and $\lambda^T C(q)$ are the constraints of ground contacts, which are unilateral and time-varying. In the absence of constraints, friction and control, this equation becomes autonomous (i.e. with a right-hand side equal to zero), with mechanical energy as first integral, in which the continual exchange between kinetics energy and potential energy produces a periodic motion.

Let us now consider a single coordinate q_a, which can be for example the thigh angle. Starting from the autonomous version of eq. (10), we can express its dynamics as:

$$H(.)\ddot{q_a} + F(q_a, .) = T(.) \tag{11}$$

where T is a set of bounded perturbations depending on all the variables and their derivatives, $H(.)$ the equivalent of a mass term, F is analogous to a potential function. Therefore, the behaviour of q_a is the one of a periodic solution, issued from a nonlinear second-order equation, with a potential term and disturbances. This incitates to research the nonlinear oscillator preferably within the class of modified and disturbed second-order mass-spring systems.

3.1.2 Is the concept of limit cycle licit for human walking?

As seen previously, the natural behaviour of mechanical robotics system without dissipative and other inputs is an oscillator with constant energy. Nevertheless, this does not correspond with the idea of an attractive limit cycle which underlies the oscillator-based approach. To justify this point of view, we have to refer to another class of mechanical systems: the passive walking machines. Indeed, let us consider the case of a planar compass, walking above a slope, with instantaneous an inelastic step transition, as addressed in [McG90] and several others [GTE98, CWR01]. It then can be shown that, for a given slope, such a system exhibits a limit walking cycle, with a rather large basin of attraction (see Fig. 7). This behaviour can be compared to the concept of "natural gait" or "comfort gait" which is spontaneously reached and followed by a human in steady state walking, and which corresponds to a minimum of the metabolic energy consumption with respect to distance.

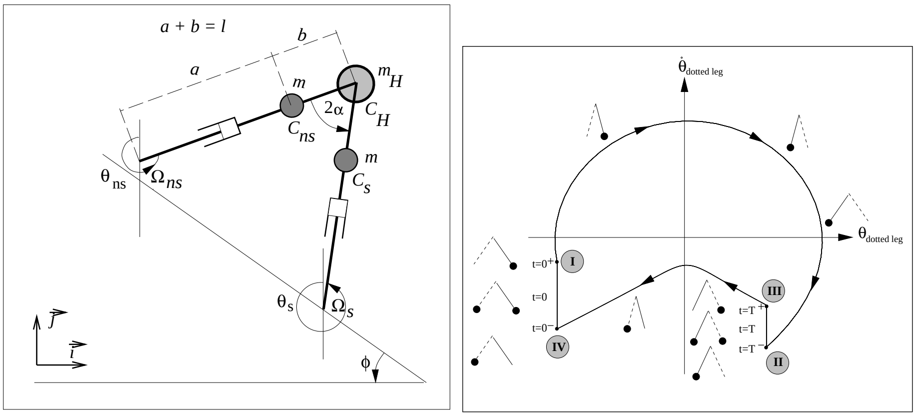

Figure 7: Mechanical model of a passive walking machine, and its stable limit cycle

In conclusion, it appears that searching for a oscillator of second-order type and exhibiting a limit cycle is a natural way of modelling the periodic walking behaviour of a human link measured with an adequate sensor.

3.1.3 Structure choice and parameters setting

From the previous sections, we know that we have to choose a nonlinear oscillator which belongs to the lur'e class (defined earlier in section 2.3), and which is derived from a second-order mass-spring system. Two common oscillators fit these requirements: the van der Pol oscillator and the Rayleigh oscillator, which are very similar.
So we start from the van der Pol eqation in order to simulate our input. However, depending on the sensor measurement, it has to be slightly modified to be able to correctly simulate it. In our case, we are interested in bipedal walking, thus a natural idea is to use the leg "position": for example, we could use a measure of joint coordinate, such as knee angle. Another possibility is to use the absolute orientation of one link; for practical reasons due to our sensors (see section 4.1), we chose the thigh inclination with regards to vertical. During humain gait, this inclination presents a dissymetrical pattern, with a ascending phase shorter than the descending one. The van der Pol equation provides with symmetrical signals; thus, we have to introduce a new term in the equation:

$$\ddot{x} - \mu(1\text{- }\mathbf{b}\ \mathbf{x} - x^2)\dot{x} + \omega_0^2 x = 0 \tag{12}$$

with $b > 0$.

