

# Control of Legged Locomotion using Dynamical Systems: Design Methods and Adaptive Frequency Oscillators

THÈSE N° 4222 (2008)

PRÉSENTÉE LE 7 NOVEMBRE 2008

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS

GROUPE IJSPEERT

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

**Ludovic RIGHETTI**

ingénieur informaticien diplômé EPF  
et de nationalité française

acceptée sur proposition du jury:

Prof. A. Martinoli, président du jury

Prof. A. Ijspeert, directeur de thèse

Prof. M. Hasler, rapporteur

Prof. O. Khatib, rapporteur

Prof. S. Schaal, rapporteur



ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE

Suisse  
2008



---

# ABSTRACT

---

Legged robots have gained an increased attention these past decades since they offer a promising technology for many applications in unstructured environments where the use of wheeled robots is clearly limited. Such applications include exploration and rescue tasks where human intervention is difficult (e.g. after a natural disaster) or impossible (e.g. on radioactive sites) and the emerging domain of assistive robotics where robots should be able to meaningfully and efficiently interact with humans in their environment (e.g. climbing stairs). Moreover the technology developed for walking machines can help designing new rehabilitation devices for disabled persons such as active prostheses. However the control of agile legged locomotion is a challenging problem that is not yet solved in a satisfactory manner.

By taking inspiration from the neural control of locomotion in animals, we develop in this thesis controllers for legged locomotion. These controllers are based on the concept of Central Pattern Generators (CPGs), which are neural networks located in the spine of vertebrates that generate the rhythmic patterns that control locomotion. The use of a strong mathematical framework, namely dynamical systems theory, allows one to build general design methodologies for such controllers.

The original contributions of this thesis are organized along three main axes. The first one is a work on biological locomotion and more specifically on crawling human infants. Comparisons of the detailed kinematics and gait pattern of crawling infants with those of other quadruped mammals show many similarities. This is quite surprising since infant morphology is not well suited for quadruped locomotion. In a second part, we use some of these findings as an inspiration for the design of our locomotion controllers. We try to provide a systematic design methodology for CPGs. Specifically we design an oscillator to independently control the swing and stance durations during locomotion, then using insights from dynamical systems theory we construct generic networks supporting different gaits and finally we integrate sensory feedback in the system. Experiments on three different simulated quadruped robots show the effectiveness of the approach. The third axis of research focus on dynamical systems theory and more specifically on the development of an adaptive mechanism for oscillators such that they can learn the frequency of any periodic signal. Interestingly this mechanism is generic enough to work with a large class of oscillators. Extensive mathematical analysis are provided in order to understand the fundamental properties of this mechanism. Then an extension to pools of adaptive

---

frequency oscillators with a negative feedback loop is used to build programmable CPGs (i.e. CPGs that can encode any periodic pattern as a structurally stable limit cycle). We use the system to control the locomotion of a humanoid robot. We also show applications of this system to signal processing.

**Keywords:** Locomotion, legged robots, central pattern generators, adaptive frequency oscillators, dynamical systems.

---

# RÉSUMÉ

---

Ces dernières décennies, les robots marcheurs sont devenus le centre d'une attention particulière. En effet, ils offrent une technologie prometteuse pour une large gamme d'applications dans des environnements non structurés où l'utilisation de robots avec des roues devient clairement limitée. De telles applications incluent les tâches d'exploration et de sauvetage où les interventions humaines sont difficiles (ex: après une catastrophe naturelle) ou impossible (ex: sur un site radioactif) ainsi que le domaine en pleine expansion de la robotique d'assistance où les robots doivent être capables d'interagir de manière sensée et efficace avec des humains dans leur environnement (ex: monter des escaliers). De plus, la technologie développée pour les robots marcheurs peut aider à la conception de nouveaux appareils de réhabilitation pour des personnes handicapées comme par exemple des prothèses actives. Malheureusement le contrôle de robots marcheurs agiles est toujours un problème difficile qui n'a pas encore été résolu de manière satisfaisante.

En prenant inspiration du contrôle neuronal de la locomotion chez les animaux, nous développons dans cette thèse des contrôleurs pour la locomotion à pattes. Ces contrôleurs sont basés sur le concept de Générateurs de Patrons Centraux (GPCs), c'est à dire des réseaux de neurones localisés dans l'épine dorsale des vertébrés qui génèrent les motifs rythmiques contrôlant la locomotion. L'utilisation d'un cadre mathématique rigoureux, celui des systèmes dynamiques, nous permet ainsi de construire des méthodes de conception pour ces contrôleurs.

Les contributions originales de cette thèse sont organisées selon trois axes principaux. Le premier est un travail sur la locomotion animale et plus spécifiquement sur la marche à quatre pattes des enfants. Des comparaisons de la cinématique détaillée et de la démarche des enfants avec celles des autres mammifères quadrupèdes montrent beaucoup de similarités. Ceci est assez surprenant puisque la morphologie des enfants ne convient pas à la marche quadrupède. Dans une seconde partie, nous utilisons ces observations comme une source d'inspiration pour la conception de contrôleurs pour la locomotion. Nous essayons de développer une méthode systématique pour la construction de GPCs. Plus spécifiquement, nous concevons un oscillateur dédié à la locomotion pour contrôler de manière indépendante la durée des phases de levé et de posé pendant la locomotion, ensuite en utilisant des développements récents de la théorie des systèmes dynamiques nous construisons des réseaux génériques qui produisent différentes démarches. Finalement nous intégrons de l'information sensorielle dans le système. Des expériences sur

---

trois différents robots en simulation montrent l'efficacité de l'approche. Le troisième axe de recherche se concentre sur la théorie des systèmes dynamiques et plus particulièrement sur le développement d'un mécanisme adaptatif pour les oscillateurs afin qu'ils puissent apprendre la fréquence de n'importe quel signal périodique. Ce mécanisme s'avère être suffisamment générique pour fonctionner avec une grande classe d'oscillateurs. Nous apportons une analyse mathématique extensive dans le but de caractériser les propriétés essentielles du mécanisme d'adaptation. Ensuite nous proposons une extension avec un ensemble d'oscillateurs adaptatifs et une boucle de rétroaction négative qui est utilisée pour construire des GPCs programmables (c'est à dire des GPCs qui peuvent encoder n'importe quel motif périodique dans un cycle limite stable). Nous montrons aussi des applications de ce système pour contrôler la locomotion d'un robot humanoïde et dans le cadre du traitement de signal.

**Mots clefs:** Locomotion, robots à pattes, generateurs de patrons centraux, oscillateurs avec adaptation de fréquence, systèmes dynamiques.

---

# ACKNOWLEDGMENTS

---

This is the place where I have the opportunity to acknowledge all the great people that I met and worked with and who made this thesis possible.

First of all I would like to infinitely thank Professor Auke Jan Ijspeert who made this thesis possible by hiring me as a PhD student in his group. I am really grateful to him because he trusted me and let me investigate my own ideas, while always being very supportive and always having good advice when I got stuck. I am really impressed on how he can find in ten minutes a solution to a problem I have been working on for weeks. Moreover he gave me the opportunity to go to many conferences and workshops, meet a lot of interesting people, to take an active part in the RobotCUB project and he always took the time to present my work during his presentations, all of which I know is not always the rule in research. Furthermore he is a great person to know and to work with, always friendly and available even when he is overloaded by huge amounts of work that would drive anyone else crazy.

Here I want to immensely thank Dr. Jonas Buchli because the main piece of work of this thesis would not exist without the great collaboration I had with him. He first was my supervisor for the master thesis, then he became a colleague of mine and most important a very good friend. I thank him for making me discover the cool world of dynamical systems and its ramifications. With him I discovered that a good discussion and many good beers were sometimes more efficient than a week of work (and always a lot more fun).

Now I would like to thank the members of my thesis committee: Martin Hasler, Ousama Khatib and Stefan Schaal as well as the president Alcherio Martinoli for accepting to be part of the committee, for the time they spent for it and for giving me very good feedback.

Then I would like to thank Sarah Dégallier for all the good times we had working on the RobotCUB project and specially on the iCub, for the endless philosophical discussions until we got kicked out of the hotel bar and for the good mathematical advice that she gave me.

Now I want to thank all the people in the RobotCUB consortium. I learned a lot working in this great project and the meeting were always very interesting and bringing many new ideas. A special thanks goes to the Babylab at Uppsala University and to Kerstin Rosander and Anna Nylèn for the wonderful experiments with crawling infants

---

that they made for us. I thank also the IIT team, specially Giulio Sandini and Giorgio Metta for being so welcoming when we came working with the iCub. Here a special thanks goes to Lorenzo Natale and Francesco Nori because they were always very friendly and patient and without them we would have never made the iCub drum and crawl.

I would like to thank an essential person in the lab: Marlyse Taric. Thanks for always being friendly, helpful and for finding convenient solutions to all these administrative issues.

The second essential person in the lab is Alessandro Crespi. I thank him for his precious help with all that computer stuff, without him things would have been much more complicated. I will miss his good mood and crazy hoaxes.

I would like to thank the people at Cyberbotics: Olivier Michel and Yvan Bourquin, thanks for developing such a great simulator and for always finding good solutions to our simulation needs.

Then I would like to thank my first office mates, Yann Thoma for all the cool music he made me discover and Fabien “no limit” Vannel for all these unlikely parties without end. Moreover I thank them for their warm welcome when I joined their office.

I want also to thank Pierre-André Mudry who was my last office mate. I will miss the “bureau du bonheur”, full of cool music and great spirit.

I want to thank Joël Rossier for the great parties we had and for the cool books he made me discover.

Another thanks goes to Andres Upegui for these great times we had sharing pitchers in Satellite.

To Masoud Asadpour for his sense of humor and for sharing with us a different culture.

To Alexander Sproewitz for the nice discussions, the great sushis, the train accident and the billion of pictures.

To Jérémie Knuesel for understanding and sharing my passion for cheese.

To Kostas Karakasiliotis, the last student who joined the lab for his good mood and motivation.

To André “Chico” Badertscher for all his great jokes which were enlightening my day. I hope someday he will write a book with all of them. Thanks also for the great book you recommended me.

To Andrej Gams, who came to visit us for a few months and stayed almost one year, thanks for the hawaiians and the Laibach concert.

To all the great people that came to work in the lab for a few months, that was great discussing and sharing ideas with you.

I would like to thank the members of the former Logical Systems Lab which was headed by Daniel Mange: André Stauffer, Eduardo Sanchez and Gianluca Tempesti. They were always very supportive and friendly to the BIRG newcomers we were.

To the students I supervised: Vlad, Julien, Martin, Andrina, Fabrizio, Cédric, Neha, Christian, Simon, Loïc, Aïsha, Martin and Ahmed. It was a pleasure working with you and I surely benefited from these collaborations.



---

To the LASA crew, for all the great ski outings and fruitful discussions we had. A special thanks goes to Prof. Aude Billard for the RobotCUB project and to Micha Hersch, who started the RobotCUB project at the same time as me, for the exciting discussions we had during these meetings.

To Max-Olivier Hongler and Julio Rodriguez for their enthusiasm for adaptive frequency oscillators and for the fruitful discussions we had during our meetings.

Pursuing my passion for research would not have been possible without a good balance in my life. For that I wholeheartedly thank Jeanne, Jean, Adriano, Solaiman, Lisa, Michel and all the others for all the great times we have that make a tremendous difference and certainly helped me a lot these last four years. A special thanks goes to Jean and Adriano, for the great vacations and because things are simple with you guys.

And last but certainly not least, I would like to express all my gratitude to my parents Joséphine and Jean-Pierre as well as to my sister Sabrina for their unfailing support and encouragements these last 27 years. This would not have been possible without you.

★

This thesis was possible thanks to the support of the European Commission Cognition Unit, project no. IST-2004-004370: RobotCUB and to a grant from the Swiss National Science Foundation (Young Professorship Award to Auke Ijspeert). I would therefore like to thank these two funding agencies for the great research opportunities they create.

*Ludovic Righetti*  
Lausanne, November 2008

---

---

# CONTENTS

---

<b>Abstract</b>	<b>i</b>
<b>Résumé</b>	<b>ii</b>
<b>Acknowledgments</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Context and related work</b>	<b>7</b>
2.1 The RobotCUB Project . . . . .	7
2.2 Animal locomotion . . . . .	9
2.3 Central Pattern Generators . . . . .	10
2.4 Control theory and optimization . . . . .	12
2.5 Control of legged locomotion . . . . .	13
2.6 Dynamical systems . . . . .	14
<b>3 Crawling human infants</b>	<b>17</b>
3.1 Introduction . . . . .	18
3.2 Materials and methods . . . . .	19
3.2.1 Subjects . . . . .	19
3.2.2 Procedure . . . . .	19
3.2.3 Measurements . . . . .	20
3.2.4 Data evaluation . . . . .	20
3.2.5 Swing - Stance measurement . . . . .	22
3.2.6 Statistical measures . . . . .	22
3.3 Results . . . . .	22
3.3.1 Gait analysis . . . . .	23
3.3.2 Kinematics . . . . .	25
3.3.3 Improvement of locomotion: effects of experience . . . . .	32
3.4 Discussion . . . . .	34
3.4.1 Main gait parameters related to speed . . . . .	34
3.4.2 Kinematics . . . . .	35

3.4.3	Lateral sequence footfalls . . . . .	36
3.4.4	Spine kinematics . . . . .	37
3.4.5	Development of the gait through experience . . . . .	37
3.4.6	Conclusion . . . . .	38
3.5	Outlook . . . . .	38
<b>4</b>	<b>Design of Central Pattern Generators</b>	<b>39</b>
4.1	Open-loop CPG design . . . . .	41
4.1.1	Motivations . . . . .	41
4.1.2	Crawling in Infants from the robotics perspective . . . . .	42
4.1.3	CPG model . . . . .	44
4.1.4	Validation of the model . . . . .	51
4.1.5	Conclusion . . . . .	54
4.2	Inclusion of sensory feedback . . . . .	55
4.2.1	Motivation . . . . .	56
4.2.2	CPG design . . . . .	58
4.2.3	Experimental setup . . . . .	69
4.2.4	Experiments and results . . . . .	73
4.3	Conclusion . . . . .	81
<b>5</b>	<b>Adaptive Frequency Oscillators</b>	<b>85</b>
5.1	Frequency adaptation mechanism . . . . .	86
5.1.1	Motivation . . . . .	86
5.1.2	Learning frequencies with a Hopf oscillator . . . . .	88
5.1.3	Numerical simulations . . . . .	96
5.1.4	Generalization to non-harmonic oscillators . . . . .	102
5.1.5	Discussion . . . . .	110
5.2	Properties of adaptive phase oscillators . . . . .	111
5.2.1	Motivation . . . . .	111
5.2.2	Strongly coupled adaptive frequency phase oscillator . . . . .	112
5.2.3	Error of convergence for the adaptive Hopf oscillator . . . . .	121
5.2.4	Conclusion . . . . .	125
5.3	Pool of adaptive frequency oscillators . . . . .	125
5.3.1	Motivation . . . . .	125
5.3.2	Frequency analysis with a pool of adaptive frequency oscillators . . . . .	126
5.3.3	Dynamic adaptation to the energy content of the frequency spectrum . . . . .	131
5.3.4	Conclusion . . . . .	136
5.4	Programmable central pattern generators . . . . .	138
5.4.1	Motivation . . . . .	138
5.4.2	Generic Central Pattern Generators . . . . .	139
5.4.3	Application to biped locomotion . . . . .	142

## Contents

---

5.4.4	Experimental results . . . . .	150
5.4.5	Conclusion . . . . .	151
5.5	Comments on programmable CPGs . . . . .	153
<b>6</b>	<b>Conclusion</b>	<b>155</b>
6.1	Original contributions . . . . .	155
6.2	Final conclusion . . . . .	157
	<b>List of Figures</b>	<b>159</b>
	<b>List of Tables</b>	<b>168</b>
	<b>Bibliography</b>	<b>169</b>



*“Mais si l’on a peur de la science, c’est surtout parce qu’elle ne peut nous donner le bonheur. [...] Aussi l’homme ne peut être heureux par la science, mais aujourd’hui il peut bien moins encore être heureux sans elle.”*

Henri Poincaré, **La valeur de la science**, 1905.





---

# CHAPTER 1

## INTRODUCTION

---

This manuscript presents the research conducted during my four years of doctoral studies at the Biologically Inspired Robotics Group (BIRG), at the Ecole Polytechnique Fédérale de Lausanne in Switzerland. The main topic of this research was the development of controllers based on nonlinear oscillators for the locomotion of legged robots. These controllers are inspired by the way the nervous system of animals controls their locomotion and are grounded in a strong mathematical formalism.

**Why is the control of locomotion interesting?** Although moving robots are funny, the real interest of studying the locomotion of legged robots is that it is still an unresolved and challenging problem. From a more practical point of view legged robots, as opposed to wheeled ones, have intermittent contacts with the ground and therefore are particularly well suited for locomotion on non-smooth terrains (e.g. climbing up stairs or stepping on stones across a river). They offer then a promising technology for many applications in unstructured environments where the use of wheeled robots is clearly limited. Such applications include exploration and rescue tasks where human intervention is difficult (e.g. after a natural disaster) or impossible (e.g. on radioactive sites) and the emerging field of assistive robotics where the robots should be able to meaningfully and efficiently interact with humans in their environment (e.g climbing stairs). Moreover understanding the fundamental principles of legged locomotion and the technology developed for walking machines can help designing new rehabilitation devices for disabled persons.

