Adaptive frequency oscillators and applications

Ludovic Righetti*, Jonas Buchli† and Auke Jan Ijspeert*

*School of Communication and Computer Science
Ecole Polytechnique Fédérale de Lausanne
Lausanne, CH-1015 Switzerland
email: ludovic.righetti@a3.epfl.ch, auke.ijspeert@epfl.ch

†Computational Learning and Motor Control Lab
University of Southern California
Los Angeles, CA 90089, USA
email: jonas@buchli.org

***Abstract*— In this contribution we present a generic mechanism to transform an oscillator into an adaptive frequency oscillator, which can then dynamically adapt its parameters to learn the frequency of any periodic driving signal. Adaptation is done in a dynamic way: it is part of the dynamical system and not an offline process. This mechanism goes beyond entrainment since it works for any initial frequencies and the learned frequency stays encoded in the system even if the driving signal disappears. Interestingly, this mechanism can easily be applied to a large class of oscillators from harmonic oscillators to relaxation types and strange attractors. Several practical applications of this mechanism are then presented, ranging from adaptive control of compliant robots to frequency analysis of signals and construction of limit cycles of arbitrary shape.**

I. Introduction

Nonlinear oscillators are very important modeling tools in biological and physical sciences, and these models have received particular attention in many engineering fields over the last few decades. The models are interesting because of their capability to synchronize with other oscillators or with external driving signals. However, these synchronization capabilities are limited, and it is not always an easy task to correctly choose the model parameters to ensure proper synchronization with the external driving signals. Indeed, an oscillator has a finite entrainment region which depends on many parameters, such as the coupling strength and the frequency difference between the oscillator and the driving signal.

Recent work, however, has shown that it is possible to modify nonlinear oscillators so that they can overcome the limitations above, by adding dynamics to the parameters of an individual oscillator, allowing it to *learn* the frequency of an input signal. These attempts are often limited to simple classes of oscillators, equivalent to phase oscillators [**?**], [**?**] or to simple classes of driving signal (pulses) [**?**].

Recently we designed a learning mechanism for oscillators, which adapts the oscillator frequency to the frequency of any periodic input signal [**?**], [**?**]. The parameter with the strongest influence on the frequency of the oscillator is turned into a new state variable for the system. Interestingly, this mechanism appears to be generic enough to be applied to many different types of oscillators, from phase oscillators to relaxation types, and to strange attractors. The frequency adaptation process goes beyond mere entrainment, because, even if the input signal disappears, the learned frequency stays encoded in the oscillator. Moreover, it is independent of the initial conditions, thus working beyond entrainment basins (i.e. it has an infinite basin of attraction). We call this adaptation mechanism *dynamic Hebbian learning* because it shares similarities with correlation-based learning observed in neural networks [**?**].

In this contribution, we present our generic adaptation mechanism. Then we demonstrate several applications, ranging from adaptive control of legged robots with passive dynamics [?], [**?**], where the adaptive oscillators find the resonant frequency of the robot, to frequency analysis with systems of coupled adaptive oscillators [**?**], and finally to construction of limit cycles with arbitrary shape [**?**].

II. Adaptive frequency oscillators

A. A generic rule for frequency adaptation

We consider general equations for an oscillator perturbed by a periodic driving signal

$$\begin{aligned} \dot{x} &= f_x(x, y, \omega) + KF(t) \\ \dot{y} &= f_y(x, y, \omega) \end{aligned}$$

where f_x and f_y are functions of the state variables that produce a structurally stable limit cycle, and of a parameter ω that has a monotonic relation with the frequency of the oscillator when unperturbed, $K = 0$ (we do not require this relation to be linear). $F(t)$ is a time periodic perturbation and $K > 0$ the coupling strength.