The idea is to modify the damping coefficient $\mu(1 - bx - x^2)$ so as it is different when $x < 0$ or $x > 0$. In that way, the output of the modified van der Pol oscillator won't be symetrical anymore: for a given $|x|$ when $x < 0$, $|\mu(1 - bx - x^2)|$ is higher than when $x > 0$.

Once the structure of the nonlinear oscillator is chosen, we have to find the best parameters μ, b, and ω_0 so that the trajectory of the limit cycle of this oscillator will fit the sensor measurement. We write this identification as a least squares problem: minimizing the error between the measurements and the output of the oscillator:

$$\begin{cases} \min\limits_{\mu,b,\omega_0,x_s^i} \quad \sum\limits_{i=1}^{m} (x_s^i - x_m^i)^2 \\ \\ \ddot{x}_s^i - \mu(1 - b x_s^i - x_s^{i^2})\dot{x}_s^i + \omega_0^2 x_s^i = 0 \end{cases} \tag{13}$$

where x_m^i are the discretized sensor measurements (for example, over one given cycle), and x_s^i are the simulated oscillator outputs, thus following the dynamics of eq. (12).

One can notice that this problem is similar to an optimal control problem, that can be solved using a direct method [Bet97]. We add the discrete output of the oscillator in the parameters to optimize, and we add constraints on them following the dynamical model of the oscillator. The discretization of this problem leads to a "nonlinear programming" problem which has been solved using a successive quadratic programming solver (FSQP [LZT97]). Practically, this method gave good results: we obtained a very good match between the measurements and the oscillator output (see fig. 8).

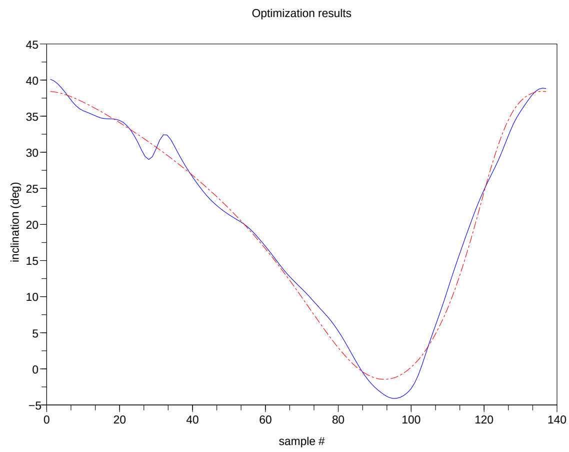

Figure 8: Comparison of the sensor measurment cycle (solid line) with the optimized oscillator output (dotted line)

3.2 Observer design

We want to build an observer of the dynamical system described by the modified van der Pol equation (12), which can be written as:

$$\Sigma : \begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = \mu(1 - bx_1 - x_1^2)x_2 - \omega_0^2 x_1 \\ y = x_1 \end{cases} \tag{14}$$

Let's introduce the variable z:

$$z \ = \ x_2 \ + \ k_1 y \ + \ k_2 y^2 \ + \ k_3 y^3 \tag{15}$$

which satisfies the following equation:

$$\begin{aligned} \dot{z} &= \dot{x_2} + k_1 \dot{y} + 2k_2 \dot{y} y + 3k_3 \dot{y} y^2 \\ \dot{z} &= \dot{x_2} + k_1 \dot{x_1} + 2k_2 \dot{x_1} x_1 + 3k_3 \dot{x_1} x_1^2 \\ \dot{z} &= \dot{x_2} + k_1 x_2 + 2k_2 x_2 x_1 + 3k_3 x_2 x_1^2 \\ \dot{z} &= \mu(1 - bx_1 - x_1^2)x_2 - \omega_0^2 x_1 + k_1 x_2 + 2k_2 x_2 x_1 + 3k_3 x_2 x_1^2 \\ \dot{z} &= (\mu + k_1)x_2 + (2k_2 - \mu b)x_1 x_2 + (3k_3 - \mu)x_1^2 x_2 - \omega_0^2 \, x_1 \end{aligned} \tag{16}$$

so, setting:

$$\begin{array}{lcl} \mu + k_1 = -1 & \rightarrow & k_1 = -\mu - 1 \\ 2k_2 - \mu b = 0 & \rightarrow & k_2 = \mu b/2 \\ 3k_3 - \mu = 0 & \rightarrow & k_3 = \mu/3 \end{array} \tag{17}$$

we get:

$$\begin{aligned} \dot{z} &= -x_2 - \omega_0^2 x_1 \\ \dot{z} &= -(z - k_1 y - k_2 y^2 - k_3 y^3) - \omega_0^2 x_1 \\ \dot{z} &= -z \ + \ (k_1 \ - \ \omega_0^2)y \ + \ k_2 y^2 \ + \ k_3 y^3 \end{aligned} \tag{18}$$