The issues related to legged locomotion cover various aspects of mechanical and control engineering. What is a good mechanical design for a legged robot? How do we control the redundancy of the degrees of freedom? How do we coordinate them? What about the posture of the robot? What type of actuators should we use? What if the robot is underactuated? How does it keep balance? How do we get by without a precise model of the environment? These are the typical questions that we can ask when working with legged robots. They also impose several constraints on the design space and as such they can serve as an interesting testbed for the development of advanced engineering

techniques. For example all the computational power and energy must be embedded in the robot and its size is limited by the application it is designed for (e.g. operation in a human environment or micro-robots for medical applications). Therefore the type of actuators and the control algorithms have severe limitations. Moreover locomotion is intrinsically discontinuous because of intermittent contacts of the legs with the ground and has serious real-time constraints (we cannot stop the action of external forces and a falling robot needs to react extremely fast). This and the fact that perfect modeling of the environment is impossible limits the class of controllers that can be used. These past decades, this field has made a lot of progress but it suffices to compare current robots to any animal to see that the problem is far from being solved in a satisfactory manner and requires the development of new techniques and methodologies.

**Why being inspired by biology?** Well, at first sight the answer is straightforward: because animals outperform current robots. Taking inspiration from Nature is not new but these past decades the substantial progress in life sciences and more particularly in neurobiology have established solid foundations on which we can formulate design principles for engineering.

However one has to take care when talking about biological inspiration and to carefully define what is the goal. Our purpose in this thesis is to look at biology to grasp ideas to solve engineering problems and we do not make models to explain biology. Although these two aspects (modeling and engineering) are equally interesting research areas and can be interconnected in some ways, they are very different in their nature and in their goals. On one hand if one wants to model biology, then one's models should make predictions and be confronted systematically to real data. From this aspect what is important is not to know whether these models can be efficient engineering solutions but if they reflect natural processes. On the other hand biological inspiration is interesting only if it brings solutions that are better than the ones coming from traditional approaches. Of course these solutions do not need to outperform other approaches from the very beginning but this should be the long term goal. Then the designer is also completely free in the manipulation of the biological concept as long as it benefits the engineering field since biological plausibility is not important here.

I believe this clear distinction between modeling and biological inspired engineering is important because in robotics we often see controller designs that are complex to stay close to biology while simpler (and possibly more efficient) designs would be possible. For example in the design of locomotion controllers an extensive use of "neural" oscillators, which are quite complex, is made when there exist simpler oscillator models. Furthermore we rarely find convincing arguments that justify the use of such complex models from the engineering perspective (see for example our discussion on the topic in [18]).

In this thesis we tried to get insights from biology to design locomotion controllers that can eventually deal with complex terrains, a problem that is currently not solved

---

with traditional approaches. We do not claim that we outperform traditional methods (at least not yet!) but we keep in mind that this is the long term goal to assess the pertinence of our approach. During this research we always tried to develop design methodologies out of these insights from biology and to ground our work in a strong mathematical framework.

**Mathematical framework** In this thesis we use the mathematical framework of dynamical systems and more particularly we make an extensive use of oscillators (i.e. systems possessing a stable limit cycle). One reason to use this framework comes from the fact that Central Pattern Generators (CPGs), which are neural networks responsible for the control of locomotion in animals, are often modeled in theoretical biology as coupled oscillators. Then many results can be directly transferred in our controllers. Moreover oscillators offer a natural way of describing coordination between degrees of freedom of a robot thanks to their synchronization properties.

Another reason is that I believe that when thinking about locomotion, we should not separate the controller from the object it controls because we would like to have a strong coupling between these two entities. The natural way to describe mechanical systems is with the Lagrangian, or Hamiltonian, formalisms, i.e. with differential equations. Then a natural way to describe controllers that generate control policies for these systems should also be with differential equations. We must state this is nothing new, since control theory uses dynamical systems (or at least differential equations) in an extensive way. However since the advent of the digital world, our way of thinking is mainly algorithmic which is a very different way of thinking than to think from a dynamical systems perspective. For example learning or optimization methods always separate the algorithm from the system to optimize while in the brain the learning algorithm is part of the dynamics of the learning substrate. I do not claim that one view is better than the other and there are certainly many equivalences between these views, but the dynamical systems perspective can lead us to different (and possibly fruitful) views on engineering problems. In this thesis we show an example, the adaptive frequency oscillator, that illustrates these ideas (Chapter 5). We developed a mechanism for oscillators such that they can learn the frequency of any input signals. Interestingly the learning process is embedded in the dynamics of the oscillator and there is no explicit separation between the learning “algorithm” and the learning substrate. One advantage of this approach is that the final system is very simple from the computational point of view (a simple set of differential equations) as compared to the kind of algorithms from signal processing that we could use to have the same result (e.g. Fourier transforms).

In addition, dynamical systems theory is an interesting framework because it gives us the possibility to describe complex behaviors such as synchronization, chaos or bifurcations in a rather synthetic way. We must note, however, that although dynamical systems are a well studied branch of mathematics and that many physical systems where

extensively described using these tools, there are yet no clear methodologies to exploits these concepts from an engineering perspective.

**Contributions** In this work we developed three main axes of research. First we explored biological locomotion through the study of crawling human infants as compared to other quadruped mammals. This study, which was the first quantitative study on the subject, was originally driven by curiosity in the context of the RobotCUB project (see Chapter 2.1), which is the European project that funded this research. Although it was far from a pure robotics project, this gave us significant insights in mammalian locomotion that were useful from the robotics perspective, for example to design the gait pattern for the crawling humanoid robot, the iCub, or on our choice of controlling independently stance and swing durations in quadruped robots.

The second axis of research that derived from this study was the development of a methodology to design networks of coupled oscillators that can generate appropriate gaits for legged robots. These networks are composed of locomotion specific oscillators in which we can independently control the swing and stance durations. The most important aspect was the inclusion of sensory feedback which strongly couples the controller with the mechanical system it controls. We showed the robustness of the locomotion to parameter choices in unknown environments with three different simulated quadruped robots.

The final axis of research went a bit further in dynamical systems theory. We focused on the development of an adaptation mechanism for oscillators such that they can learn the frequency of any input signal. This research led us to several fruitful directions including adaptive control of robots with passive dynamics (work of Jonas Buchli [16, 15]), construction of limit cycles of arbitrary shape applied to the design of CPGs, and a novel way of doing signal processing (namely extracting a frequency spectrum from a periodic signal).

The original contributions of this thesis are

- The first quantitative study of the kinematics of crawling human infants
- The design of a locomotion specific oscillator model in which we can independently control the durations of ascending and descending phases of the oscillations
- A design methodology for CPG based controllers that was validated on three different simulated quadruped robots
- The design and mathematical analysis of an adaptive frequency mechanism for oscillators such that they can learn the frequency of any periodic signal
- The use of pools of adaptive frequency oscillators to perform dynamic Fourier series decomposition and to encode periodic patterns in limit cycle systems to construct CPGs

---

**Outline** The remainder of the thesis is organized as follows. In the next chapter we present the context and the work related to this project (Chapter 2). The three following chapters detail the main axes of research. Chapter 3 presents the results on crawling human infants, Chapter 4 presents our design methodology of CPG based controllers for the locomotion of quadruped robots and Chapter 5 discusses adaptive frequency oscillators and their applications. The last chapter (Chapter 6) summarizes and discusses the results of the research.



---

## CHAPTER 2

# CONTEXT AND RELATED WORK

---

This chapter briefly presents the context and inspirations that led to the developments of the research presented in the following chapters. We first present the RobotCUB project, the European project which funded this research. Then we discuss animal locomotion and more precisely its neural control, this will lead us to the control of legged locomotion in robotics. Finally we give a few words on control theory and dynamical systems in the context of this research.

### 2.1 The RobotCUB Project

First, we must talk about the RobotCUB project [122] without which this research would not exist. RobotCUB stands for ROBotic Open-architecture Technology for Cognition, Understanding and Behavior. It is a five years long project funded by the European Commission that has two main objectives. First, it aims at building an open source humanoid robot of the size of a two year old child, the iCub. It is meant to serve as an open research platform in embodied cognition and as such its design is released under a free and open license. Second it aims at modeling and understanding cognitive systems through the use of the iCub platform in the study of cognitive development.

This project has been developed on the fundamental assumption that intelligence emerges from the interaction of the agent with its environment. Cognition is viewed as “the working of a system to preserve its *organization* in face of environmental perturbations. Cognition is an instance of a process of self-organization of co-development between agent and environment” [89]. This idea of co-development is fundamental: cognition can appear only through the interaction of an embodied artificial system with its environment and “one cannot short-circuit the ontogenetic development because it is the agent’s own experience that defines its cognitive understanding of the world in which it is embedded” [142]. Action is then viewed as a fundamental component of development that leads to cognition [145] and it justifies the need for a humanoid robot able to interact with its world.

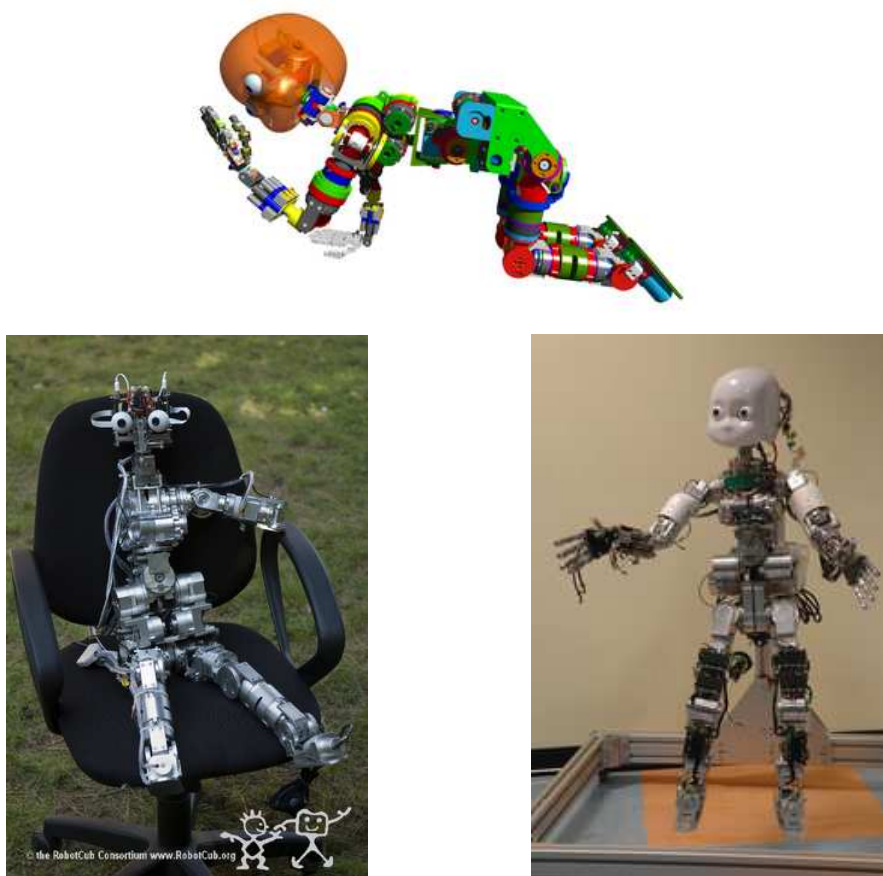


Figure 2.1: Pictures of the iCub at various stages of its development.



## 2.2. Animal locomotion

---

The humanoid robot developed, called the iCub, is a 53 degrees of freedom robot with the dimensions of a 2 years and a half infant (94cm tall and less than 23kg weight) [141]. It is designed to be able to crawl on hands and knees and to sit. Its hands allow dexterous manipulation and its head and eyes are fully articulated [11]. Figure 2.1 shows pictures of the iCub. As part of this project, our goal was to develop the locomotion controller for the crawling iCub. The robot will crawl and not walk (even though it is designed to be able to walk as well) because it should have the basic motor capabilities of an infant. The developed cognitive architecture [142] will then work on top of this basic motor capabilities. One constraint for our locomotion controller was that it should be easily integrated in the whole architecture and that higher level controllers should be able to activate it using simple parameters. The CPG-based approach that we took was then very well adapted to these constraints as we will see later (few input parameters can define complex motor outputs).

Due to its interdisciplinary nature this project involves ten research laboratories in Europe and a robotic company. This brings together roboticists as well as psychologists and neurobiologists. The cognitive architecture [142] that was developed for the iCub is then grounded in recent discoveries in natural sciences.

## 2.2 Animal locomotion

The locomotion of animals is a vast subject that covers fields from biomechanics to neurosciences. It is difficult to give a brief overview on animal locomotion, since when observing different animals locomoting, one is amazed by the diversity present in nature. Even when restricted to quadruped locomotion of mammals, the field of study is huge. Surprisingly enough, invariant quantities can be observed in this diversity.

For example all quadruped mammals use a symmetric walk at slow speeds (in the sense of Hildebrand classification of gaits [59, 61]). When increasing their speeds, they use generally a trot gait or a pace for big mammals such as elephants and giraffes. At the fastest speeds they then switch to gallop gaits. The basic limb kinematics of many quadruped mammals are also very similar [43, 56]. Another interesting observation is that despite one mammal can move at very different speeds using different gaits, we observe a strong correlation between the stance duration (the time that one limb spends on the ground during one gait cycle) and the speed of locomotion while the swing duration (the time during which one limb is off the ground) is kept almost constant over a wide range of speeds and gaits [56, 87, 143].

Maybe one of the most surprising theory on locomotion that illustrates this idea of invariants of locomotion is dynamic similarity [6]. The theory says that animals of different sizes and moving at different speeds can move in a similar way. Animals are considered to be dynamically similar if they are geometrically similar (their linear dimensions can be made equal when multiplied by the same constant), they have same relative timing

of the limb movements, equal duty factor (ratio of stance duration over one step cycle), equal relative stride lengths (step length normalized by leg length), equal relative ground reaction forces and equal relative mechanical power outputs. The hypothesis of this theory is that animals are dynamically similar if they have equal Froude numbers (squared forward velocity divided by gravitational acceleration and animal leg length). Actually they found in [6] that despite large differences in sizes and velocities, animals move in a dynamically similar way at equal Froude numbers.

It is thus very interesting to get insights from animal locomotion by looking at these invariants because they probably emphasize the fundamental principles governing locomotion. We will discuss in more details these aspects of basic gait properties common to different animals in Chapter 3.

### 2.3 Central Pattern Generators

The Central Pattern Generator (CPG) is a concept central to this thesis. It comes from the way the neural system of animals controls their rhythmic movements. The neuronal circuits that can coordinate the muscles to produce rhythmic movements are located in the spinal cord, they constitute the locomotor Central Pattern Generator. They are activated by simple, non-phasic, inputs from the brainstem. CPGs are defined more generally as neural networks able to produce coordinated rhythmic activities without any peripheral inputs (as opposed to a reflex-based mechanism). There exist CPGs for swallowing, chewing, breathing or flying [32]. As explained in [58], the CPG concept refers to a function and not to the detailed neural organization. CPGs are distributed oscillatory centers over the whole spinal cord in vertebrates. Experiments have shown that small sections of the spinal cord of the lamprey were capable of producing rhythmic activities and thus oscillatory centers are distributed along the spinal cord. It has generally been observed that CPGs are organized as coupled oscillatory centers with at least one per articulation.

A very important experiment that illustrates well this concept comes from Shik, Severin and Orlovski [131]. A decerebrated cat (i.e. a cat where the higher part of the brain is removed and only the brainstem and spinal cord remains) is placed on a treadmill. The cat does not move until an electric stimulation is applied on the brainstem. Then the cat stands up and starts to walk. When the strength of the stimulation is increased the cat walks faster until it reaches a trot gait. Then the trot speed increases and finally the gait switches to a gallop. If the speed of the treadmill is adjusted such that it is different from the cat's speed, then the cat is able to adapt its speed and synchronize with the treadmill (up to a certain difference in speeds). Spontaneous walking appears if instead of the electric stimulation, it is the treadmill that entrains the limbs and for various speeds of the treadmill we observe the same gait transitions as before. The decerebrated animal can support its body weight but does not have control of balance.

### 2.3. Central Pattern Generators

---

CPGs for locomotion have been found in a wide variety of vertebrates located on both ends of the phylogenetic ladder (see for example [9, 57, 58, 67, 100] for reviews). It is in primitive vertebrates such as the lamprey or the salamander that these mechanisms are best understood. For example it was shown that the isolated spinal cord of both animals can produce *fictive locomotion*, i.e. locomotion like patterns of the spinal cord but in isolated preparation outside the body (see [25] for the lamprey and [33] for the salamander). But their detailed neuronal organization is not well known yet since many neurons are involved and things become even more complicated for other vertebrates such as mammals or birds. In humans there exists good evidence that the CPGs play an important role [34] and many rehabilitation treatments for patients with spinal cord injuries are based on these concepts [91, 150], the spinal cord having some sort of plasticity [118].