In order to enable the oscillator to learn the frequency of $F(t)$, we transform the ω parameter into a new state variable, with its own dynamics. The generic rule that allows us to transform the basic oscillator into an adaptive frequency oscillator is as follows

$$\dot{\omega} = \pm KF(t)\frac{y}{\sqrt{x^2 + y^2}}$$

where the sign depends on the direction of rotation of the limit cycle in the (x, y) plane.

B. Properties of the adaptation mechanism

We proved in [?] that the adaptation mechanism causes an oscillator's frequency to converge to the frequency of any periodic input signal, for phase and Hopf oscillators. In the case where there are several frequencies in the spectrum of $F(t)$, the oscillator converges to one input frequency component, depending on the initial frequency of the oscillator.

Further, the higher the coupling strength K, the faster convergence occurs. It can be shown that for suitable coupling strength, the convergence is exponential (of order e^{-t}) [**?**]. Examples of frequency adaptation for the Hopf oscillator, with several different inputs, are shown in Figure 1. The corresponding equations for the adaptive Hopf oscillator are

$$\begin{aligned} \dot{x} &= (\mu - x^2 - y^2)x - \omega y + KF(t) \\ \dot{y} &= (\mu - x^2 - y^2)y + \omega x \\ \dot{\omega} &= -KF(t)\frac{y}{\sqrt{x^2 + y^2}} \end{aligned}$$

We can note from Figure 1(d) that the adaptation mechanism works for time-varying signals (i.e. with time-varying frequencies). The tracking ability is limited, however, by the exponential convergence rate of the adaptation mechanism. Further examples of such tracking and limitations can be found in [?], [?] for pools of oscillators.

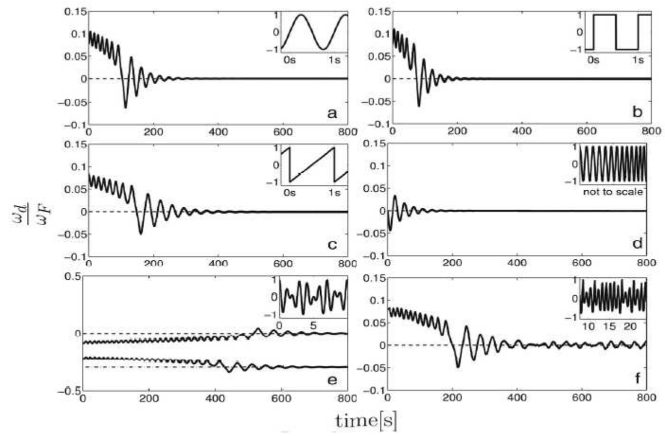
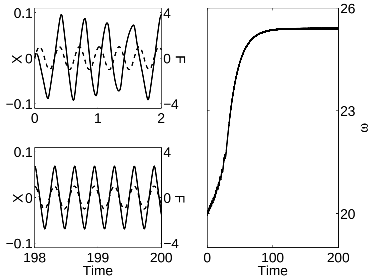


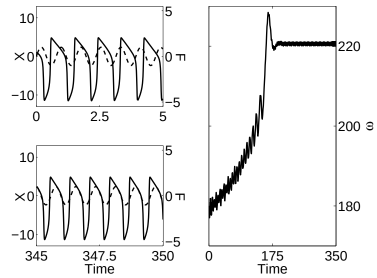
Fig. 1. (a) Typical convergence of an adaptive frequency Hopf oscillator driven by a harmonic signal ($(F(t) = \sin(2\pi t))$. The frequencies converge towards the frequency of the input (indicated in dashed line). After convergence the frequency oscillates with a small amplitude around the frequency of the input. In all figures, we plot in the main graph the time evolution of the difference between ω and the input frequency, normalized by the input frequency. The top right panel shows the driving signals (note the different scales). (b) Square pulse $F(t) = \mathrm{rect}(\omega_F t)$, (c) Sawtooth, $F(t) = \mathrm{st}(\omega_F t)$, (d) Chirp $F(t) = \cos(\omega_c t)$, where $\omega_c = \omega_F(1 + \frac{1}{2}(\frac{t}{1000})^2)$. (Note that the graph of the input signal is illustrative only since changes in frequency takes much longer than illustrated). (e) Signal with two non-commensurate frequencies $F(t) = \frac{1}{2}\left[\cos(\omega_F t) + \cos(\frac{\sqrt{2}}{2}\omega_F t)\right]$, i.e. a representative example how the system can evolve to different frequency components of the driving signal depending on the initial condition $\omega_d(0) = \omega(0) - \omega_F$. (f) $F(t)$ is the non-periodic output of the Rössler system. The Rössler signal has a $1/f$ broad-band spectrum, yet it has a clear maximum in the frequency spectrum. In order to assess the convergence we use $\omega_F = 2\pi f_{\max}$, where $f_{\max}$ is found numerically by FFT. The oscillator converges to this frequency.