So, z dynamics is fully described by its single eigenvalue, here setted through k_1 to -1. Finally:

$$\Sigma' : \begin{cases} \hat{x}_1 \ = \ y \\ \hat{x}_2 \ = \ z - \ k_1 y \ - \ k_2 y^2 \ - \ k_3 y^3 \\ \hat{y} \ = \ \hat{x}_1 \end{cases} \tag{19}$$

3.3 Trajectory generation

Injecting a measurement input in the observer (19), we get an estimation of the two state variables $\hat{x}_1$ and $\hat{x}_2$. Since this observer is also an (forced) oscillator, we can compute its phase. This can be easily done using the isochrones defined in section 2.1. Isochrones can be computed, using the free nonlinear oscillator equation (12), in two different ways. The first idea, analytical, is to write the oscillator equation in polar coordinates (R, θ), define the phase φ such as it grows uniformly, and compute the lines of constant phase on the (R, θ) plane.

The second idea is to obtain them by simulation; first, let's assess the free oscillator period T_0. Then, for each point $\mathbf{x}_i$ on the phase plane which is in the vicinity of the limit cycle, simulate its trajectory under oscillator dynamics during a time nT_0, with n being an integer large enough such that the distance from the point $\mathbf{x}_i(nT_0)$ to the limit cycle is small. In that case, the original point $\mathbf{x}_i$ has the same phase as $\mathbf{x}_i(nT_0)$, which is known, since it is on the limit cycle.

Finally, let's say that the (cyclical) trajectory we want to generate is parametrized by its phase: we thus have a trajectory pattern $T(\varphi)$, for $\varphi \in [0, 2\pi]$.

The online computation scheme for trajectory generation is the following:

1. inject the sensor measurement y in the adapted observer
2. from the observer state variables x_i, compute the phase φ of the oscillator
3. provide with the command trajectory: $C = T(\varphi)$

and is embedded in the global scheme presented in Fig. 9:

4 Implementation and Results

4.1 Angle estimation

We estimate the thigh inclination with a micro-sensor, developed by CEA-LETI (Grenoble, France), which associates 3 accelerometers and 3 magnetometers in a minimal volume (see Fig. 10). This attitude sensor is able, through the processing algorithms associated, to rebuild the orientation in space of the segment to which it is attached [BHar, NBC+04]. Figure 11 shows an example of thigh inclination estimation during human gait, which will be used as input in the following.