However the role of sensory feedback in movement generation should not be underestimated, especially when one studies mammalian locomotion. Indeed, the experiment on cats that we discussed previously shows that the CPG can be activated by mechanical entrainment and thus that there is a significant coupling between the body and the CPG. The same type of mechanical entrainment can be seen in lampreys and salamanders. But the nature of the interaction between sensory information and the CPG is much more complex than only entrainment behavior. Indeed, sensory information from cutaneous and muscle receptors is continuously integrated in the locomotor CPG and they can strongly modify its activity [101, 119]. Moreover the effect of this sensory information is phase dependent, it means that the same input will have different effects depending on the state of the CPG. In addition to the CPG, there are reflexes that modulate muscles activities in faster control loops. These reflexes are also phase dependent as for example the stumbling corrective reflex during swing phase to step over an obstacle when the dorsal surface of the hind foot is touched. During stance the same sensory excitation leads to different contraction patterns of the muscles and the limb does not try to step over an obstacle. And the situation becomes even more complex when we consider integration of vestibular information (for balance control) and vision. An in depth review on sensorimotor interactions during locomotion can be found in [119] and a review from the modeling point of view in [44].

We find models of CPGs at different levels of abstraction, each one serving to explain different phenomena. The most detailed ones are based on Hodgkin-Huxley neuron models that include representation of the ion channels and are mostly used to understand rhythmogenesis (i.e. generation of rhythmic activity in small neural circuits [140]). Simpler neuron models such as leaky-integrator models are also used to understand the importance of network properties to generate rhythmic properties and coordination between oscillatory centers (e.g. [39, 151]). More abstract models based on coupled nonlinear oscillators to simulate population dynamics try to understand interlimb coordination and gait transitions induced by the topology of the network [24, 27, 26, 52, 53, 80, 65, 127], independently on the particular dynamics of the oscillators. Neuromechanical models

have also been developed to understand the effects of the interaction of the CPG with a mechanical body through sensory feedback [39, 66, 70, 136, 137, 138] as well as implementations on real robots [46, 65, 69]. In this thesis we use coupled oscillators similar to those used in theoretical biology to construct our CPG models for the control of locomotion.

## 2.4 Control theory and optimization

Control theory is very wide topic. Its core deals mainly with the problem of stabilizing a system at a given state, to make the system follow a trajectory in state space and whether the system is controllable (i.e. if we can drive the system from any state to any other). The largest part of control theory was developed for linear systems. They are simpler to analyze and to control and most of the time nonlinear plants are linearized. Then robust control methods can assert the stability of the system under small parameter uncertainties (or nonlinearities) and adaptive control methods can fine tune the unknown parameters of the plant. Control of nonlinear systems is a growing field and many different methods have been developed from feedback linearization to sliding mode control or geometric control (see for example [20, 132]). However all these methods never address the problem of finding the trajectories that the plant should follow, which is one of the problems we try to address in this thesis.

Optimal control tries to solve this problem. The traditional formulation of the problem is to find a control trajectory for a system such that it minimizes a cost function over time (see [139] for an introduction). The main problems of optimal control is that it works for low dimensional systems and that, unless the problem can be formulated as a linear quadratic problem, convergence of the algorithms is difficult and computationally expensive which make them hard to be used for online trajectory generation. It is even worse in the case of non differentiable systems, which include the problem of locomotion of legged robots because of intermittent contacts. However some recent results showed that these problems can be solved to a certain extend (see for example [78, 121]). Most of the time these problems are formulated as local optimization problems in order to find numerical solutions.

There exist many learning or optimization algorithms that can be used to solve the problem of finding control policies. They are all based on the idea of optimizing a cost function. These cost functions can be very abstract as for reinforcement learning algorithms (see for example [79, 102]) or very explicit such as the one used in supervised or statistical learning. From the optimization point of view, there are many different types of algorithms. Gradient based methods are very powerful to find local optimum but one need to be able to define a gradient of the function to optimize, which is not always feasible. There are also gradient free methods, among the most popular are stochastic optimization algorithms such as genetic algorithms and simulated annealing. The interest of these algorithms is that they work well in complicated and high-dimensional spaces

## 2.5. Control of legged locomotion

---

and are robust to the problem of local optimality. However there is no guarantee that the algorithm converges properly.

All these different algorithms have shown very good results in many applications. They can be viewed as different faces of the same problem. The common aspects of these methods is that they always need an explicit cost function to be optimized and the actual choice of this cost function is most of the time more difficult (and critical) to find than actually solving the optimization problem. Moreover the computational needs to solve such problems are very high. Another aspect of such algorithms, that can be viewed as an advantage or disadvantage depending on the problem, is that there is an explicit separation between the learning (or optimization) algorithm and the learning substrate. On the contrary, Nature does not separate these two aspects, in the brain the learning algorithm is implemented in the learning substrate. It is a conceptually very different way of seeing optimization or learning. Nevertheless I do not claim that this view is better but only that it may help to find new approaches for solving optimization problems. In Chapter 5 we present an example of such a view, namely adaptive frequency oscillators, where the learning algorithm and the learning substrate are not distinguishable. In that case the cost function is implicitly defined in the dynamics of the system.

## 2.5 Control of legged locomotion

Traditional approaches to find control policies for the locomotion of a legged robot are based on optimal control and optimization methods that we discussed before. For biped locomotion the idea is to define a stability criterion such as the ZMP [146, 147] and then to find offline a trajectory that satisfies this criterion under some constraints (e.g. torque limits). However the stability obtained in this way is limited and small perturbations can lead to a fall. Moreover ZMP-like criterion are difficult to define for non flat terrains and most of the time cannot cope with running gaits since they are not defined during the lift-off phase (no legs touch the ground). Ultimately the robot tracks the reference trajectories found using the previous criterion using different methods such as impedance or force controllers (e.g. recent development of whole-body controllers [63, 128]). Heuristic methods are then developed to online adapt the trajectories to improve stability of the robot (e.g [149]). Recently a formulation of the problem as a model predictive control problem allows to generate online control trajectories from step to step (see for example [35]). But in that case the computational cost is still very important and difficult to implement in realtime.

There are also methods based on physical insights such as virtual model controllers that can be implemented online [105, 106, 107] at a moderate computational cost. The interest of such methods is that they are quite robust on unknown terrains and it is easy to integrate passive elements in the robot. However because there is a need of a finite state machine to switch between different control phases, such approaches may not be

able to benefit of entrainment and resonance phenomena with compliant robots.

The approach we follow in this thesis for the problem of online trajectory generation is based on CPG like controllers. These methods are inspired by the control of locomotion in animals and gained interest since the seminal work of Taga [135, 136, 137, 138] who showed that a system of neural oscillators (based on the model of Matsuoka [85]) coupled with a biped mechanical model was able to produce stable walking in unpredictable environments. The model is quite complex and involves many parameters but manage to show that stable biped locomotion is possible through mutual entrainment of the oscillators and the mechanical system they control. Since then many successful applications of this concept, most of the time using simpler controllers, were applied to control the locomotion of robots, from snake and salamander like robots [65, 68] to quadruped [14, 15, 46, 75, 77] and biped robots [7, 8, 40, 41, 93, 95]. Details on specific approaches are given in Chapter 4.

Usually robots are built as fully actuated rigid machines, however for locomotion tasks requiring high agility such robots can become very slow and energy consuming. Moreover very fast control loops are needed in order to compensate for disturbances. Adding compliant elements in robots is generally not desirable because it makes the control problem and state estimation harder. But for legged robots, it might to some extent simplify the problem because this can help to reject small perturbations via self-stabilization mechanisms and exploiting the natural dynamics of the robot can lead to very energy efficient solutions. For example passive dynamic walkers exploit natural inverted pendulum dynamics for locomotion and it leads to highly energy efficient locomotion [28, 29, 86]. Recently several robots with passive elements showed very good performances, from quadruped robots with the Tekken robot [75] and the Puppy robot [64] to hexapod robots [23]. Even the BigDog robot [104], which is certainly the most advanced quadruped robot in the world, use some compliant elements. Unfortunately very few informations are available on this robot. CPG based approaches offer interesting solutions to control such robots since their natural synchronization properties can be used to exploit the passive dynamics of the robot. Recent CPG controllers have shown very good performances with such robots [17, 75, 76]. We will see in Chapter 4 that our CPG model while working very well with traditional robots is also very effective with robots having passive dynamics.

## 2.6 Dynamical systems

In this thesis we make an extensive use of dynamical systems and more specifically of oscillators because of their synchronization capabilities. Synchronization is a very well studied phenomena, see for example [3, 81, 103] for good references. Oscillators are therefore well suited to model CPGs and were extensively used in theoretical biology [24, 27, 26, 52, 53, 80, 65].

## 2.6. Dynamical systems

---

Recent results on dynamical systems theory insisted on the importance of the topology of coupling between oscillators for synchronization phenomena rather than the detailed dynamics of the oscillators [12, 50, 54, 51, 148]. We show in Chapter 4 how we can use these ideas to easily construct CPGs.





---

## CHAPTER 3

# CRAWLING HUMAN INFANTS

---

Prior to present the work done in robotics and dynamical systems in the next two chapters, I would like to discuss our work on analyzing crawling in human infants. I believe this research is interesting mainly for two reasons. First, from the scientific point of view, this is the first quantitative study on crawling human infants and it is quite striking to see all the similarities between the gaits of these infants and those of other quadruped mammals. Second, this is the occasion to give a view on locomotion from the biological side and to point out several important concepts that will be used in the design of our controllers in the next chapter.

Why studying crawling infants? In the RobotCUB project, our goal was to design a crawling controller for the iCub humanoid and we wanted to have a locomotion that resembled the one of infants. However there existed very little research on crawling and almost no quantitative information was available. Fortunately, there were developmental psychologists involved in the project and they proposed to make experiments with crawling infants to collect some quantitative data in order to have a better intuition on how infants were crawling. As the first results of these experiments were very interesting and unexpected, we decided to make more precise experiments and to make a careful analysis of the data.

All the experiments were conducted at the Babylab in Uppsala University in Sweden by Kerstin Rosander and Anna Nylén, and I am really grateful to them for the time they took doing the experiments since it is far from easy to get infants to crawl in front of a motion capture system with markers attached on them. The analysis of the data was done by me.

The next sections present the results from these experiments, the text is taken from a journal paper that is currently under review [115].

## 3.1 Introduction

Most humans, before standing and walking on two feet, start to locomote on their four limbs. Usually, infants start crawling around 9 months [49] and continue until they start walking. When locomoting on four limbs, infants can have very different strategies for crawling [5], but the gait that is most often observed is alternated locomotion on the hands and knees (which we call the standard crawling gait).

Despite being the first locomotion behavior humans have, there are very few studies of the human crawling gait and most of them have been done from the developmental point of view. It was shown for example that the head orientation and hand preferences while rocking affected the onset of crawling [49] and that the different locomotion strategies (e.g. crawling on the belly) before an infant crawled on four limbs had an impact on the future efficiency of standard crawling [5]. Albeit it was reported by Mucino et al. [94] and Niemitz [96] that the crawling gait has a duty factor higher than 50% and that the diagonal limbs move together, the experiments were done on one and two infants respectively and no quantitative data were available. Therefore, to the best of our knowledge, there are no quantitative studies available about the crawling gait.

Nevertheless, human infant crawling locomotion offers a very interesting subject of research in two aspects. First, infants have a posture that is different from other quadrupeds and that is not optimized for quadrupedalism. Indeed, crawling infants have only two functional limb segments for the fore limb (arm and forearm) and one for the hind limb (since knees are on the ground). Consequently, studying crawling infants allow one to study quadrupedalism in one of its simplest form. All the gait characteristics of infants common to other mammals would suggest that these are independent of the functional limb geometry of the quadruped and therefore emphasize the importance of the neural control and the constraints imposed by quadrupedalism in the emergence of these common characteristics. Second, from the human locomotion point of view, it is interesting to know what strategy infants find to locomote and how it is related with primate gaits, which are known to be different from other quadrupeds [83, 124, 126]. In this chapter, we are therefore interested in answering two main questions. First, despite of their simple, not optimized, limb geometry how different is the infant crawling gait from other quadrupeds gaits? Second, how does this locomotion relate to non-human primate locomotion?

All quadruped mammals share several basic principles for their gaits, both in terms of their temporal characteristics, kinematics and neural control [43, 56, 87, 100, 143]. First, most mammals have different gaits for different speeds, starting from slow symmetrical gaits such as walk to faster gaits like trot and pace and finally asymmetric ones such as gallop (in this chapter we use the gait definitions of Hildebrand [59]). In contrast, there are no report of gait transitions in infants, they mainly use a single crawling gait (even if there exist different gaits among infants). In vertebrate quadruped locomotion, the duration of stance phase is directly related to the locomotion speed and the swing phase

## 3.2. Materials and methods

---

stays almost constant for most speeds [87, 143]. Mammals share also a lot of similarities in their basic limbs kinematics (similar one period flexion and extension of the shoulder and hip, two period flexion/extension of the more peripheral joints, lateral and sagittal movement of the spine) see for example comparative results in [43, 56, 129, 143]. However primate locomotion can be distinguished from other quadrupeds in several ways. Despite most mammals use a lateral sequence walking gait (left-hind (Lh), left-front (Lf), right-hind (Rh), right-front (Rf)), primates use mainly a diagonal sequence gait LhRfRhLf [60, 144] with grasping hind feet, even if lateral sequences can be observed (i.e. for infants), the diagonal sequence is always the dominant one [130]. It was proposed that this gait was first evolved for fine branch locomotion [126]. Primates also have a more protracted arm at touch down (over 90° relative to horizontal body plane) than other mammals [83]. The aim of this study is thus to give a detailed description of the standard crawling gait in human infants and to compare its characteristics with other quadrupeds and especially to primates. These comparisons will be done in terms of 1) the basic limb kinematics, 2) the relation between speed of locomotion and swing/stance durations, 3) the preferred footfall sequences and 4) the coordination between the limbs and the spine.

## 3.2 Materials and methods

### 3.2.1 Subjects

Nine infants, from 9 to 11 months old have been studied. They were selected as crawlers practicing the standard gait using hands and knees in an alternated fashion. The parents were asked about approximate start of crawling. Three infants were seen twice. We had to discard the data of 2 infants since we did not get complete data on steady crawling. These infants were not making enough steps while crawling during the experiments, they often made only one step and stopped. More details are shown in Table 3.1 for the 7 infants whose data were considered.

### 3.2.2 Procedure

When the parents came to the lab they were informed of the experiment and signed a consent form that included permission for video recording. The parents undressed the infant and 18 small markers were attached to the skin on places on or close to the joints. Three markers were put on the spine (neck, thoracal and lumbal). A hat with three markers (1 midsagittal, 2 coronal) was put on. The markers on the wrists and knees were glued to a velcroband. This gave stability to the critical parts that were close to the floor during locomotion. One disadvantage was that the knee markers were just above the joint. The remaining height markers (elbows, shoulder, hips and feet) were attached with collars used for skin electrodes. When all 18 markers were properly attached the

Infant name	Age [days]	Experience [days]	Number of complete cycles				Weight [kg]
			Left arm	Right arm	Left leg	Right leg	
A.	253/296	28/71	17/6	16/10	13/2	16/7	7/7.5
Al.	273/301	21/49	12/5	10/8	10/6	11/7	11/12
E.	286/332	15/61	16/5	15/6	8/6	10/6	9.5/10
J.	290	59	6	5	5	5	9
M.	304	39	12	13	11	13	11
O.	319	89	1	5	4	2	10.5
V.	290	21	13	7	10	3	10

Table 3.1: Date of birth, experience of crawling (i.e. the number of days since estimated start of crawling), number of complete steady crawling cycles that were extracted from the experiments for each limb, and body mass the day of the experiment for the seven infants (note that A., Al. and E. were seen two times)

infant was encouraged to crawl on a rug (polypropylene, size 230 x 170 cm) on the floor. The parent and one experimentator were sitting on the floor on opposite sides of the rug using attractive toys to catch the infants attention. The second experimentator handled the measurements, and was sitting close to the rug observing the infants behavior.

### 3.2.3 Measurements

A motion capture system, Qualisys, with passive markers (size 5 and 10 mm) was used in an external triggering mode. Data was collected at 240 Hz for 12 s periods. In close synchrony with the measurement sessions, a camera monitored the infant during the trial. Before each experiment the system was calibrated. Five Qualisys cameras were used, two were placed at a ceiling stand and three were placed on the floor so that the crawling area was covered (Fig. 3.1). When the infant showed intention to start crawling, the measurement was started by the second experimentator. Usually between 20 and 40 trials per infant were recorded.

### 3.2.4 Data evaluation

The raw data was then processed in order to interpolate for missing data in small time intervals and to remove high frequency noise. In order to do that, we used two consecutive methods. First we interpolated the missing data using a piecewise cubic Hermite interpolation. We did not perform any extrapolation of the data at the beginning and end of each time series. Then, in order to remove the noise, we smoothed the data using a locally weighted scatter plot smoothing using least squares linear polynomial fitting, with 20 data points for each local smooth calculation (span of 83 ms). An example of processing is given in Figure 3.2.

### 3.2. Materials and methods

---

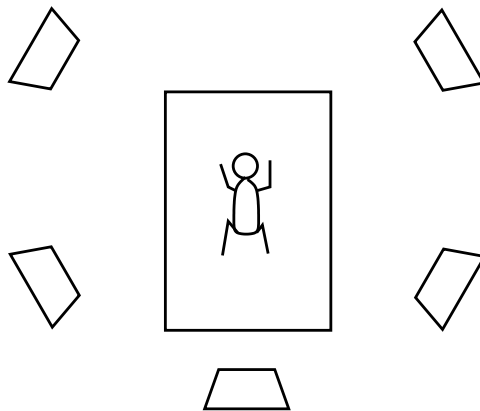


Figure 3.1: Configuration of the experimental setup, with the 5 cameras and the crawling scene (top view).