Our extensive numerical simulations also show that this adaptation mechanism works for many different types of non-harmonic oscillators. Some examples, shown in Figure 2, are an adaptive Rayleigh oscillator, an adaptive Fitzhugh-Nagumo oscillator and a Rössler system in chaotic mode. For the first two cases there is no linear relation between ω and the frequency of oscillations, but the adaptive mechanism is able to find a suitable value for ω such that the frequency of the oscillator is the same as the frequency of the input signal. For the Rössler system, the frequency of the system is not well defined since the system is not periodic, but we can define a pseudo-frequency and the system can then adapt it to the frequency of a periodic input.



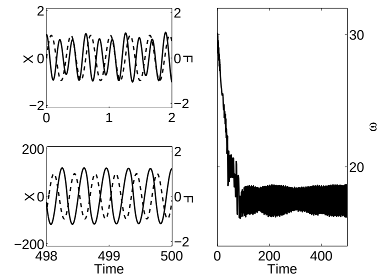
$$\dot{x} = y + KF$$
$$\dot{y} = \delta(1 - qy^2)y - \omega^2 x$$
$$\dot{\omega} = KF\frac{y}{\sqrt{x^2+y^2}}$$

(a) Adaptive Rayleigh oscillator



$$\dot{x} = x(x-a)(1-x) - y + KF$$
$$\dot{y} = \omega(x - by)$$
$$\dot{\omega} = -KF\frac{y}{\sqrt{x^2+y^2}}$$

(b) Adaptive Fitzhugh-Nagumo oscillator



$$\dot{x} = -\omega y - z + KF$$
$$\dot{y} = \omega x + ay$$
$$\dot{z} = b - cz + xz$$
$$\dot{\omega} = -KF\frac{y}{\sqrt{x^2+y^2}}$$

(c) Adaptive Rössler system

Fig. 2. For each oscillator, ω corresponds to the adaptive parameter. Each figure is composed of 3 plots. The right plot shows the evolution of ω. The left plots are the time evolution of the oscillators (the x variable) and of the input signal F (dashed line), before (upper plot) and after (lower plot) adaptation.

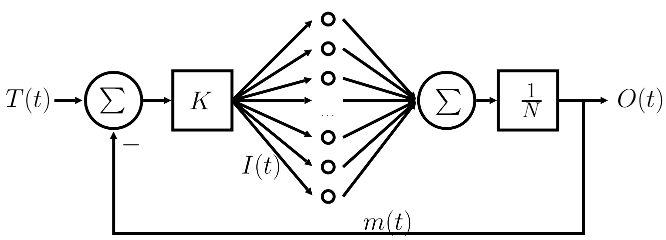


Fig. 3. Structure of the pool of adaptive frequency oscillators that is able to reproduce a given teaching signal $T(t)$. The mean field produced by the oscillators is fed back negatively to the oscillators.

III. Applications

We now present several applications for the adaptation mechanism, ranging from robot control to frequency analysis and automatic construction of limit cycles of arbitrary shape.