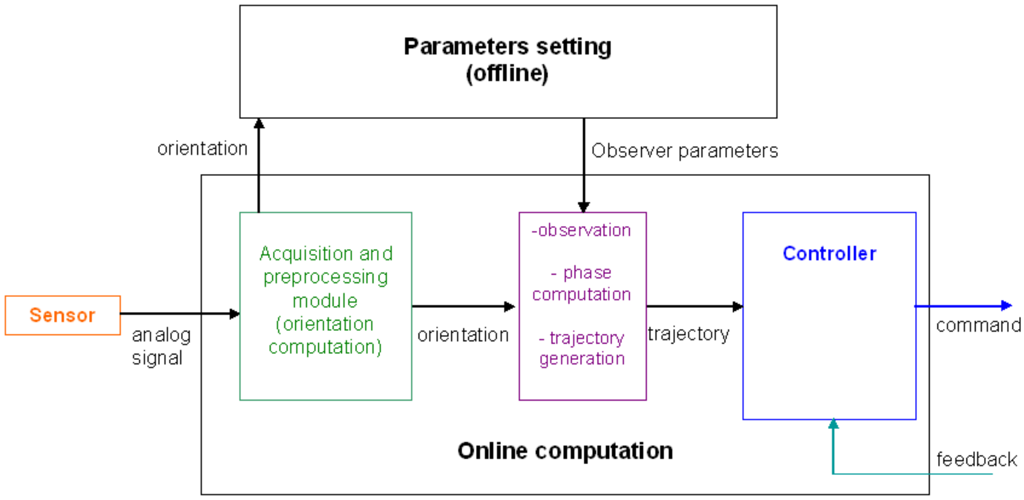

Figure 9: Overall schematics of our method

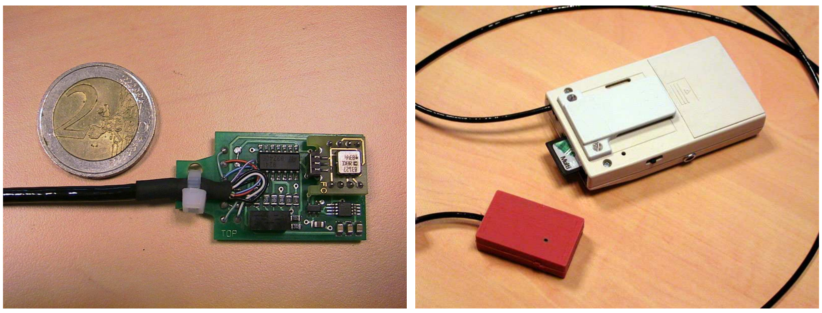

Figure 10: CEA-LETI attitude sensor. Left: Sensor size, compared to a coin. Board is then embedded in a silicon-like material. Right: Final view of the sensor, with its datalogger.

4.2 Results

To validate our method, we recorded sensor measurement during humain gait, in order to generate a synchronous command. This was achieved in a first time offline (in simulation but using real measurements, in order to have an insight on generated trajectory). Figure 12 shows three consecutive measurement cycles, together with the reference limit cycle:

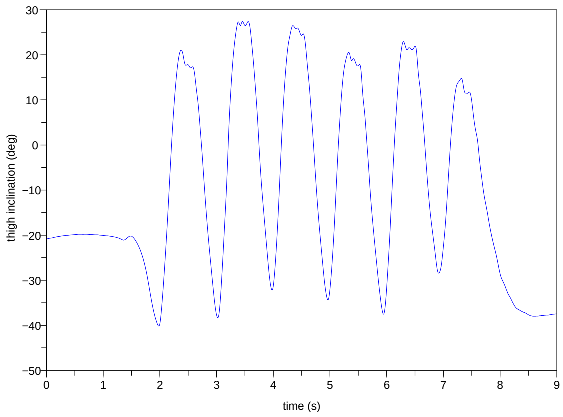

Figure 11: Thigh inclination estimation during human gait, using the attitude sensor

although the estimated state variables of the oscillator do not always lye on the limit cycle, they remain close to it.

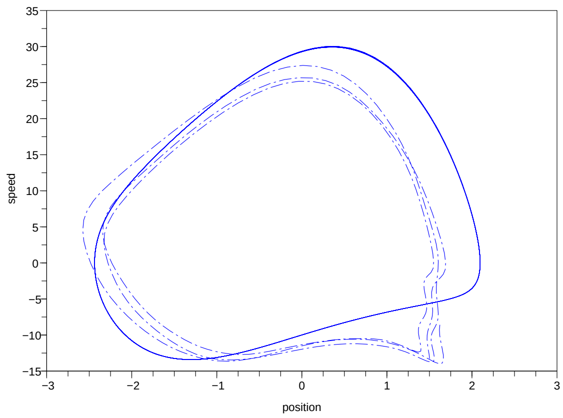

Figure 12: v

Combined with the ischrones-based estimation of phase, we get a good phase estimation (i.e. monotonous, and quasi- piecewise linear), well synchronous with the input signal. Finally, we can generate a command parametrized by this phase variable. Without loss of generality, this command can be the joint coordinates of a poly-articulated system. Such a system has several Degrees Of Freedom (DOFs) which have to be well coordinated, even more in the case of walking systems. A phase parametrization of the desired trajectory on each joint ensures coordination of the overall movement. An example of generated trajectory for one DOF is shown in Fig. 13.