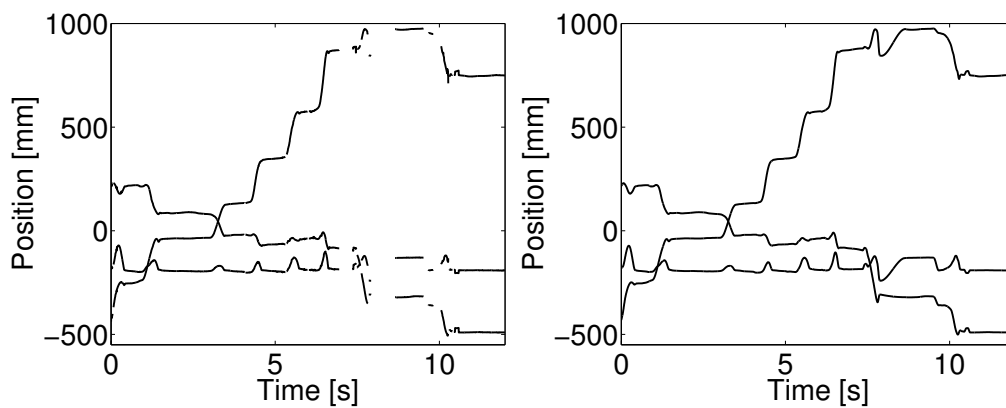


Figure 3.2: This figure shows an example of preprocessing of the data, the left graph is raw data before processing and the right one shows the data after processing. The data corresponds to the position of the left hand in 3D (each line is a dimension).

### 3.2.5 Swing - Stance measurement

In order to study steady state crawling and to have comparable data, we selected only the crawling sequences in which the infant was crawling straight toward a goal and without stopping to do something else. Thus a stance phase was always measured between 2 swing phases. Although this limitation lowered the quantity of data, the analyzed data are then comparable because the infants are doing steady crawling and not superposing another movement on it (like reaching for an object, preparing for sitting, stopping and looking around). Table 3.1 shows the final number of complete cycles we have for each infant.

The swing phase of a limb is the phase during which the limb does not touch the ground. In order to find this phase in our data set, we defined the swing phase for the arms as the phase during which the hand is moving. We defined similarly the swing phase for the legs as the phase when the knees are moving forward. In order to calculate the onset of the swing phase, we calculated the squared time derivative of the position of hands and knees markers and define this phase as the phase when the velocity is bigger than a threshold. We use the squared time derivative to have always positive values of speed and because squaring increases the difference between very low and high velocities. The threshold is defined as the value just above the maximum value found during the middle of the stance phase (Fig. 3.3). We always used the videos to check the consistency of the measures.

### 3.2.6 Statistical measures

Below, we use the median and the interquartile range estimators for the data instead of the mean and standard deviation estimators because these estimators are more robust against noisy data and outlier values [92] and allow for the use of non parametric tests that does not account for a Gaussian distribution of the data. Whenever it is needed, we use Spearman correlation tests and Wilcoxon rank sum tests.

## 3.3 Results

This section details the results of the data collected during the crawling experiments. We first characterize the crawling gait in details and then we study the kinematic data of the 4 limbs in the sagittal plane and the kinematics of the spine in the horizontal plane. Finally we study the effects of experience on the gait.

### 3.3. Results

---

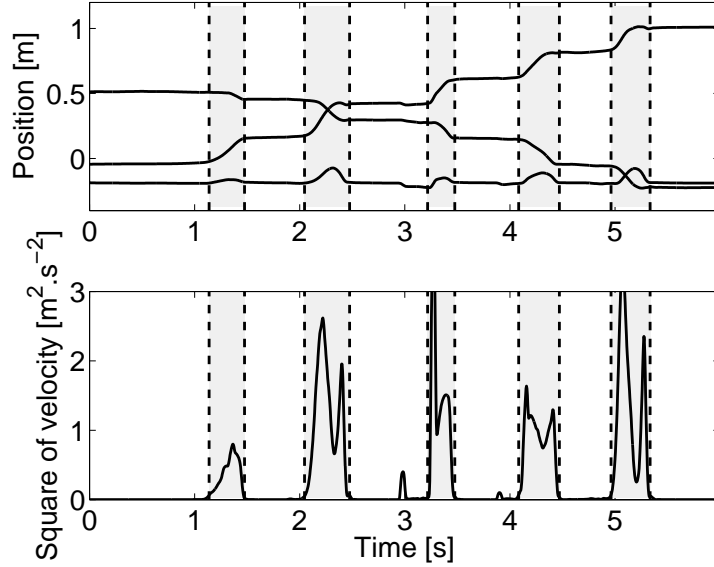


Figure 3.3: This figure shows an example of the velocity profile during the swing and stance phases of the left arm of an infant (lower graph) and the corresponding time series of 3D positions (upper graph). The vertical lines show the separation between the swing (in gray) and stance (in white) phases.

#### 3.3.1 Gait analysis

##### Swing and stance durations

All the 7 infants studied here were locomoting using a standard crawling gait, i.e. crawling on hands and knees (this was the criterion of selection). Diagonally opposed limbs move almost in synchrony and are half a period out of phase with the other limbs. The left arm is synchronized with the right leg and they are half a period out of phase with the other limbs. The swing phases of the ipsilateral limbs never overlap. This gait thus resembles a trot gait in terms of the temporal symmetries between the limbs.

We calculated the mean duty factor (stance period of the hind legs as a percentage of the stride duration) and the diagonality (the percentage of the cycle period by which the left hind footfall precedes the left fore footfall) of the gait for each infant as defined by Hildebrand [59]. The duty factor is comprised between 54 and 66%, as can be seen in Figure 3.4. The diagonality is between 33 and 40%. Thus all infants have a gait between a walking trot and a lateral sequence diagonal couplets walk. Figure 3.5 shows the typical footfall sequence of this gait.

In Figure 3.6 we show the median values of swing and stance durations for each infant, together with their interquartile ranges. We clearly notice the small variability in the durations of the swing phase, both within each measured infant and between all the

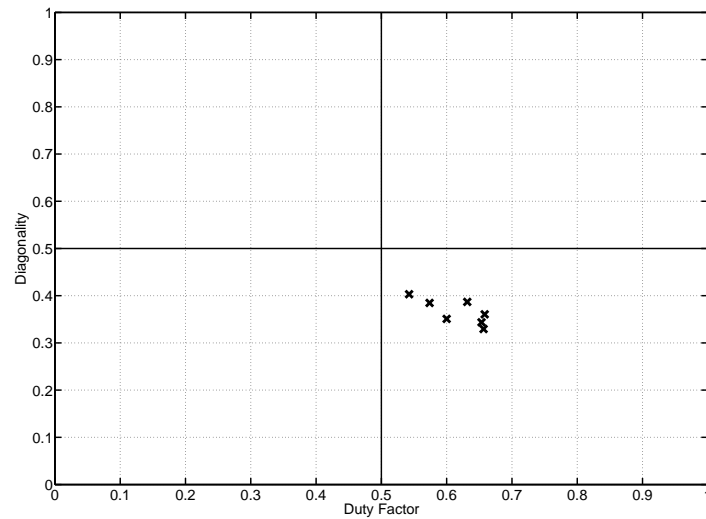


Figure 3.4: Hildebrand diagram for the seven infants.

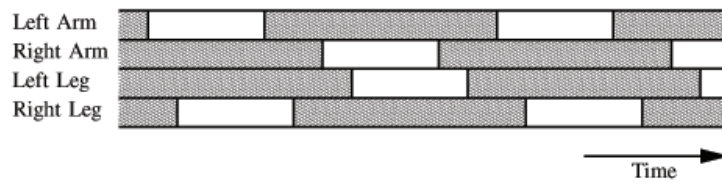


Figure 3.5: Typical footfall sequence of the infant standard crawling gait. The gray boxes show the stance phases and the white ones the swing phases.



### 3.3. Results

---

infants and we also notice the high variability in the durations of the stance phase, both within the measurements of each infant and when we compare the infants.

The median duration of the swing of the arms is between 300 and 446 ms for the 7 infants, the median duration of the swing of the legs is comprised between 354 and 554 ms. Thus the infant with the slowest median arm swing is about 49% slower than the infant with the faster arm swing. For the swing of the legs, this corresponds to about 56%. The variability of the swing phase (ratio of the interquartile range of the swing duration with its median duration for one infant) has a median of 16%.

Compared to this, the median duration of the stance of the arms is comprised between 367 and 1035 ms, the median duration of the stance of the legs is between 373 and 975 ms among the infants. In the case of the stance, the variation between the slowest median stance and the fastest is about 182% for the arms and 161% for the legs. The variability of the stance duration (ratio of the interquartile range of the stance duration with its median duration for one infant) has a median of 28%.

From these observations, we can conclude that the variability in the median duration of the stance phase is bigger than the one of the swing phase. Furthermore the main timing parameter that varies between the infants is the duration of the stance phase. The same conclusions can also be drawn for the variability of the measurements for each infant.

#### Relation between speed and cycle duration

We now investigate the influence of the durations of swing and stance on the speed of locomotion of the infants. We measured the displacement of the markers located on the spine of the infants during each crawling sequence to evaluate the crawling speed.

Figure 3.7 shows respectively the cycle frequency,  $1/\text{stance duration}$  and  $1/\text{swing duration}$  as a function of the speed of locomotion. We find a strong linear relation between the frequency of the cycle and the speed of the infants (correlation of 0.860,  $p < 0.001$ ). We also observe that this strong linear relation also exists with the inverse of the stance duration (correlation of 0.820,  $p < 0.001$ ), that certainly explains the high variability of stance duration seen previously. We also notice that there is no correlation with the swing duration (correlation of 0.105,  $p=0.34$ ). Thus the duration of the stance has the most important influence on speed locomotion and the duration of the swing phase is not correlated with it. As will be discussed later, this is in agreement with observations made in vertebrate quadrupeds.

#### 3.3.2 Kinematics

In this section, we are interested in the detailed kinematics. We study 5 different degrees of freedom of the infant: the angle of the shoulder in the sagittal plane and the elbow for both arms, the hip degree of freedom in the sagittal plane and the knee for both legs and

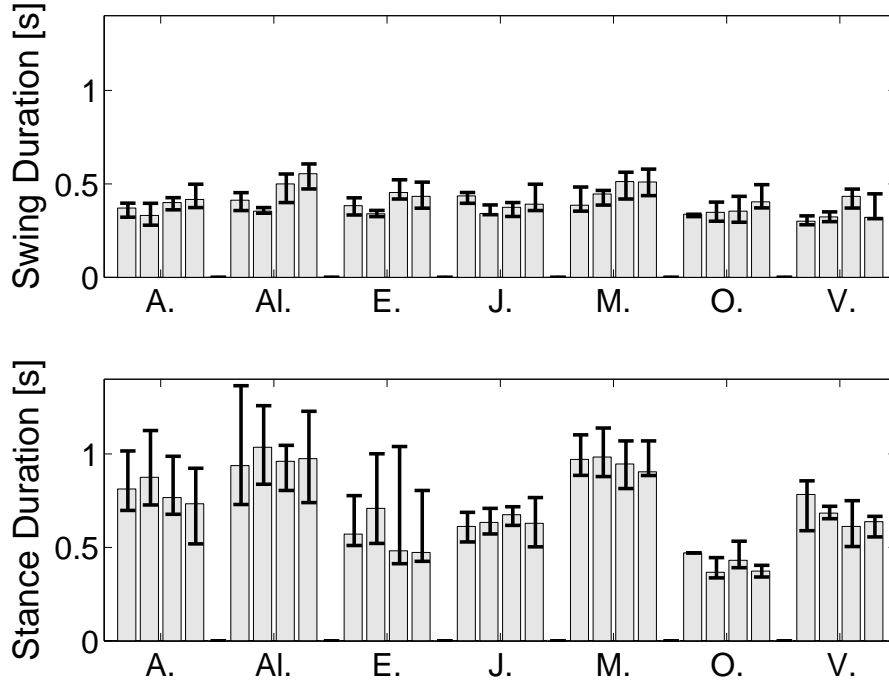


Figure 3.6: Median duration of the swing (top figure) and stance phases (bottom figure) with the interquartile range as error bars. Each group of data represents an infant, the four bars represent the left arm, the right arm, the left and right legs respectively.

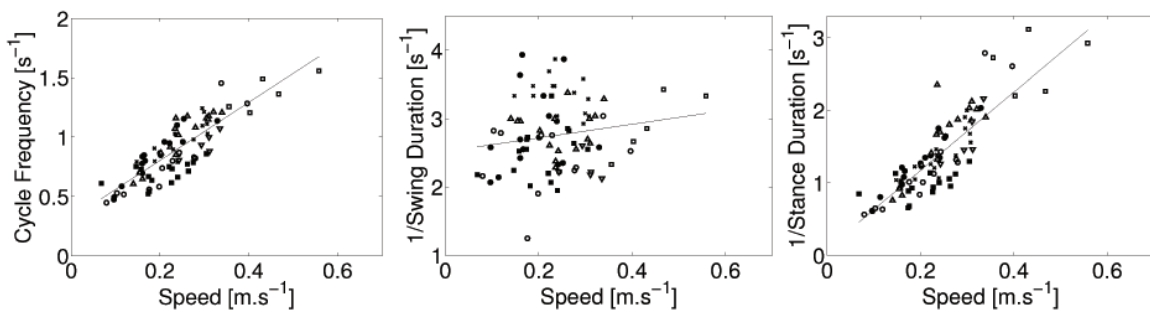


Figure 3.7: Cycle frequency, 1/stance duration and 1/swing duration as a function of the speed of locomotion for 7 infants. The line shows the linear relation obtained by linear regression on the data. The legend is as follow: ● for A. ○ for Al. △ for E. ▽ for J. ■ for M. □ for O. and × for V.

### 3.3. Results

---

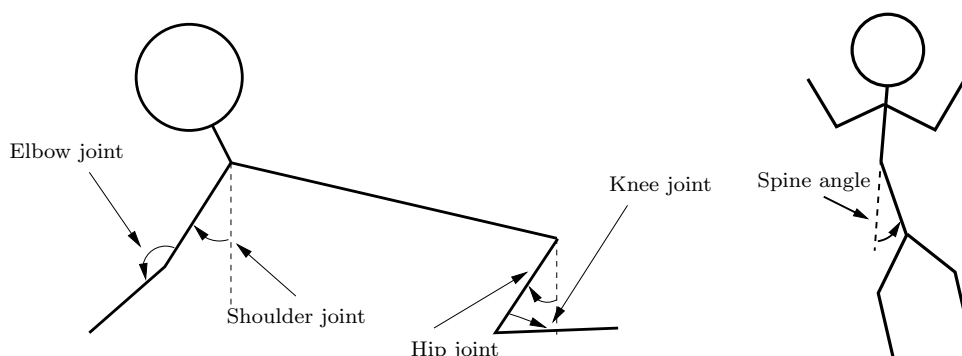


Figure 3.8: Schematic of the joint angles we measured. We look at the movement of the limbs in the sagittal plane and the spine in the horizontal plane

the spine angle in the horizontal plane. Figure 3.8 shows the different degrees of freedom of the infant, the shoulder and hip joints have angle 0 when they are vertical, we used the same measurements as in [84] to be able to compare the limb excursion patterns. The elbow and knee joints are taken to be 0 when they are completely folded (note that it is physically impossible to have a  $0^\circ$  angle). Stick figures of a typical crawling sequence of an infant can be seen in Figure 3.9.

#### Kinematics of the arms

Figure 3.10 shows the median angular values of the degrees of freedom of the arms for each infant. To calculate the median values for each infant, we first separately scale in time the swing and stance phases for each trajectory by means of linear interpolation and then calculate the median value of all the data set for each point in time. We set the duration of the swing to 40% of a complete cycle and the stance to 60%.

During the swing, the shoulder is mainly flexing and we notice an extension of the joint at the end of the swing, before touch down. We notice a quite protracted arm posture at touch down, where the joint angle is between  $14$  to  $39^\circ$  for the different infants (median  $24^\circ$ ). The movement of the shoulder during stance is mainly an extension to make the body move forward, lift off happens at joint angles between  $-10$  and  $-34^\circ$  (median of  $-26^\circ$ ). We also note that the movement of the limbs is qualitatively the same for every infant, only the total excursion angle changes. Its range is between  $39$  to  $72^\circ$  with a median total excursion angle of  $46^\circ$  among the infants.

The elbow joint is flexed at the beginning of swing and extended at the end. It stays almost in the same position during stance, which is characteristic of a non compliant gait. The main difference between the infants is the amplitude of movement of the elbow. The median excursion angle of the elbow movements is  $33^\circ$ . We also calculated the correlation between the speed of locomotion and the amplitude of movement of the shoulder joints but we found no statistically significant correlation (correlation  $<0.15$  and p-values  $>0.7$ ).