A. Robot with passive dynamics

The adaptation mechanism can be used to find the resonant frequencies of legged robots with passive elements (i.e. springs) [?], [**?**], [?], [**?**]. A controller based on adaptive frequency oscillators is able to tune itself to the resonant frequency of the robot, via a simple feedback loop using sensors on-board (e.g. position or inertial sensors). Locomotion can therefore be made very efficient by exploiting the intrinsic dynamics of the robot. Another advantage is that one does not need to tune the controller for a specific robot; the controller can also track any changes in resonant frequency automatically, if, for example, the frequency changes due to a variation in mass or spring stiffness, or because of a gait transition (e.g. the resonant frequency is different if the robot is standing on two feet or four feet).

B. Frequency analysis

Another application is the use of a pool of adaptive frequency Hopf oscillators to perform frequency analysis on an input signal [?]. The oscillators are coupled via a negative mean field with the input teaching signal, as is shown in Figure 3. The oscillators converge to the frequencies present in the spectrum of the teaching signal and due to the negative feedback, each time an oscillator finds a correct frequency, this one loses its amplitude. Thus, the other oscillators only *feel* the remaining frequencies to learn.

The pool of oscillators is able to approximate the frequency spectrum of any input signal. This works for signals with discrete spectra, and also for those with continuous and time-varying spectra. The spectrum is approximated by the distribution of the frequencies of the oscillators, and so the resolution of the approximation can be made arbitrary good by increasing the number of oscillators in the pool.

Figure 4 shows how the system can approximate the spectrum of a broad-band chaotic signal from the Rössler system. As can be seen, the important features of the spectrum are caught by the system, especially the broad spectrum and the major frequency peaks.

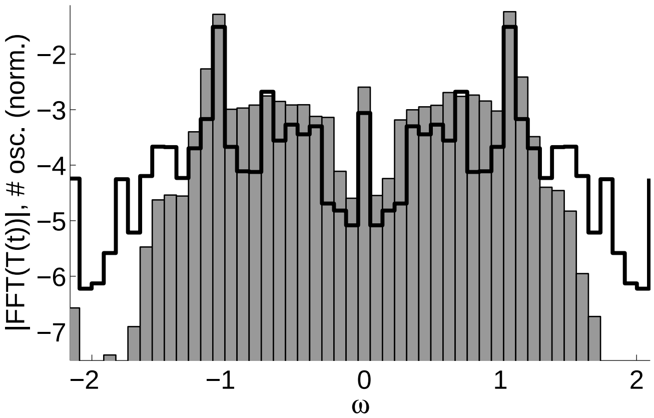


Fig. 4. FFT of the Rössler signal (black line) in comparison with the distribution of the frequencies of the oscillators (gray bars normalized to the number of oscillators, $N = 10000$). The spectrum of the signal has been discretized into the same bins as the statistics of the oscillators in order to allow for comparison with the results from the full-scale simulation.

C. Construction of limit cycles with arbitrary shape

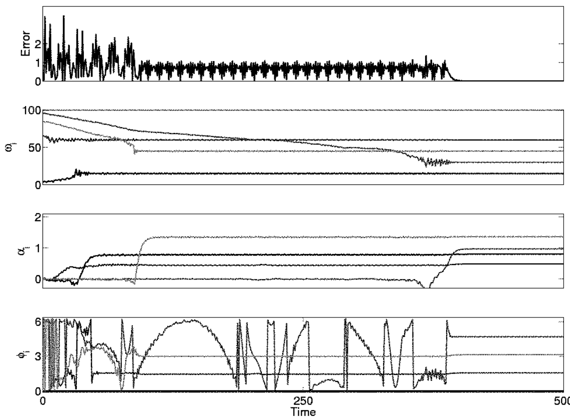
The previous pool of oscillators can be extended by adding a weight to each oscillator in the mean field sum, and a coupling between oscillators, in order to ensure stability of the output pattern. The result is that an individual oscillator will be able to fully match the energy content of a frequency in the spectrum of the teaching signal. Moreover, the coupling ensures that the system exhibits a stable limit cycle. Here, amplitudes and phase differences become system state variables, in addition to frequencies. The governing differential equations of the system are then