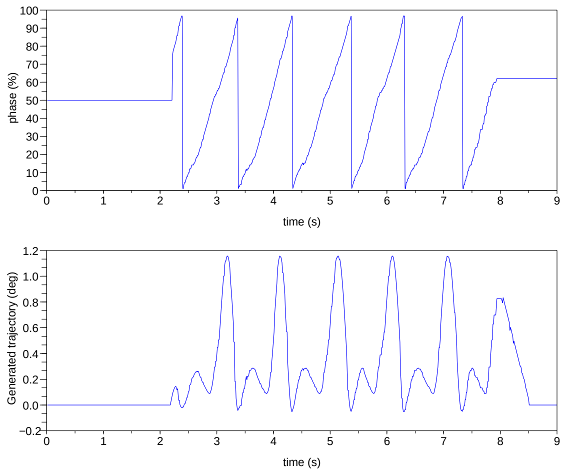

Figure 13: Phase (top) and trajectory generated for one DOF (bottom)

4.3 Experiment: Robot teleoperation

By installing the sensor on the leg of a human, we observe the thigh angle and compute online a biped robot command, such as the robot "follows" the human gait, in a synchronous way. This is done by first generating a desired trajectory for each active DOF of the robot (in our case: ankle, knee and hip sagittal angles on both legs), and then following this trajectory with a PID controller. Such experiments were conducted on the BIP robot (Fig. 14, [AtBt00]), the robot being hanged. We thus fully validated the online trajectory generation based on sensor measurement.

5 Evaluation of the method

In this paper, we consider the use of a nonlinear oscillator in the framework of interaction with human, through actual sensor inputs. It is therefore necessary to assess the practical efficiency of the method. We consider in the following three important issues:

- the theoretical behavior of the modified oscillator in terms of periodic solutions

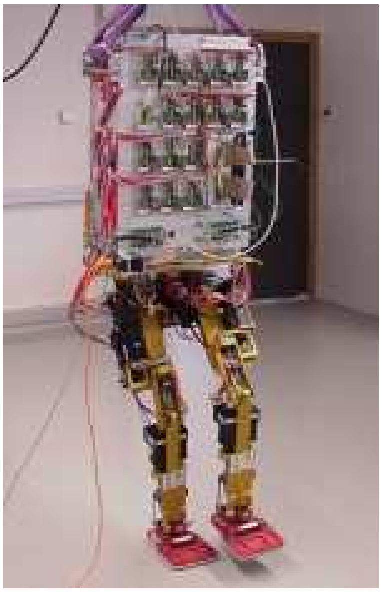

Figure 14: The BIP robot

- the robustness of the approach with respect to errors in parameters
- the ability of the method to track changes in the input dynamics, which allows in particular to cope with transient walking stages

5.1 Robustness

5.1.1 Oscillator properties

We again start from eq. (12), which can be written:

$$\begin{aligned} \dot{x} &= y = f_1(x,y) \\ \dot{y} &= \mu(1 - bx - x^2)y - \omega_0^2 x = f_2(x,y) \end{aligned} \tag{20}$$

with μ, b, and ω all > 0.

Fixed points can be computed:

$$\begin{aligned} \dot{x} &= 0 \rightarrow y = 0 \\ \dot{y} &= 0 \rightarrow x = 0 \end{aligned} \tag{21}$$

There is thus a single fixed point $(x,y) = (0,0)$. Let's study is stability through its Jacobian:

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \mu(-2x-b)y - \omega_0^2 & \mu(1-x^2-bx) \end{pmatrix} \tag{22}$$

which gives, in $(0,0)$:

$$J(0,0) = \begin{pmatrix} O & 1 \\ -\omega_0^2 & \mu \end{pmatrix} \tag{23}$$

The Jacobian trace determines the system behaviour. Here, as $Tr\,(J(0,0)) = \mu > 0$, $(0,0)$ is a repulsive point. Let's express the eigenvalues:

$$\begin{aligned} &\lambda^2 - \mu\lambda + \omega_0^2 = 0 \\ &\Delta = \mu^2 - 4\,\omega_0^2 \end{aligned} \tag{24}$$

And so, using the Poincaré - Bendixson Theorem [GH90]:

- if $\mu < 2\omega_0$ then $\Delta < 0$, and $(0,0)$ is a repulsive source
- if $\mu > 2\omega_0$ then $\Delta > 0$, and $(0,0)$ is an unstable node

In our experiements, the optimized parameters were: $\mu = 2.03$, $b = 2.29$, $\omega = 5.34$. We thus had $\mu = 2.03 < 2\,\omega_0 = 10.68$, far from the bifurcation.

5.1.2 Parameters sensitivity

We now focus on the consequences of an error in parameters estimation. This is an important issue, since in real experiments sensor measurements cannot be exactly the same for a trial to another, even from a cycle to another. Practically, the estimated parameters always are slightly different from what they should be.