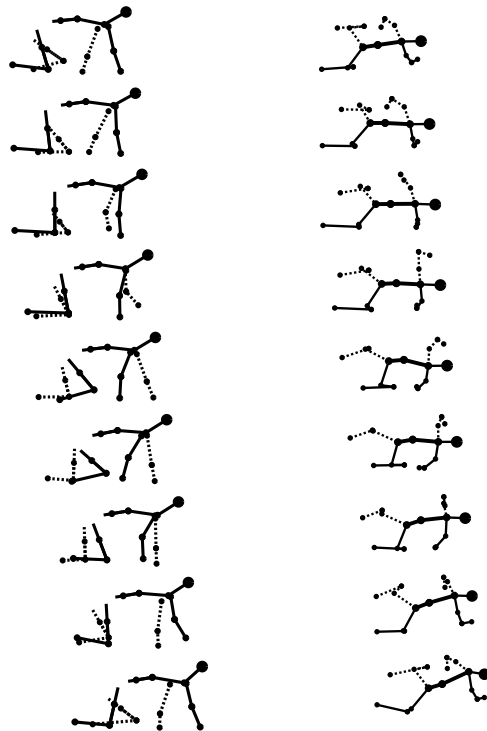


Figure 3.9: Stick figures from real data of a crawling sequence. The left graph shows a lateral view of the infant, the dashed line representing the left limbs. The right graph shows the same sequence from a top view, so the movement of the spine in the horizontal plane is visible. The black dots represent the position of the markers on the infant. There is a 100 ms duration between each figure.

### 3.3. Results

---

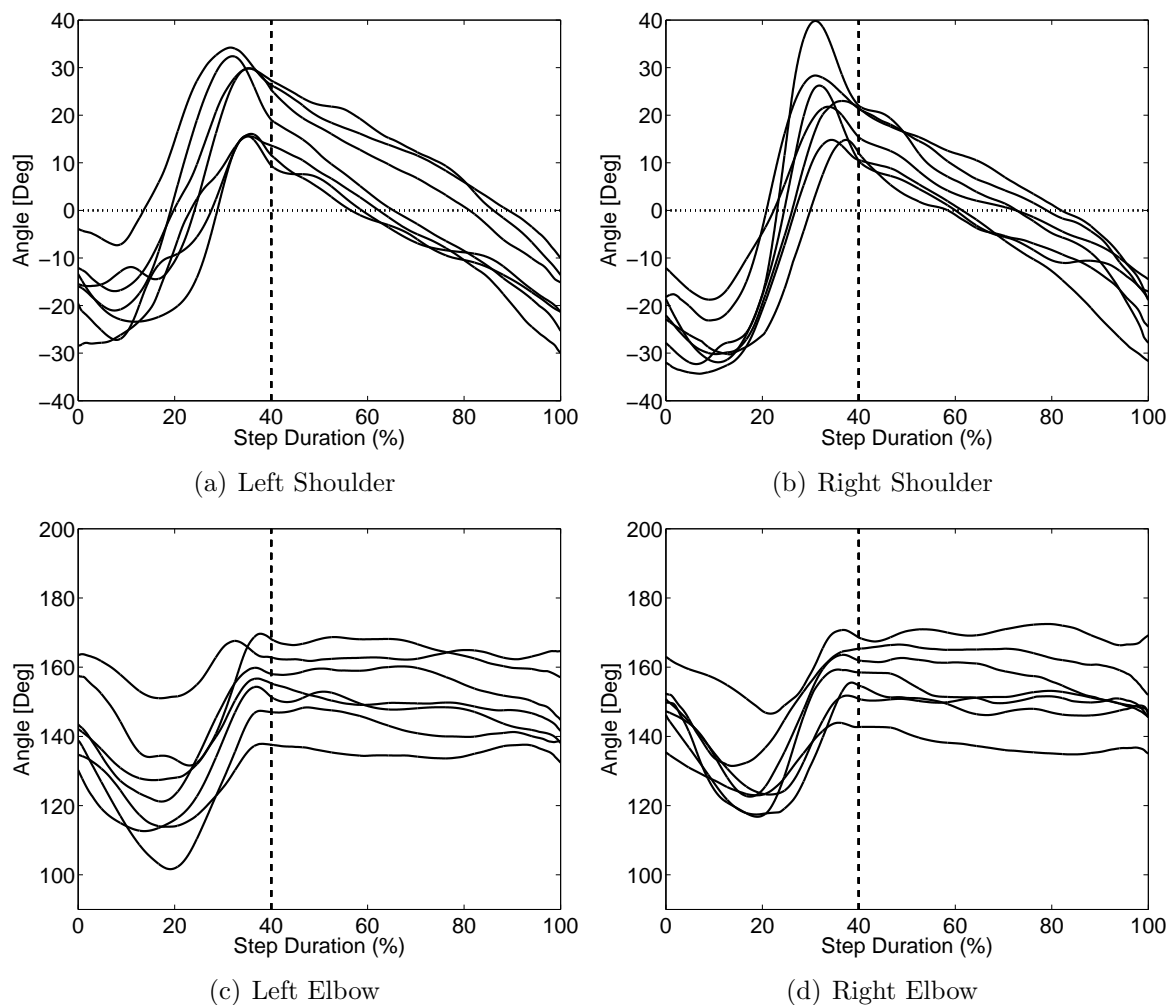


Figure 3.10: Median fore limbs kinematics for the seven different infants. We show the shoulders and elbow angles. For each infant, we rescaled the trajectory into normalized swing and stance phases. The vertical dashed line indicates touch down of the hand.

### Kinematics of the legs

Figure 3.11 shows the median angular values of the degree of freedom of the legs for each infant and their velocity profile. As for the arms, these angular values are rescaled and separated into swing and stance phases.

During swing, the hip joint mainly flexes with a slight extension before touch down of the knee, at touch down the hip is very much protracted with an angle between 26 to 64° (median 44°) at touch down. During stance the hip is extended, with an angle at lift off between -31 to 5° (median -11°). This behavior is qualitatively similar to the movement of the shoulder joints although the extension before touch down is less visible than for the arm movement. The qualitative behavior of the legs is the same for all the infants, except that the amplitudes of motion are different. The total excursion range goes from 52 to 75° (the median is 57°).

The knee angle motion is quite important, even if the knee is always on the ground during stance where it is mainly used as a pivot around which the hip rotates. The median amplitude of movement of the knee is 35°. We notice that the knee flexes during swing and extends during stance.

As for the arms, we did not find any significant correlation between the amplitude of the hips and the speed of locomotion of the infants (correlation <0.25 and p-values >0.5).

### Kinematics of the spine

Figure 3.12 shows the median movement of the spine angle in the horizontal plane for each infant. This median movement was rescaled in the same way we did for the other joints, except that it was also centered on 0. Furthermore, we show 2 distinct figures, in which we separated the movement into 4 different phases. For the left figure, the phases are swing of the left arm, moment during the 4 limbs are on the ground, swing of the right arm, moment during which the 4 limbs are on the ground. The right figure shows the same information, except that the separation is done according to the swing of the legs (swing of right leg, complete support, swing of left leg and complete support). A positive value for the spine angle means that the spine is folding in the left direction (the left arm and left leg are closer). During the swing phase of the left arm, the spine is moving from a positive angle to a negative one (same values in magnitude). Then when the 4 limbs are on the ground, the spine is almost not moving. At the onset of the swing phase of the right arms, the spine is going back to the position it has at the beginning of the swing phase of the left arm. Finally when the right arm touches the ground, the spine does not move until the left arm starts its swing. Interestingly, we notice that the spine has an oscillatory movement which is synchronized with the swing phase of the limbs. The median amplitude of this movement is 23° with interquartile range of 9. We notice that the maximum curvature is attained after the swing of the arms and most of

### 3.3. Results

---

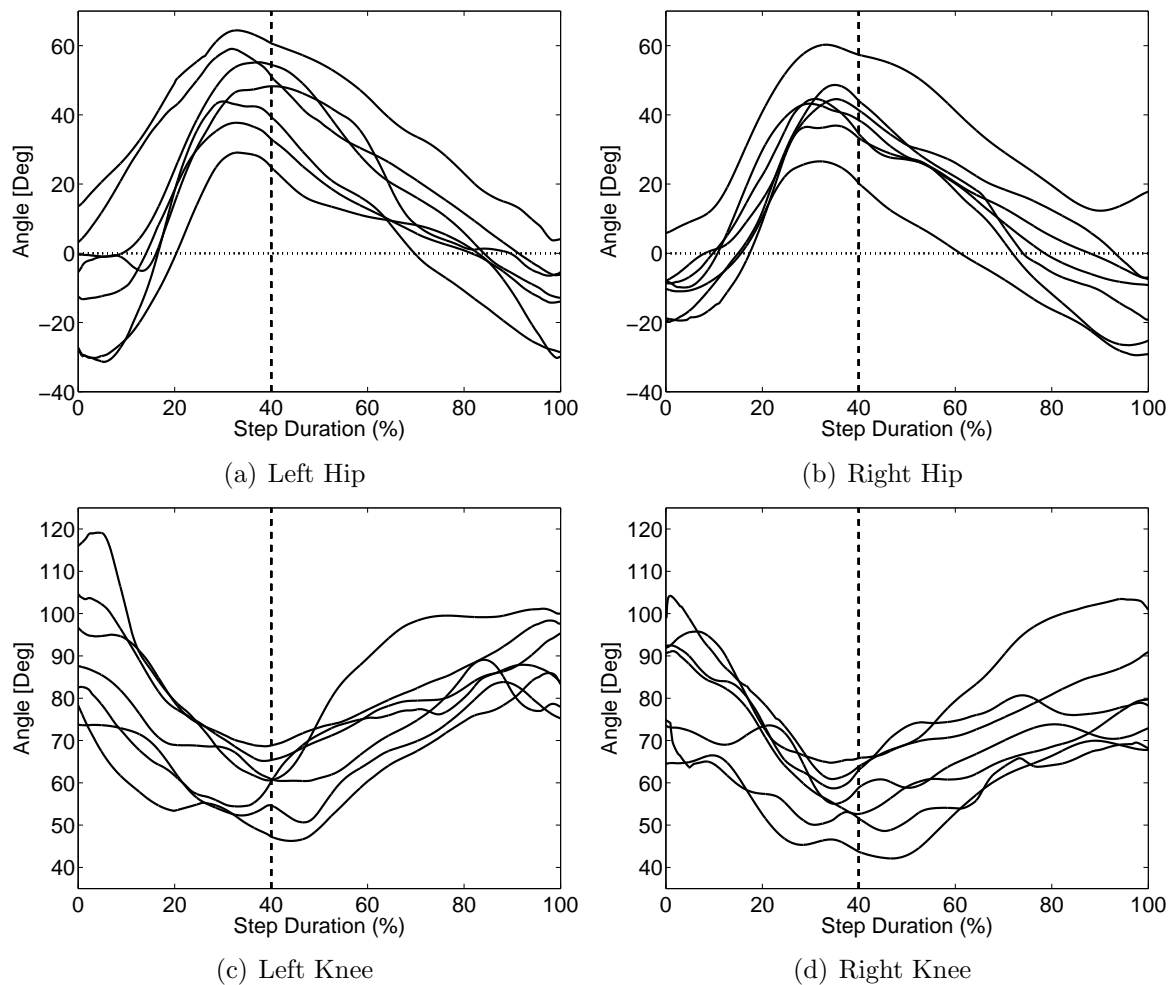


Figure 3.11: Median hind limbs kinematics for the seven different infants. We show the hips and knees angles. For each infant, we rescaled in time the trajectory into normalized swing and stance phases. The vertical dashed line indicates touch down of the knee.

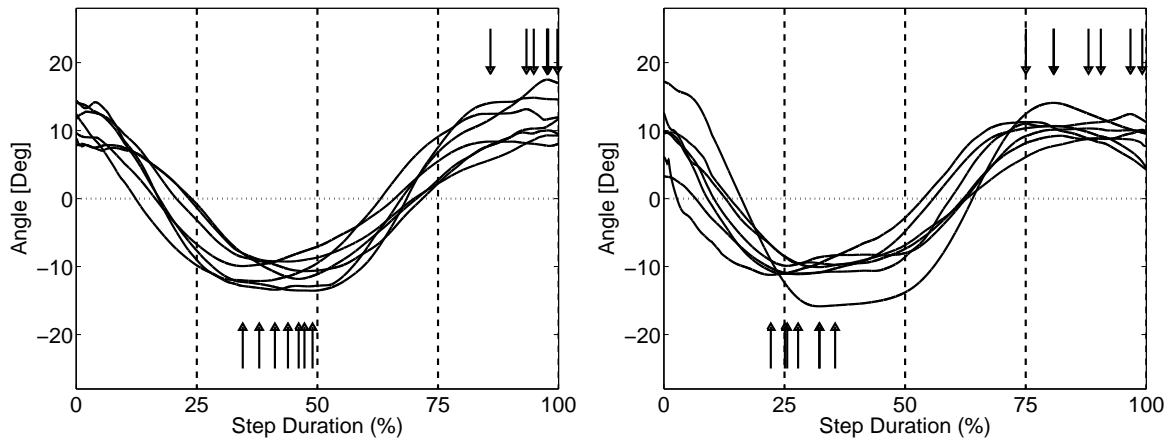


Figure 3.12: Median movement of the spine for each of the seven infants. We show 2 figures, for comparison with the movement of the limbs of the infants. For the left figure, the first vertical bar indicates the end of the swing of the left arm, the second one the start of swing of the right arm and the last bar shows the end of the swing of the right arm. For the right figure, the first vertical bar indicates the end of the swing of the right leg, the second one the start of swing of the left leg and the last bar shows the end of the swing of the left leg. On both figures the arrows denote the maximum curvature of the spine.

the time after the swing phase of the legs (see vertical arrows in Figure 3.12).

### 3.3.3 Improvement of locomotion: effects of experience

In this section, we are interested in the parameters that change according to the experience of the infants. To do so, three crawling infants (A, Al. and E.) were recorded again between one and two months after the first experiment (see Table 3.1). We compare the results of these experiments. Then we also show that the relation between speed of locomotion and experience.

#### Changes in crawling after more experience

Figure 3.13 shows the median swing and stance durations with the corresponding error bars. For the second series of measurement, we notice that for infants A. and Al., the median cycle duration has been reduced. The most impressive change is the decrease of the duty factor that gets closer to 50%. However this change is not so clear for the 3rd infant. Such changes have an important impact on the speed of locomotion of the infants (which doubled for the first 2 infants), as is shown in Table 3.2, which support the results in the previous section on the relation between stance duration and speed of locomotion.



### 3.3. Results

---

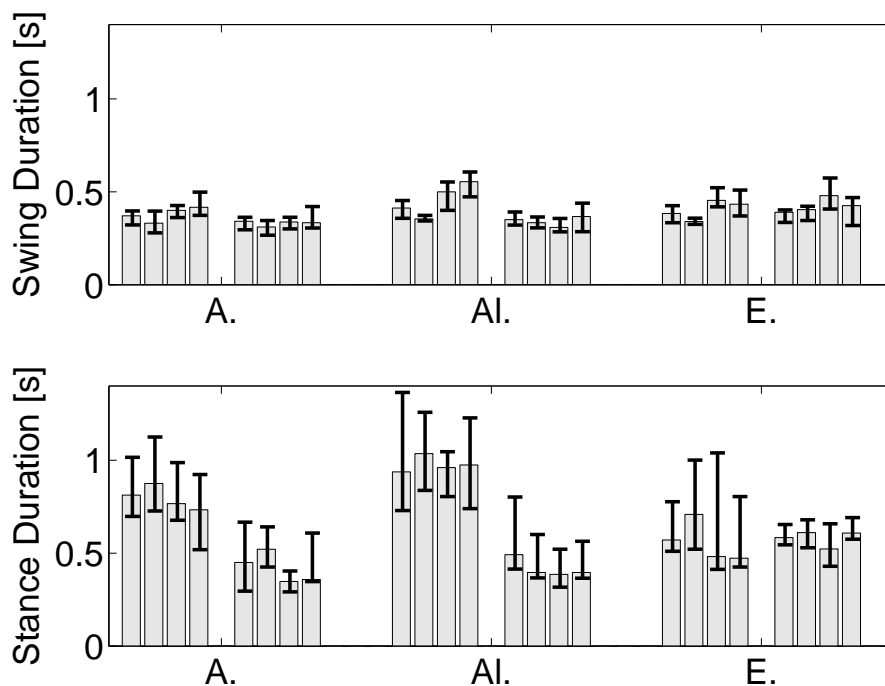


Figure 3.13: This figure shows the changes in swing duration (top figure) and stance duration (bottom figure) for the three infants that we measured twice. For each infant, the left set of bars corresponds to the first experiment, the right one corresponds to the second one. The data is represented as in Figure 3.6.

We tested the significance of the change in the median duration of the swing and stance phases after the second session of crawling (Table 3.2). For A., it turns out that the duration of the stance phase for all limbs has decreased significantly (duration decreases of about 50%). For Al., the duration of the stance phase of the right arm and both legs and the duration of the swing phase of the left leg have decreased significantly, but we notice that this change is bigger for the stance phases (60%). Finally for E., no change is significant, except for the duration of the swing phase of the right arm, which has increased by 18%.

From these experiments, it seems that the parameter that changes most is the duration of the stance phase of the limbs. The duration of the swing phase changes very little and as can be seen in Figure 3.13, its inter-quartile distance is also very small.