$$
\begin{aligned}
\dot{x}_i &= (\mu - r_i^2)x_i - \omega_i y_i + KF(t) \\
&\quad + \tau \sin(\frac{\omega_i}{\omega_0}\theta_0 - \theta_i - \phi_i) \\
\dot{y}_i &= (\mu - r_i^2)y_i + \omega_i x_i \\
\dot{\omega}_i &= -KF(t)\frac{y_i}{r_i} \\
\dot{\alpha}_i &= \eta x_i F(t) \\
\dot{\phi}_i &= \sin\left(\frac{\omega_i}{\omega_0}\theta_0 - \theta_i - \phi_i\right)
\end{aligned}
$$

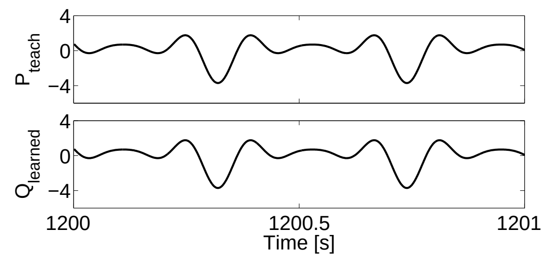
with

$$
\begin{aligned}
\theta_i &= \operatorname{sgn}(x_i)\cos^{-1}\left(-\frac{y_i}{r_i}\right) \\
F(t) &= P_{\text{teach}}(t) - Q_{\text{learned}}(t) \\
Q_{\text{learned}}(t) &= \sum_{i=0}^{N} \alpha_i x_i
\end{aligned}
$$

where τ, K and η are positive constants. The output of the system, Q_{learned}, is the weighted sum of the output of each oscillator. $F(t)$ represents the negative feedback, which on average is the remainder of the teaching signal $P_{\text{teach}}(t)$ that the network still has to learn. α_i represents the amplitude associated with the frequency ω_i of oscillator i. The evolution equation maximizes the correlation between x_i and $F(t)$, which means that α_i will increase only if ω_i has converged to a frequency component of $F(t)$ (the correlation will be positive on average) and will stop increasing when the frequency component ω_i disappears from $F(t)$ because of the negative feedback loop. ϕ_i is the phase difference between oscillator i and 0. The value converges to the phase difference between the instantaneous phase of oscillator 0, θ_0, scaled for frequency ω_i, and the instantaneous phase of oscillator i, θ_i. Each adaptive



(a) Evolution of the state variables of the system



(b) Result of learning

Fig. 5. Construction of a limit cycle by learning an input signal ($P_{\text{teach}} = 0.8\sin(15t) + \cos(30t) - 1.4\sin(45t) - 0.5\cos(60t)$). Figure 5(a) shows the evolution of the state variables of the system during learning. The upper graph is a plot of the error ($\|P_{teach} - Q_{learned}\|$). The 3 other graphs show the evolution of the frequencies, ω_i, the amplitudes, α_i and the phases, ϕ_i. We clearly see that the system can learn the teaching signal perfectly – the frequencies, amplitudes and phase differences converge to the correct values and the error becomes zero. Figure 5(b) shows the result of learning (teaching signal in upper graph, output of the system in lower graph), we note the perfect reconstruction of the signal.

oscillator is coupled with oscillator 0, with strength τ, to maintain the correct phase relationships between oscillators.