Let's so consider the case where our estimation of parameter μ is different from its "true" value: $\hat{\mu} = \mu + \epsilon$. Then:

$$
\begin{array}{ccccc}
\hat{\mu} + k_1 = -1 & \rightarrow & k_1 = -\hat{\mu} - 1 & \rightarrow & \mu + k_1 = -1 - \epsilon \\
2k_2 - \hat{\mu}b = 0 & \rightarrow & k_2 = \hat{\mu}b/2 & \rightarrow & 2k_2 - \mu b = -\epsilon b \\
3k_3 - \hat{\mu} = 0 & \rightarrow & k_3 = \hat{\mu}/3 & \rightarrow & 3k_3 - \mu = -\epsilon
\end{array}
$$

and:

$$
\begin{aligned}
\dot{z} &= (-1-\epsilon)x_2 - \epsilon b x_1 x_2 - \epsilon x_1^2 x_2 - \omega_0^2 x_1 \\
\dot{z} &= (-1-\epsilon)(z - k_1 y - k_2 y^2 - k_3 y^3) \\
&\quad -\epsilon b y(z - k_1 y - k_2 y^2 - k_3 y^3) \\
&\quad -\epsilon y^2 (z - k_1 y - k_2 y^2 - k_3 y^3) \\
&\quad -\omega_0^2 y \\
\dot{z} &= \left[-1 - \epsilon(1 + by + y^2)\right] \cdot z \\
&\quad + \left[(1+\epsilon)k_1 - \omega_0^2\right] \cdot y \\
&\quad + \left[(1+\epsilon)k_2 + \epsilon b k_1\right] \cdot y^2 \\
&\quad + \left[(1+\epsilon)k_3 + \epsilon b k_2 + \epsilon k_1\right] \cdot y^3 \\
&\quad + \left[+\epsilon b k_3 + \epsilon k_2\right] \cdot y^4 \\
&\quad + \epsilon k_3 \cdot y^5
\end{aligned}
$$

In the oscillator dynamics, the error can be thus expressed by:

$$
\begin{aligned}
e = \dot{z}_{\hat{\mu}} - \dot{z}_{\mu} = &-\epsilon(1 + by + y^2) \cdot z \\
&+\epsilon k_1 \cdot y \\
&+\epsilon \left[k_2 + b k_1\right] \cdot y^2 \\
&+\epsilon \left[k_3 + b k_2 + k_1\right] \cdot y^3 \\
&+\epsilon \left[b k_3 + k_2\right] \cdot y^4 \\
&+\epsilon k_3 \cdot y^5
\end{aligned}
$$

It thus appears that the error is linear with respect to the parameter estimation error ϵ. A similar demonstration can be held for each parameter.

We also tested in simulation the effects of a wrong parameter estimation. We compared the observer output $\hat{y}$ when parameters have error with respect to the nominal output. Figure 15 presents the error in $\hat{y}$ estimation with an estimation error on μ, b, or ω_0.

Two remarks can be made: first, the output error is linear with respect to the parameter error, and so, whatever the parameter is. Second, the minimum output estimation error is found for 0% parameter error; this result was of course expected, and means that the parameters have been well estimated.

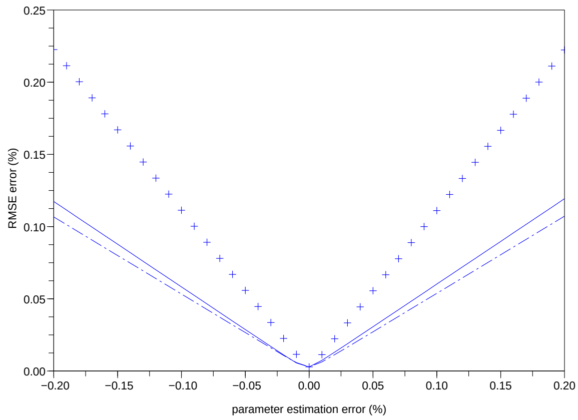

Figure 15: The 3 lines show the output error (in % RMSE) with respect to the 3 parameters; solid line stands for b estimation error, dotted line for μ, crossed line for ω

5.2 Transitions

We simulated the effects of a quasi-step input frequency change. The results show very good adaptation: the generated output follows frequency change without distinguishable disturbance (see Fig. 16).