We also compared the limb kinematics of the infants for the 2nd experiment but we do not notice a significant change in the median kinematics of each infant (data not shown).

	Infant A.				Infant Al.				Infant E.			
	Swing		Stance		Swing		Stance		Swing		Stance	
Left Arm	0.144	-0.08%	<b>0.002</b>	<b>-0.45%</b>	<b>0.044</b>	<b>-0.15%</b>	0.136	-0.48%	0.914	0.02%	0.955	0.02%
Right Arm	0.319	-0.06%	<b>0.000</b>	<b>-0.40%</b>	0.135	-0.06%	<b>0.002</b>	<b>-0.62%</b>	<b>0.025</b>	<b>0.19%</b>	0.267	-0.14%
Left Leg	<b>0.003</b>	<b>-0.16%</b>	<b>0.019</b>	<b>-0.55%</b>	<b>0.000</b>	<b>-0.38%</b>	<b>0.003</b>	<b>-0.60%</b>	0.882	0.06%	0.852	0.09%
Right Leg	<b>0.042</b>	<b>-0.20%</b>	<b>0.006</b>	<b>-0.51%</b>	<b>0.000</b>	<b>-0.34%</b>	<b>0.006</b>	<b>-0.60%</b>	0.402	-0.02%	0.414	0.29%
Speed	0.16 / 0.35 m · s <sup>-1</sup>				0.23 / 0.43 m · s <sup>-1</sup>				0.24 / 0.27 m · s <sup>-1</sup>			

Table 3.2: Differences in median values for the duration of the swing and stance phases of each limb for the 3 infants after the 2nd experiment (p-value and percentage of change in the median value, a negative percentage means that the value has decreased of that percentage) and differences in speed of locomotion. The bold numbers represent p-values < 5% and their associated variation in the median value.

### Relation between locomotion speed and experience

In this section we show the relation between the number of days the infant first crawled before the experiment and the speed of locomotion. In Figure 3.14 we show the average speed of locomotion of an infant as a function of the number of days since he/she started crawling. We can see a relation between the experience and the speed of locomotion, we note that the correlation is high (0.71, p-value of 0.021), this shows that speed of locomotion increases with experience.

## 3.4 Discussion

### 3.4.1 Main gait parameters related to speed

We found that the standard crawling gait is between a walking trot and a lateral sequence diagonal couplets walk, with the legs starting swing shortly after the arms. The duty factor is between 50 and 70%. We also noticed that the stance duration varies considerably both for each infant and between the infants, this variation is indeed due to changes in speed of locomotion. The swing duration varies very little and we did not find any correlation with locomotion speed. The amplitude of movements of the shoulder and hip has also no correlation with speed. Therefore the main strategy to go faster is to vary the stance duration. These results have to be compared with the results about quadruped locomotion in other mammals. It is well known that as speed increases, stance duration decreases in a similar manner [56, 143]. It is also known that the duration of swing phase is relatively constant for all speeds [56, 87, 143]. Thus the main temporal characteristics of crawling locomotion are the same as for other quadruped vertebrates.

### 3.4. Discussion

---

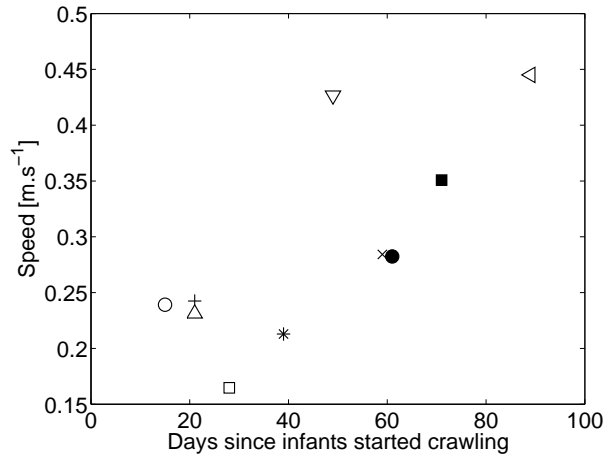


Figure 3.14: This figure shows the relation between the median speed of locomotion and the number of days since the infant started crawling. □ and ■ represent infant A. for experiment 1 and 2 respectively. △ and ▽ are for Al., ○ and ● for E., × for J., \* for M., ◁ for O. and + for V.

#### 3.4.2 Kinematics

From the kinematics point of view, we noticed that the general pattern of locomotion is qualitatively the same for all infants. We should however note that these infants were selected because they used the standard crawling gait (alternated locomotion on hands and knees); for infants having other gaits than the standard one, we may certainly find differences in the kinematics. The standard gait consists of a single period of extension (during stance) and flexion (during swing), with beginning of extension at the end of swing for the most proximal joints (shoulder and hip) as for all other mammals [43, 56, 143]. At touch down we notice the quite protracted arm posture (shoulder joint angle between 14 to 39° relative to the vertical) typical of primates [83], whereas other mammals have a more retracted posture (lower than 10°). The total excursion angle for the forelimbs is very low (median 46°) compared to other mammals of the same weight [84], where it is higher than 55°. It is especially true for primates where the excursion angle is very high.

In the case of the hind limbs, Larson et al. [84] reported data on the hindlimb excursion angles for different mammals (including primates). They show the femoral excursion angle at touch down, lift off and the total excursion angle (which is the sum of the absolute values of the touch down and lift off angles). They also show excursion data of the complete hindlimb (which is represented by the segment from the most proximal joint to the foot). Comparing our data with theirs is not easy since the femoral excursion in the case of crawling infants accounts for femoral and complete hindlimb excursion data since the knee touches the ground. In the data of Larson for every mammal, at touch down the total hindlimb angle is lower than 40° while the femur angle is higher. For

lift off the femoral angle is lower than  $-10^\circ$  for primates and carnivores while for all other mammals it is higher than  $0^\circ$ . The total hindlimb angle is always lower than  $15^\circ$ . Finally the total excursion angles are comparable for both the femur and the complete hindlimb. Compared to that study we see that the femoral excursion of crawling infants resembles the one of other primates but it is different from the total hindlimb excursion angle. When compared to other mammals of the same weight, infants have high hindlimb total excursion angle (median  $57^\circ$ ) that is similar to the excursion angles of primates of the same weight. The last observation is that the hindlimb total excursion angle is much higher than that of the forelimb for infants, which is an observation that we see only in a few primate species and the koala but in no other mammal.

The movement of the elbow joint consists of a single flexion and extension during swing and no movement during stance, mainly to allow the limb to move forward. The swing part corresponds well to the movement of the elbow joints of other mammals; however the stance part is different since we would expect also a flexion/extension of the limb during this phase as in many other mammals including primates [125]. The crawling gait is thus not compliant. This could be explained because infants have weakly developed forelimb muscles and a straight limb minimizes the effort on the joints (compliant gaits involve an increase in metabolic power [88] which implies an increased effort on the muscles of the elbow). During standard crawling, the forelimb consists of two functional segments and the hindlimb of only one functional segment, making the length of the forelimb much greater than the one of the hindlimbs, which is very different of other mammals. These differences could explain the quite protracted fore limb at touch down, since a more retracted limb would lead either to an unstable gait if the elbow did not yield or to the reduction of the visual field if the elbow yielded.

### 3.4.3 Lateral sequence footfalls

We know that the footfall sequence during quadruped walking of primates is generally a diagonal sequence gait (left-hind (Lh), right-front (Rf), right-hind (Rh), left-front (Lf)) while other non-primate quadrupeds use a lateral sequence gait (LhLrRhRf sequence) [143]. Despite some primates and especially their infants can also choose a lateral sequence gait [144], the diagonal sequence gait is the dominant one [130]. The crawling gait is thus closer to non-primate quadrupeds, since they choose only a lateral sequence gait as non-primate quadrupeds do. A tentative explanation would be that from a theoretical point of view, the primate pattern is less stable than the non-primate one. In [55] it is shown that the lateral sequence gait (non-primate) is the only pattern where the projection of the center of mass on the ground stays always in the support polygon of the quadruped when at least three feet are on the ground. When at least two limbs are on the ground, this pattern then minimizes the duration of phases where the projection of the center of mass is outside the support polygon and certainly increases static stability. This observation is valid only for slow gaits (statically stable) which seems a reasonable

### 3.4. Discussion

---

assumption for the walking trot of infants. However it seems that infants can only have walking gaits that are statically stable and no running gaits, for which they would need to be dynamically stable, which seems impossible due to their limited limb geometry. Since infants reach and manipulate for objects all the time and that the fore limbs are the only limbs in the visual field, this could also explain the preference of the fore limbs for the start of swing, i.e. the start of locomotion.

#### 3.4.4 Spine kinematics

We also noticed that the spine makes an undulation during locomotion, with maximum amplitude reached just after ipsilateral hind limb touch down. The observation of such undulations<sup>1</sup> has already been made for many tetrapods, from salamander [10, 45, 65], lizards [22, 117, 116] to primates (strepsirhines) [129]. For primates the maximum amplitude curvature of the spine is reached after ipsilateral hind limb touchdown and for the lizards it is reached before (except for very low speeds). As hypothesized by Shapiro et al. [129], the difference in the timing of the maximum curvature in lizards and primates could come from their respective gait. Lizards use a lateral sequence walking gait while primates use a diagonal sequence walking gait. However we found that infants have maximum curvature of the spine similar to primates but a different gait. A tentative explanation would be that these differences are seen because of the limb geometry of the crawling infants, having long fore limbs compared to hind limbs which again is different from primates. If the spine movements and gait of infants were closer to slowly walking lizards than primates, an increase in their speed would lead to a maximum curvature happening before the end of swing. More experiments both on human infants and non-human primate infants using sometimes a lateral sequence gait would be needed to further compare gaits and spine movements and explore further explanations.

#### 3.4.5 Development of the gait through experience

It seems that, apart from stability, the main parameter that is changed in crawling after some experience is the duration of stance phase, the swing phase staying almost constant. The crawling speed being related to experience of the crawler, we can say that the crawling gait of an infant is tending to a faster gait where a complete support phase (where the four limbs are on the ground) is as small as possible. It means that infants begin with a gait in between a walk and a walking trot and tend to a perfect walking trot with more experience. Interestingly we can find similarities with the development of the gaits of other quadrupeds. In several rodent species lateral walking appears first closely followed by trotting, more specialized gaits (asymmetric or biped gaits) develop only later [38]. It

---

<sup>1</sup>We do not have enough angles along the spine to show that the undulation is actually a standing wave but we have the intuition that it is one, as we observe in walking salamanders.

is interesting to note the similarities of the ontogeny even if more experiments would be needed to be able to affirm that they are the same.

### 3.4.6 Conclusion

Finally, despite the fact that humans crawl for a short time period of their life, can adopt many different strategies for locomotion (instead of the standard crawling gait) and have a mechanical structure that is not optimized for quadrupedalism, we found that the infant standard crawling gait shares most of the basic principles of other vertebrate quadruped gaits. They are quite close to other primates in term of joint excursion pattern, although the elbow is not compliant, but are closer to other mammals in terms of the lateral sequence gait. It seems that the gait characteristics are not so much determined by the detailed limb geometry of the quadruped, since very different functional geometries (infants versus other quadruped vertebrates) lead to very similar temporal characteristics and kinematics. Indeed it seems that stable quadrupedalism implies some gait properties that are independent of the specific mechanics. Thus this may emphasize the role of the neural control in the determination of these characteristics, the mechanics being important for optimizing the control.

## 3.5 Outlook

The results from this chapter show that crawling of human infants is not a special case of quadruped locomotion and that many similarities exist between the gaits of infants and those of other quadruped mammals. In the next chapter, we present our design methodology for CPG based locomotion controllers of quadruped robots. The development of this methodology was inspired by some results from this chapter, namely the trot gait for the iCub robot and the importance to separate the swing and stance durations in our oscillators. We postulate that as for mammals, stance duration in robots can be used to control speed of locomotion while swing duration is kept to a fixed value for stability reasons. We will see in numerical simulations in the next chapter that this assumption is indeed very useful.

---

## CHAPTER 4

# DESIGN OF CENTRAL PATTERN GENERATORS

---

One crucial question for the control of locomotion that is not solved yet in a satisfactory manner is how to generate control policies that can adapt to a changing environment? As we discussed in Chapter 2, several approaches can be taken to solve this problem. On one side model-based approaches optimize control trajectories offline, under several constraints (for example using the ZMP criterion [147]), such that the robot will have a stable locomotion under certain conditions. The problem is then to adapt these trajectories online to unmodeled perturbations, the algorithms that calculate the trajectories offline are usually computationally expensive and therefore not suitable for running in real time. There are many attempts to design online algorithms, for example using model predictive control approaches like [35]. Although the computational problem can be simplified to a great extent, it is still difficult to run online and requires a precise model of the robot. On the other side, researchers have come up with new approaches that do not use explicit models, often inspired by the way Nature solves the problem. One promising approach is based on Central Pattern Generators (CPGs) to generate control policies for robots.

As we have seen previously, a CPG in biology is a distributed neural network located in the spine of animals, that is able to generate automatically the rhythmic patterns of control to perform periodic movements (locomotion for example) [57]. These neural networks are modulated by low dimensional signals from higher parts of the brain (e.g. the brainstem) and produce high dimensional patterns that control the muscle synergies during locomotion. They are also strongly coupled to sensory information [44], the effect of feedback being phase dependent during locomotion (i.e. the same sensory inputs produce different effects according to the state of the animal).

In robotics, as well as in theoretical biology, CPGs are often modeled as coupled dynamical systems, mostly oscillators [26, 52, 65, 80]. The advantage of such controllers is their stability properties (limit cycle behavior), where transient perturbations are rapidly

forgotten. We talk about limit cycle behavior in the sense that coupled oscillators produce structurally stable limit cycles and the controller is then well suited for feedback integration. But we also talk about limit cycles in the sense that the CPG coupled to the robot it controls via sensory feedback produces an overall limit cycle (i.e. the robot is stable). Although we can observe this behavior in CPG based controllers, it is always difficult to prove such a property. Another advantage of using oscillators is their synchronization properties that allow easy coordination between the different degrees of freedom of a robot and entrainment between the robot dynamics and the controller. Controllers based on coupled oscillators allow also to reduce the dimensionality of the control problem, since only simple parameters as frequency, amplitude and phase differences between the oscillators need to be chosen to generate high dimensional control policies. Moreover these systems does not only produce a fixed trajectory but encode a whole state space (i.e. ways to come back to this trajectory) and it is easy to smoothly modulate the generated patterns with simple parameter changes. The resulting controller, made of simple differential equations is also computationally inexpensive. And it is easy to robustly distribute the control on parallel micro controllers at a low level, which makes the CPG approach well suited for the control of autonomous robots. Finally, this method is model free and is thus interesting to control robots in unknown environments. However, it is not always an advantage, indeed, since we do not optimize an explicit criterion as a ZMP, we cannot explicitly prove the stability of the robots. The CPG approach has been successfully applied to the control of several robots, from snake and salamander like robots [65, 68] to quadruped robots [15, 46, 75, 77, 76] and biped robots [7, 41, 93, 95].

The problem with CPG controllers is that there exists almost no methodologies to design such systems. Despite some attempts to provide systematic tools to build them, either based on learning [40, 63, 72] or engineering design [15], most of the time they are tailor-made for a specific application. Very few design principles exist, especially for the integration of sensory feedback.

In this chapter we present our work on quadruped locomotion and more specifically our attempt to provide a design methodology for CPGs. The methodology is based on three distinct steps: first we choose the basic unit of the CPG, the oscillator that has suitable properties for the given task, second we design the coupling architecture of the CPG, independently from the previous choice of oscillators, and finally we integrate sensory feedback at the oscillator's level.

This chapter is composed of two parts that are related to three original contributions [111, 113, 114]. The next section presents a preliminary work on the control of the crawling iCub based on early data sets on human infants crawling. It already introduces the main ideas of the CPG design: a locomotion specific oscillator and the design of the coupling architecture for the CPG. However the control is purely open-loop at that stage. The second section extends these ideas and provides a generic controller for quadruped robots. The central aspect of this section is the introduction of local sensory feedback in the CPG. We show that the same controller can be used on three different quadruped



## 4.1. Open-loop CPG design

---

robots.

### 4.1 Open-loop CPG design

In this section, we use recent results from dynamical systems theory to create a network of oscillators for the control of crawling of the iCub robot. This is a preliminary work on the subject, especially the design of the oscillator is not the final one and the coupling between the oscillators is very application specific. However this contribution is interesting in the sense that it shows the initial ideas on which is built the rest of the chapter and that these ideas are not limited to a specific choice of implementation (oscillator choice and coupling). We must note that the detailed coupling architecture (i.e. the inhibiting coupling) is based on preliminary data from a crawling infant. However this observation was specific to this infant and was not observed generally in the other infants we studied (see Chapter 3 for details).

The material in the remainder of the section is taken from a conference paper that we published in [111].

#### 4.1.1 Motivations

Our design methodology follows a biologically inspired approach. Indeed, to design the controller we study the crawling behavior of infants in order to extract important principles. Then we present a mathematical model of CPG based on coupled nonlinear oscillators to reproduce the crawling gait of infants. The originality of the model resides both in the design of the oscillator and in the design of the coupling scheme of the CPG. Note that while we design a CPG for a specific task, we develop tools that are general enough to be used in other applications.