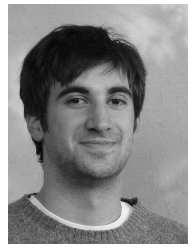
Figure 5 shows an example of the convergence of a network of oscillators with amplitudes and coupling, together with the resulting learned signal. We see that the individual oscillator frequencies first converge to the frequency components present in the teaching signal. Individual amplitudes increase when the associated frequency matches one frequency of the input signal. Finally, the phase differences stabilize and we see that the error is zero, which means that the system has perfectly reconstructed the teaching signal. Further, the teaching signal is now encoded into a structurally stable limit cycle and it is easy to smoothly modulate its frequency and amplitude by changing $\vec{\omega}$ and $\vec{\alpha}$. These properties can be very useful, together with sensory feedback, for robotics control (see for example [?]). This system can be viewed as a dynamic Fourier series decomposition where there is no need of explicitly define a time window or to perform any preprocessing of the input signal.

IV. Conclusion

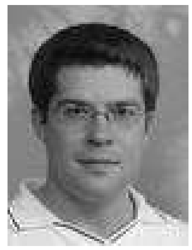
In this contribution we presented a generic mechanism for building adaptive frequency oscillators from a given, existing oscillator. We showed that our approach can be applied to many different types of oscillators, and that the resulting systems are able to learn the frequencies of any periodic input signal. Interestingly, there is no need to preprocess the signal and no external optimization procedures are required to obtain the correct frequency. All the learning is embedded in the dynamics of the adaptive oscillators. Moreover, our results go further than entrainment, since the learned frequency is maintained in the system even if the external driving oscillation disappears and the basin of attraction is infinite (i.e. the system can start from any initial frequency). Finally, we discussed some applications of this mechanism, ranging from adaptive control for compliant robots, to frequency analysis and construction of limit cycles of arbitrary shape.

Acknowledgment

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, J.B. and A.I.)



Ludovic Righetti is working towards a PhD at the Biologically Inspired Robotics Group (BIRG) at the Ecole Polytechnique Fédérale de Lausanne (EPFL). He has a BS/MS in Computer Science from EPFL (March 2004). He is particularly interested in the theoretical aspects of locomotion, both in animals and legged robots, and more generally in control theory and dynamical systems.



Jonas Buchli is currently a Post-doctoral Fellow at the Computational Learning and Motor Control Lab at the University of Southern California in Los Angeles. He received a Diploma in Electrical Engineering from the ETHZ (the Swiss Federal Institute of Technology, Zürich) in 2003 and a Doctorate from the EPFL (the Swiss Federal Institute of Technology, Lausanne) in 2007. His research interests include self-organization and emergent phenomena in complex systems, the theory of nonlinear dynamical systems and information concepts. And, especially the applications of these topics to the intersecting field of engineering and biology.

Auke Ijspeert is a SNF (Swiss National Science Foundation) assistant professor at the EPFL (the Swiss Federal Institute of Technology at Lausanne), and head of the Biologically Inspired Robotics Group (BIRG). He has a BSc/MSc in Physics from the EPFL, and a PhD in artificial intelligence from the University of Edinburgh (with John Hallam and David Willshaw as advisors). He carried out postdocs at IDSIA and EPFL (LAMI) with Jean-Daniel Nicoud and Luca Gambardella, and at the University of Southern California (USC), with Michael Arbib and Stefan Schaal. Before returning to the EPFL, he was a research assistant professor at USC, and an external collaborator at ATR (Advanced Telecommunications Research institute) in Japan. He is still affiliated as adjunct faculty to both institutes. His research interests are at the intersection between robotics, computational neuroscience, nonlinear dynamical systems, and adaptive algorithms (optimization and learning algorithms). He is interested in using numerical simulations and robots to get a better understanding of the functioning of animals (in particular their fascinating sensorimotor coordination abilities), and in using inspiration from biology to design novel types of robots and adaptive controllers. He is regularly invited to give talks on these topics. With his colleagues, he has received the Best Paper Award at ICRA2002, the Industrial Robot Highly Commended Award at CLAWAR2005, and the Best Paper Award at the IEEE-RAS Humanoids 2007 conference. He is/was the Technical Program Chair of 5 international conferences (BioADIT2004, SAB2004, AMAM2005, BioADIT2006, LATSIS2006), and has been a program committee member of over 35 conferences.