5.3 Some comments about experimental results

Let us note that in the case of a practical experiment, where the sensor is placed on a real person's thigh, teleoperating a robot, it is impossible to have access to the "actual" intrinsic phase of the observed system (the human). For this reason, is it difficult to evaluate the performance and robustness of our approach with respect to real values that are unknown. This is why quantitative results were computed through simulation, while the presented experimental are qualitative, since they cannot be evaluated more accurately.

6 Conclusion

In this paper, we proposed a method to synchronize a command with a given sensor measurement, based on a nonlinear oscillator observer. We assessed the robustness properties of this method, and demonstrated that it is suitable for human gait observation.

A first application we consider is hemiplegic stroke patients' gait rehabilitation. Hemiplegia induces a situation where a vertical half of patient's body sensory-motor pathways

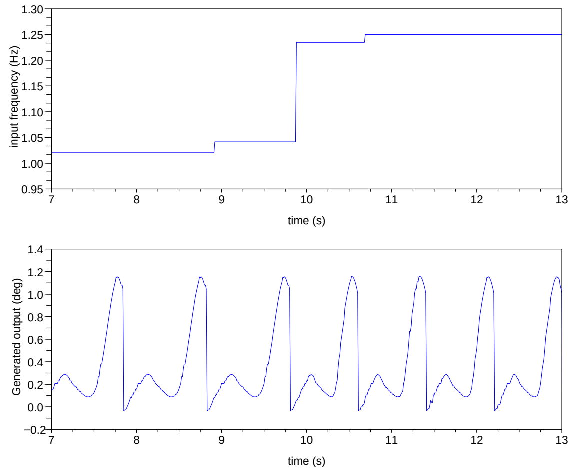

Figure 16: Frequency change adaptation. Top: Input signal frequency; bottom: generated command

do not function properly. A direct consequence is that one side of patient is weak or paralyzed, thus severely impairing walking. Training of hemiplegia is likely to promote changes in neural circuits based on excitability and plasticity cortical properties. Some patients relearn how to walk without assistance. Functional Electrical Stimulation (FES) allows for controlling artificially movements in patients with motor disability by applying electrical stimuli to impaired muscles through surface or implanted electrodes. FES has been shown to be a valuable method for training stroke patients in early phase of hemiplegia to improve recovery of walking skills. In this context, the timing of muscle stimulation sequences is critical. Today, the existing FES systems usually provide with fixed stimulation patterns, over 1 gait cyle, parameterized and pre-programmed off-line. The triggering of stimulation sequences is often achieved manually by the clinician assisting the patient. If the patient walks faster or slower than the programmed sequence, the movements on the heatlhy and the paretic leg might not be well coordinated. A critical issue is the command of this stimulation: there is a need to apply the proper stimulation amplitude at the **proper moment**. This problem can be seen as a teleoperation issue: observing the healthy leg, the goal is to apply an adapted command on the affected one. Applied to this context, our method could be used in two different ways:

- for each muscle it is possible to adapt stimulation parameters (amplitude, pulse width, frequency) according to the computed phase,
- desired joint trajectories parameterized by the phase could be generated. Using a musculo-skeletal model and solving the inverse problem, the stimulation sequences re-

quired to achieve such a movement could be computed.

One possible developement of our method is an extension the multidimensional case: how to integrate several inputs in our scheme? One idea is to build an oscillator with multiple outputs, each one them simulating one measurement, and then synthesize a single observer with multidimensional input. Another possibility is to build as many adapted oscillators as sensor measurements, and then fusing the several phase estimations.
Also, it would be of great interest to study the behaviour of our system in the case of uneven terrain (slope, stairs, ...). If the input signal is too different from the original reference, it may loose its synchronization properties. In that case, different oscillator-observers could be prepared, each one corresponding to a given walking task; we would thus build a kind of "filter bank", switching from one to another if necessary.

References

[AAEA04] C. Azevedo, B. Amblard, B. Espiau, and C. Assaiante. Functional bases of bipedal locomotion: what do humans can tell to robots ? *Technical Report 5450, INRIA*, 2004.

[AtBt00] C. Azevedo and the Bip team. Control architecture and alghorithms of the anthropomorphic biped bip2000. In *Proc. of International Symposium on Mobile Climbing and Walking Robots (CLAWAR)*, 2000.

[Bet97] John Betts. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 1997.

[BFNP97] I. I. Blekhman, A. L. Fradkov, H. Nijmeijer, and A. Yu Pogromsky. On self-synchronization and controlled synchronization. *Systems and Control Letters*, 31(5):299–306, 1997.