We designed our oscillator from the observation that the gait pattern of animals and humans can be separated into two distinct phases for each limb. The stance phase is the phase during which the limb touches the ground. The swing phase is the phase during which the limb lifts off the ground. It is a well-known fact that when quadrupeds change their speed of locomotion, they might change their gait and the duration of the stance phase, but the duration of the swing phase tends to remain the same (see for example [87] and Chapter 3). However, most of the CPG models based on coupled oscillators are not able to separate the swing and stance phases durations. In this section, we present an oscillator model in which we can independently control the duration of each of these phases. This is an important feature, since during the swing phase, one limb is off the ground, thus making the system less stable and more dependent on dynamical properties of the controlled robot. So it seems important to control independently the duration of both phases.

Furthermore, we present a coupling scheme based on the analysis of the crawling

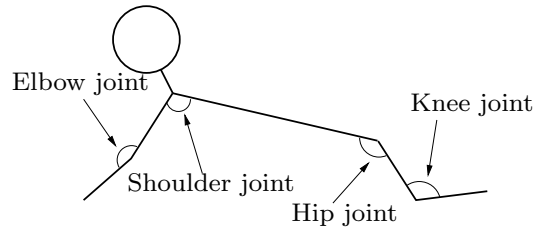


Figure 4.1: Schematic of the joint angle we measured. We look at the movement of the limbs in the sagittal plane.

pattern of a real infant to reproduce a similar gait. This coupling is based on the fact that the infants have an almost trot-like gait for the temporal synchronization of the limbs but with a high duty factor ( $> 50\%$ ). Moreover, it appears that there exists a correlation between the movement of a limb during its stance phase and the swing phase of the opposite limb<sup>1</sup>. We reproduce this influence in the coupling scheme we present and we use the theory of symmetric dynamical systems [21, 50, 51] to infer the architecture of the network of coupled oscillators. The validation of the design is done by testing the CPG controller with a physics simulation of the iCub, since the real robot is still under construction.

In the following, we first quickly recall important data on crawling infants from the robotics perspective (Section 4.1.2). We then present the design approach behind our model of coupled oscillators (Section 4.1.3). The design is done incrementally with first the construction of a nonlinear oscillator with two controlled time scales, then the addition of inter-limb influences between oscillators of opposite limbs, and finally the addition of inter-limb couplings between the complete four-oscillator system for implementing the crawling gait. The model is tested with a rigid articulated body simulation of the iCub, and compared to the original infant crawling gait (Section 4.1.4). We finish this section by a short discussion (Section 4.1.5).

### 4.1.2 Crawling in Infants from the robotics perspective

As we already discussed in Chapter 3 infants can have various types of crawling gaits and in this section we only focus on the *standard* gait. The first general remark we use for our CPG design is that standard crawling resembles a trot-like gait for its temporal relations between limbs. It means that the diagonal limbs (e.g. left arm and right leg) are in phase and half a period out of phase with the opposite limbs. Moreover this trot-like gait has a high duty factor (between 50 and 70%).

Second the speed of locomotion is controlled by the stance duration while the swing

---

<sup>1</sup>Note that this observation was made in preliminary experiments with one infant and could not be generalized to other infants, see Chapter 3 for more informations.

## 4.1. Open-loop CPG design

---

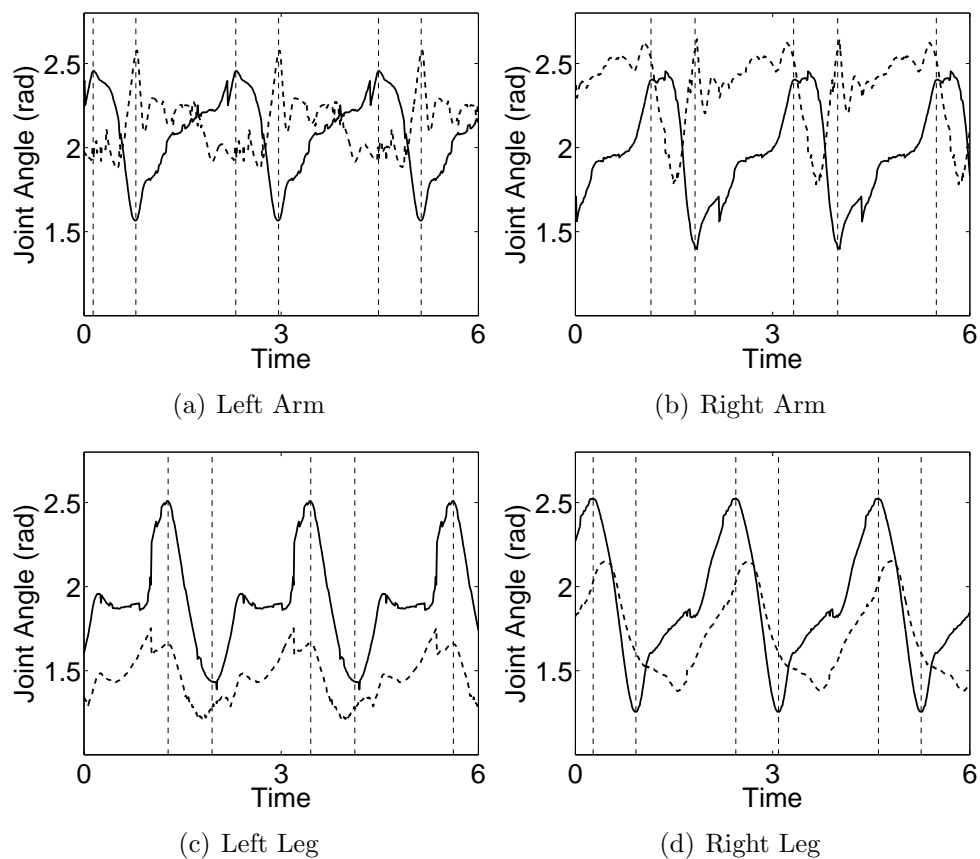


Figure 4.2: Evolution of the joint angles of an infant during crawling. This is a reconstruction of a crawling sequence from the recordings of a crawling infant. We plot the joint angles (in radian) of the 4 limbs. For each limb, we plot the joint angles as defined in Figure 4.1. Hip and shoulder joints are plotted in plain line, the knee and elbow joints are in dashed line. The vertical lines delimit the swing and stance phases, the swing phase being the shortest one.

duration is almost constant (as for other quadruped mammals [87, 100]).

When looking at the movement of the hip and shoulder joints for the infant considered here (see Figure 4.2), we notice that during the stance phase the joint slows down or even sometimes stops during the swing phase of the opposite limb. It is as if the swing phase of a limb was inhibiting the movement of the opposite limb.

The knee joint of each limb is folding in order to follow the movement of the hip. Since the infant is crawling on its knees, the exact control of this joint is less important (i.e. the tibia tends to simply rest on the ground). The elbow joints are folding during the swing phase, to allow the arm to reach a further region in front of the infant but do not move significantly during the stance phase.

Our goal here is to extract the features of crawling that seem important in order to reproduce the same gait in a robot. The main features we would like to emphasize from these observations are first that the crawling gait is a trot-like gait in terms of phase relations between the limbs but with a stance phase that is much longer than the swing phase contrary to usual trot gaits (i.e. it is a walking trot). Second, there is a correlation between the swing phase of a limb and the arrest of movement of the hip (or shoulder) joint of the opposite limb. Third, the elbow is folding to allow the arm to do the swing phase. Fourth, from the study of Chapter 3 the stance duration controls the speed of locomotion while swing duration is kept almost constant.

### 4.1.3 CPG model

In this section we construct a model of CPG by means of coupled oscillators. The CPG will be used to generate the crawling trajectories for the iCub humanoid robot. To construct a CPG model, we define a number of features we would like our model to have. First the CPG should comply with the previous observations on the crawling infant. Then from a robotics point of view, the CPG must have properties that makes it suitable for the control of a real robot. We therefore want the CPG to show limit cycle behavior and to be stable against perturbations, to allow for further integration of sensory feedback. We also want to be able to smoothly modulate the generated trajectory in frequency and in amplitude to have a larger range of possible locomotion.

In summary our CPG must have the following properties

- Smooth modulation of the generated trajectory in frequency and amplitude
- Independent control of the duration of the swing and stance phases (the ascending and descending phases of the oscillations)
- Trot-like gait with a stance phase much longer than the swing phase
- Inhibition of the movement of the hip and shoulder joints during the swing phase of the opposite limbs

## 4.1. Open-loop CPG design

---

- Stability to perturbations to allow feedback integration

### Two time-scale oscillator

We first present a model of a stable oscillator with the possibility to control independently the duration of the swing and stance phases and the amplitude of the oscillations.

If we take a simple spring-like oscillatory system, the equation of motion of the joint angle can be expressed as

$$\dot{x} = y \quad (4.1)$$

$$\dot{y} = -Kx \quad (4.2)$$

The frequency of oscillations will be  $\sqrt{K}$  and we will have harmonic oscillations whose amplitude will depend on the initial conditions of the system.

We want a duration of the stance phase different from the duration of the swing phase, thus we can think of an oscillator changing its spring constant according to the phase. It will have a  $k_{stance}$  spring constant during stance phase and  $k_{swing}$  constant during swing phase. The oscillator will switch among these two constants according to the phase, that is according to the sign of the velocity  $y$  of the system. We can thus write a general spring constant as

$$K = k_{stance} + (k_{swing} - k_{stance}) \frac{1}{e^{by} + 1} \quad (4.3)$$

where the exponential function works as a step function which selects either  $k_{swing}$  or  $k_{stance}$  according to the sign of the velocity of movement  $y$ ,  $b$  controls the speed of the switch.

Now we have a system that oscillates with different speeds according to the direction of the oscillations. Therefore we can independently control the duration of the swing and the stance phases.

The problem with such an oscillatory system is that no limit cycle exists. There exist infinitely many periodic orbits around the unstable center 0 and thus the system is not stable. We can point the flow toward one periodic orbit by constraining the total energy of the system, since it defines the maximum value  $x$  can take in a spring system. The total energy of the system is defined by

$$E = \frac{1}{2}(Kx^2 + y^2) \quad (4.4)$$

which is the sum of the potential and kinetic energies of the system (we take the mass equal to one). At  $y = 0$  we have  $E = \frac{1}{2}Kx^2$ , which gives  $x_{max} = \pm\sqrt{\frac{2E}{K}}$ . We can choose a total energy such that  $x_{max}$  is bounded to a certain value,  $E = \frac{\mu^2 K}{2}$  and  $x_{max} = \pm\mu$ .

In order to constraint the amplitude of oscillations, we add a damping term to the preceding equation which bounds the total energy of the system. We then rewrite the whole system as

$$\dot{x} = y \tag{4.5}$$

$$\dot{y} = \alpha y(\mu^2 K - (Kx^2 + y^2)) - Kx \tag{4.6}$$

where  $\alpha$  is a constant controlling the speed of convergence of the energy of the system  $\frac{1}{2}(Kx^2 + y^2)$  to the wanted total energy  $\frac{1}{2}\mu^2 K$ .

The stability of the system can be seen if we set  $E = \frac{1}{2}(Kx^2 + y^2)$ , then differentiation with respect to time gives

$$\dot{E} = \frac{1}{2}\dot{K}x^2 + \alpha y^2(K\mu^2 - E) \tag{4.7}$$

In our application,  $K$  can be approximated as a switching function, whose value equals either  $k_{swing}$  when  $y < 0$  or  $k_{stance}$  when  $y > 0$ . Thus, for  $y \neq 0$ , we see that the flow is always directed toward  $E = K\mu^2$ . When  $y = 0$ ,  $K$  changes its value from a spring constant to the other and we see that  $E$  changes also its value in the direction of this change because of the  $\dot{K}$  term. Thus the flow is always directed toward  $K\mu^2$  and the system is stable. The stable limit cycle has then equation  $Kx^2 + y^2 = K\mu^2$ . It is composed of two half-ellipses that share the same semi-minor axis  $\mu$  (so they are connected) and with foci at  $y = \pm\mu\sqrt{k_{swing} - 1}$  and  $y = \pm\mu\sqrt{k_{stance} - 1}$  respectively.

Now we have an oscillator bounded in energy for which we can independently control the duration of the swing and the stance phases. Moreover, with the bounded energy, we assure that the oscillator is stable and that we can control the amplitude of the oscillations which are equal to  $\mu$ . Figure 4.3 shows an example of oscillations with different values for the stance and the swing phases.

### Inhibitory coupling

In this section we describe the inhibitory coupling scheme we use to replicate the slow down of the hip and shoulder joints during the swing phase of the opposite limbs (see Figure 4.2). We introduce an inhibitory coupling that sets the stance spring constant  $k_{stance}$  to 0 when the opposite limb starts its swing phase, i.e. when the speed of the oscillator becomes negative

$$K_i = \frac{k_{stance}}{(e^{-by_i} + 1)(e^{-ky_j} + 1)} + \frac{k_{swing}}{e^{by_i} + 1} \tag{4.8}$$

where  $i$  denotes the oscillator that is inhibited and  $j$  the opposite oscillator,  $k$  controls the speed of slow down of the oscillator. With this coupling scheme, when one limb starts its swing phase, the opposite limb will slow down its movement. The amount of

## 4.1. Open-loop CPG design

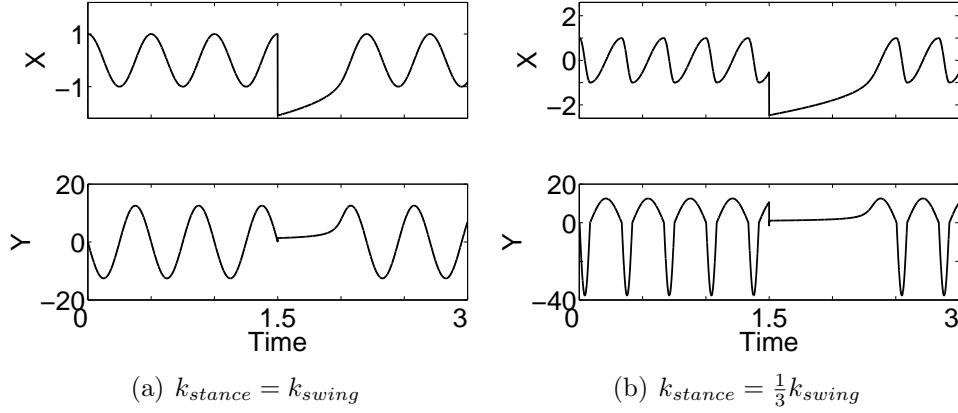


Figure 4.3: This figure shows how we can independently control the ascending and descending durations of the oscillator, in 4.3(a) we plot the oscillations when  $k_{stance} = k_{swing} = 4.(2\pi)^2$ , in 4.3(b) we plot  $k_{stance} = \frac{1}{3}k_{swing} = 4.(2\pi)^2$ . In each plot we show the oscillations  $x$  and the velocity  $y$ . At time  $t = 1.5$ , we perturb the system by setting  $x$  and  $y$  to a random value, we clearly see that the oscillations are stable.

deceleration will depend on the kinetic energy the oscillator has at this moment and the energy damping term  $\alpha$ .

To assure that this slow down will be fast enough when  $x \simeq 0$ , we change the damping term so it has a very high value when  $x \simeq 0$  and a smaller value otherwise. To do this, we transform  $\alpha$  into a Gaussian function centered around 0.

$$\alpha_i = \nu(1 + \beta e^{-\sigma x_i^2}) \quad (4.9)$$

where  $\nu$  is the damping constant,  $\beta$  controls the change of the damping around 0 and  $\sigma$  controls the width of the Gaussian. With  $\alpha_i$  we can now independently control the general damping term that constrains the total energy of the system and the damping when  $x \simeq 0$ , i.e. during the inhibition.

### Architecture of the CPG

In addition to the coupling scheme for inhibition, we have to introduce a coupling to maintain the phase relations between each limb. We want a half a period out of phase relation between opposite limbs (e.g. between the arms) and an in-phase relation between diagonal limbs (e.g. right arm and left leg).

To design such a network, we use the theory of symmetric coupled cell networks [21, 50, 51]<sup>2</sup>. By looking only at the symmetries of a network of coupled oscillators,

<sup>2</sup>Here we introduce the basic idea of CPG design using the theory of symmetric coupled cell networks. We will discuss in greater details this method in Section 4.2.

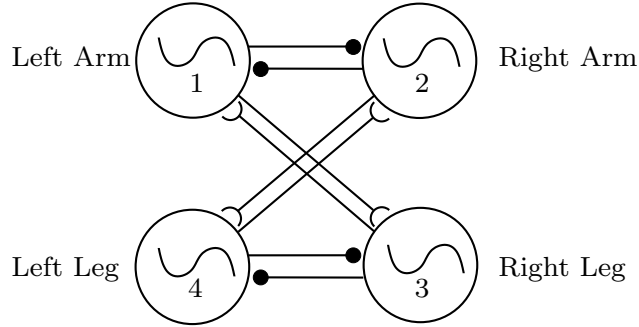


Figure 4.4: The architecture of the CPG

K	Possible Periodic Solutions	Stability
$\Gamma$	$x_1(t) = x_2(t) = x_3(t) = x_4(t)$	Unstable
$\{I, (12)(34)\}$	$x_1(t) = x_2(t) = x_3(t + \frac{T}{2}) = x_4(t + \frac{T}{2})$	Unstable
$\{I, (13)(24)\}$	$x_1(t) = x_2(t + \frac{T}{2}) = x_3(t) = x_4(t + \frac{T}{2})$	Asym. stable
$\{I, (14)(23)\}$	$x_1(t) = x_2(t + \frac{T}{2}) = x_3(t + \frac{T}{2}) = x_4(t)$	Unstable

Figure 4.5: From the symmetry of the network, we derived the possible pattern of synchronization according to the possible subgroups of spatial symmetry. For each subgroup, we indicate the possible periodic solutions and their stability. The stability of the solutions was evaluated numerically, as shown in Figure 4.6.