[BHar] S. Bonnet and R. Heliot. A magnetometer-based approach for studying human movements. *IEEE Trans. Biomedical Engineering*, to appear.

[CP91] T. L. Carroll and L. M. Pecora. Synchronizing chaotic circuits. *IEEE Trans. Circuits and Syst.*, 38:453–456, 1991.

[CRG88] A.H Cohen, S. Rossignol, and S. Grillner. *Neural control of rhyhtmic movements in vertebrates.* Wiley, 1988.

[CWR01] S.H. Collins, M. Wisse, and A. Ruina. A three-dimensional passive dynamic walking robot with two legs and knees. *The International Journal of Robotics Research*, 20(7):607–615, 2001.

[DZZZ06] H. Dong, M. Zhao, J. Zhang, and N. Zhang. Cpg-based adaptive dynamic control of a quadruped robot with sensory feedback. In *proc. Intl. Conf. on Climbing and Walking Robots (CLAWAR)*, pages 77–81, 2006.

[EMNC02] G. Endo, J. Morimoto, J. Nakanishi, and G. Cheng. An empirical exploration of a neural oscillator for biped locomotion control. In *proc. Intl. Conf. on Robotics and Automation (ICRA2004)*, 2002.

[GH90] J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields.* Springer-Verlag, 1990.

[Gri85] S. Grillner. Neurobiological bases of tythmic motor acts in vertebrates. *Science*, 228:143–149, 1985.

[GTE98] A. Goswami, B. Thuilot, and B. Espiau. A study of the passive gait of a compass-like biped robot: Symmetry and chaos. *The International Journal of Robotics Research*, 17(12):1282–1301, 1998.

[Ijs01] A.J. Ijspeert. A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. *Biological Cybernetics*, 84(5):331–348, 2001.

[Isi95] A. Isidori. *Nonlinear Control Systems.* Springer, 1995.

[Kai80] T. Kailath. *Linear Systems.* Prentice-Hall, 1980.

[LP44] A. I. Lur'e and V. N. Postnikov. On the theory of stability of control systems. *Applied mathematics and mechanics*, 8, 1944.

[Lue71] D. Luenberger. An introduction to observers. *IEEE trans. Automatic Control*, 16(6):596–602, 1971.

[LZT97] Craig Lawrence, Jian Zhou, and André Tits. User's guide for cfsqp version 2.5 : A c code for solving (large scale) constrained nonlinear (minimax) optimization problems, generating iterates satisfying all inequality constraints. Technical report, Electrical Engineering Department and Institute for Systems Research, University of Maryland, 1997.

[Mat85] K. Matsuoka. Sustained oscillations generated by mutually inhibiting neurons with adaptation. *Biological Cybernetics*, 52(6):367–376, 1985.

[McG90] T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9(2):62–82, 1990.

[NBC+04] N. Noury, P. Barralon, P. Couturier, F. Favre-Réguillon, R. Guillemaud, C. Mestais, Y. Caritu, D. David, S. Moine, A. Franco, F. Guiraud-By, M. Bérenguer, and H. Provost. A mems based microsystem for the monitoring of the activity of frail elderly in their daily life: the actidom project. In *HealthCom'2004*, 2004.

[NM97] H. Nijmeijer and I. M. Y. Mareels. An observer looks at synchronization. *IEEE Transactions on Circuits and Systems - Part I - Fundamental Theory and Applications*, 44(10):882–890, 1997.

[PPS98] S. D. Prentice, A. E. Patla, and D. A. Stacey. Simple artificial neural network models can generate basic muscle activity patterns for human locomotion at different speeds. *Experimental Brain Research*, 123(4):474–480, 1998.

[PRK01] A. Pikovsky, R. Rosenblumn, and J. Kurths. *Synchronization, a universal concept in nonlinear sciences.* Cambridge University press, 2001.

[RA06] 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.

[RBI05] L. Righetti, J. Buchli, and A.J. Ijspeert. From dynamic hebbian learning for oscillators to adaptive central pattern generators. In *proc. Adaptive Motion in Animals and Machines (AMAM2005)*, 2005.

[Tag00] G. Taga. Nonlinear dynamics of the human motor control. In *proc. Adaptive Motion in Animals and Machines (AMAM2000)*, 2000.

[Wil98] M. M. Williamson. Neural control of rhythmic arm movements. *Neural Networks*, 11(7-8):1379–1394, 1998.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399