## 4.1. Open-loop CPG design

---

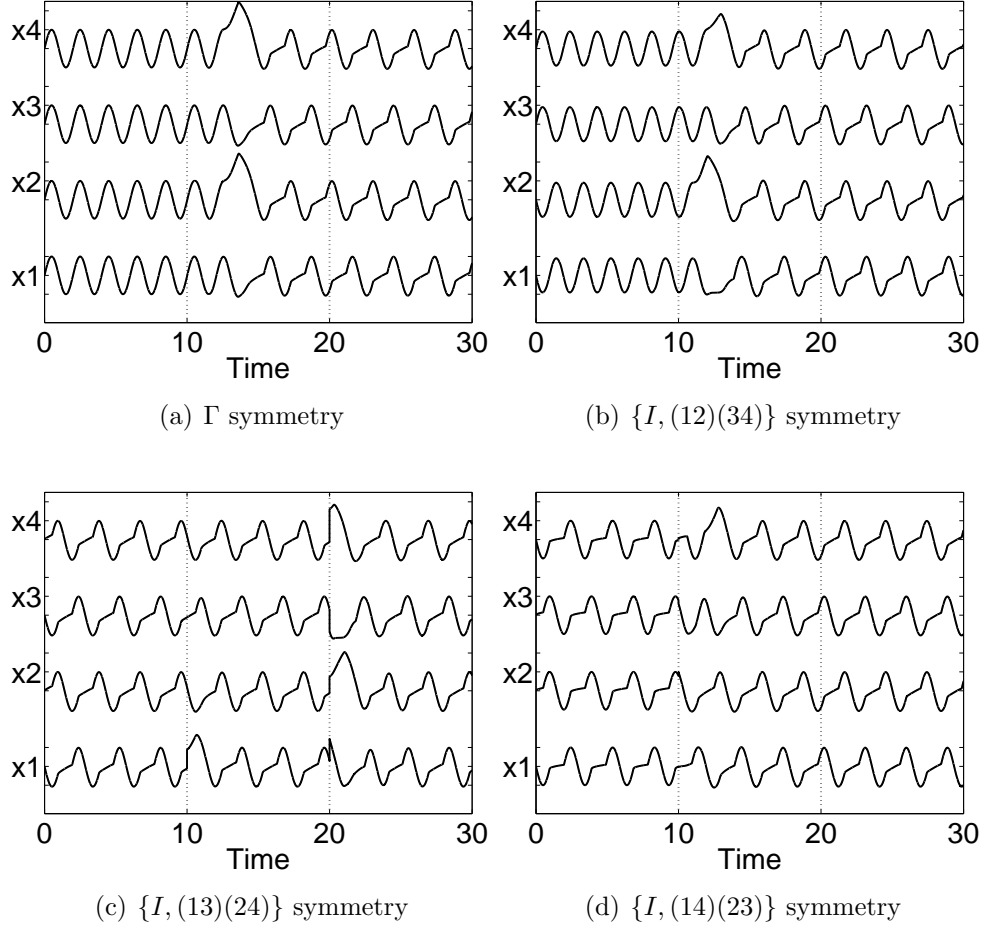


Figure 4.6: We show the 4 possible patterns of synchrony we predicted from the symmetries of the network. We also show their stability properties by perturbing the oscillators. For patterns of Figs. 4.6(a) and 4.6(b) at time  $t = 10s$  we add a perturbation of 0.01 to  $x_1$ , we see that such a small perturbation completely destroys the patterns and the crawling pattern appears. For the pattern of Figure 4.6(c), which is the crawling pattern, at time  $t = 10$  we add 1.0 to  $x_1$  and at time  $t = 20$  we set the state variables of each oscillator at a random value between  $[-2, 2]$ , it is clear that this pattern is stable. The pattern in Figure 4.6(d) is a pace gait, at time  $t = 10s$  we add a random noise between  $[-0.2, 0.2]$  on each  $x_i$ . For all the experiment, we set  $k_{swing} = k_{stance} = \pi^2$ ,  $c1 = c2 = 1.0$ ,  $\beta = 100$ ,  $\sigma = 10$ ,  $\nu = 0.45$ ,  $b = k = 100$  and  $\mu = 1$ .

we can deduce the existence of stable solutions having the same symmetries. These symmetries are defined as the group of permutations of the cells of the network which preserve its architecture.

Of course, the symmetries of the network induce that the corresponding ordinary differential equations (ODEs) describing the network have the same symmetry. In this case we can distinguish two kinds of symmetries. The spatial symmetries of a certain set of ODEs which are the symmetries  $\gamma$  such that for any solution  $x(t)$  of the set of ODEs  $\gamma x(t) = x(t)$ . The spatio-temporal symmetries are the symmetries  $\varphi$  which preserve the orbit of a solution, which means that if  $x(t)$  is a solution with orbit  $\{x(t)\}$ , then  $\varphi x(t)$  has the same orbit. In other word, if  $x(t)$  is a periodic solution, then  $\varphi x(t)$  will be the same solution with some phase shift.

For the crawling gait, if we number the limbs as in Figure 4.4, we want the permutation of the diagonal limbs (13)(24) to be a spatial symmetry and  $((12)(34), \frac{1}{2})$  and  $((14)(23), \frac{1}{2})$  to be spatio-temporal symmetries with half a period phase shift.

We can construct a coupled cell network that is symmetric under the group generated by these symmetries. By the H/K theorem, we know that the crawling gait is a periodic solution of any network having the same symmetries.

**Theorem 1** H/K Theorem[51] *Let  $\Gamma$  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.*

In our case, we have

$$\Gamma = H = \{I, ((13)(24), 0), ((12)(34), \frac{1}{2}), ((14)(23), \frac{1}{2})\}$$

and

$$K = \{I, ((13)(24), 0)\}$$

We clearly see that  $H/K \cong \mathbb{Z}_2$  is cyclic and thus the trot-like gait exists as a solution of the system as long as we choose a coupling scheme such that  $K$  is an isotropy subgroup (which is easy).

We just have to choose a coupling such that the trot gait is stable, but we already know that it is possible. Since we have inhibitory coupling between opposite limbs, we add standard subtractive coupling between these oscillators in order to enforce the half a period phase shift. As we also want in-phase relations between diagonal limbs, we also add additive coupling between opposite limbs. These couplings are well studied and we know that they make the desired phase shifts between 2 oscillators stable [103]. Figure 4.4 shows the architecture of the network with the coupling scheme.

## 4.1. Open-loop CPG design

---

The general equations of the CPG that generates the trajectories for the hip and shoulder joints are then

$$\dot{x}_i = y_i \quad (4.10)$$

$$\dot{y}_i = \alpha_i y_i (K_i (\mu^2 - x_i^2) - y_i^2) - K_i x_i - c_1 y_j + c_2 y_k \quad (4.11)$$

$$K_i = \frac{k_{stance}}{(e^{-by_i} + 1)(e^{-ky_j} + 1)} + \frac{k_{swing}}{e^{by_i} + 1} \quad (4.12)$$

$$\alpha_i = \nu(1 + \beta e^{-\sigma x_i^2}) \quad (4.13)$$

where  $i = 1..4$  denotes the  $i$ th oscillator,  $j$  the opposite oscillator and  $k$  the diagonal oscillator,  $c_1$  and  $c_2$  are positive coupling constants. We can verify that  $K$  is an isotropy subgroup for this set of equations.

Another advantage of this method to design the architecture of the CPG is that we can directly calculate the existence of other patterns of oscillations by simply calculating the other subgroups of  $\Gamma$  as is shown in Figure 4.5. It is very important to be able to calculate the possible patterns of oscillations and to investigate their stability properties in order to be able to guarantee the behavior of our controller when adding feedback loops. This method transforms the analytic problems of finding these modes of oscillations into an algebraic one, which is easier.

We see that there exist three other oscillatory regimes and we evaluated numerically the stability of each of these patterns, as can be seen in Figure 4.6. We note that the only stable pattern of oscillation with a wide basin of attraction is the trot gait. The pace gait has a small region of stability that is limited and for a random noise between  $[-0.2, 0.2]$ , this pattern disappears. The two other patterns are unstable. After the perturbations, all these patterns converge to the trot gait.

Now we have a CPG that can generate the trajectories for the hip and shoulder angles. This CPG is stable against perturbations. We can also smoothly modulate the frequency of the pattern by changing independently the frequency of the ascending and descending oscillations. A smooth modulation of the amplitude is straightforward by changing the parameter  $\mu$ . Examples of such modulations can be seen in Figure 4.7.

### 4.1.4 Validation of the model

#### Comparison with the real infant

In this section we compare the trajectories of the shoulder and hip joints of the infant considered here with the ones generated by the CPG. In Figure 4.8 we see the result of the comparison. The theoretical trajectories match quite well the experimental ones. This result shows that the CPG can reproduce the main features of crawling and therefore it supports our design methodology.

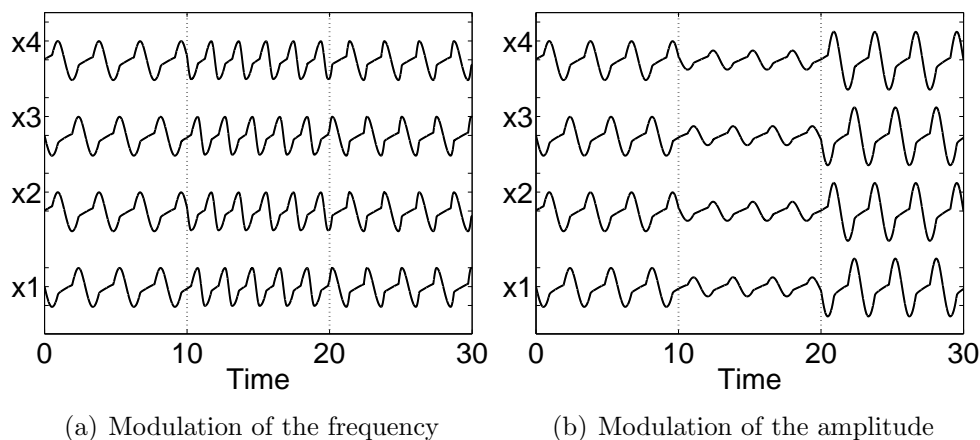


Figure 4.7: In Figure 4.7(a) we modulate the frequency of the CPG. Initially we have  $k_{swing} = k_{stance} = \pi^2$ , at  $t = 10$  we set  $k_{swing} = 4k_{stance} = 4\pi^2$ , which corresponds to a doubling of the speed of the swing and at  $t = 20$  we set  $k_{stance} = 4k_{swing} = 4\pi^2$ . In Figure 4.7(b) we modulate the amplitude of the pattern, we set  $\mu = 1$  at  $t = 0$ , then  $\mu = 0.5$  at  $t = 10$  and  $\mu = 1.5$  at  $t = 20$ . Note that an abrupt change in the control parameters ( $k_i$ ,  $\mu$ ) leads to a smooth transition in the generated pattern.

### Crawling on the simulated robot

In this section we show experiments where we use our CPG to control a crawling simulated robot. The simulation is done with Webots [90], a simulator based on ODE [2], an open source physics engine for simulating 3D rigid body dynamics. The simulation is as close as possible to the robot currently under construction. This means that we use the correct lengths and mass distributions for each limbs.

The CPG we developed generates the trajectories for the hip and shoulder joints, so we use these trajectories to control the position of the hip and shoulder joints. However, we saw in Section 4.1.2 that the elbow was also used during the swing phase of the corresponding arm. The elbow is folding during the swing phase, allowing the arm to reach the region in front of the infant.

We thus set the angle of the elbow joint according to the phase of the arm, that is, according to the sign of  $y$ . The angle of the elbow,  $\theta_i$ , will follow a Gaussian movement corresponding to

$$\theta_i = \gamma e^{-\frac{(y_i - \sqrt{k_{swing}})^2}{\tau}} \quad (4.14)$$

where  $i$  corresponds to the left or right arm,  $\gamma$  is the amplitude of the movement and  $\tau$  is the width of the Gaussian. The Gaussian is centered on  $-\sqrt{k_{swing}}$  which corresponds to the maximum speed of the shoulder during the swing phase. The oscillator reaches this value at  $x \simeq 0$ . The elbow will then fold during the swing phase, following a Gaussian movement and will not move during the stance phase. We also control the DOF of the

## 4.1. Open-loop CPG design

---

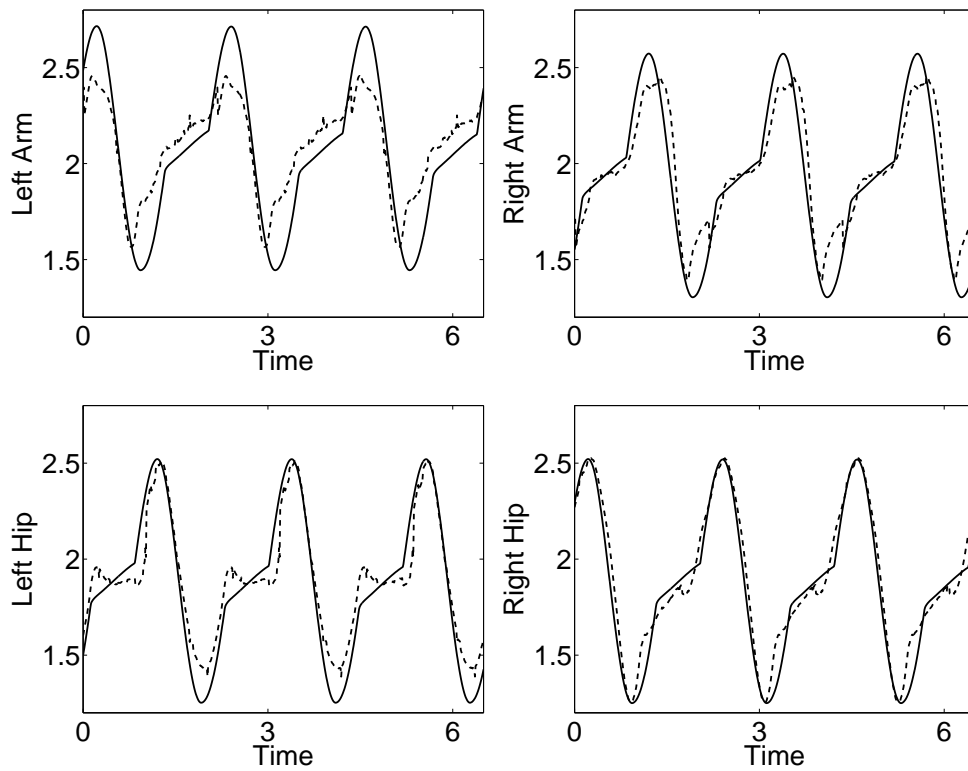


Figure 4.8: Comparison of the real trajectories of the hip and shoulder joints and the trajectories generated by the CPG. The trajectories of the CPG are only shifted to oscillate around the same mean values as the real trajectories. We see that the trajectories generated by our model fit quite well the real ones, especially for the right limbs.

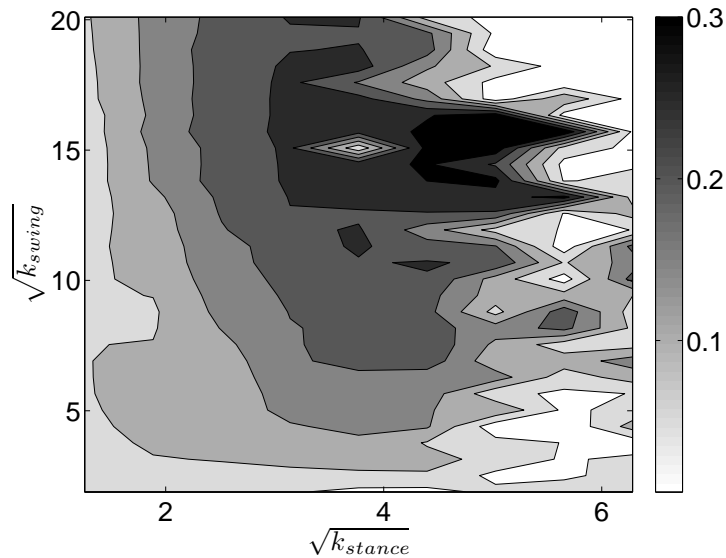


Figure 4.9: Characterization of the speed of the simulated iCub according to the duration of the stance and swing phases. The left color bar shows the correspondence between the colors and the speed (in  $\text{m} \cdot \text{s}^{-1}$ ).

arm, which is orthogonal to the sagittal plane, in the same way. This allows the hand to have more height during the swing phase. Figure 4.10 shows a crawling sequence of the simulated robot and of a real infant.

We also investigated the importance of the  $k_{swing}$  and  $k_{stance}$  constants for the speed of locomotion of the robot. The result of this experiment can be seen in Figure 4.9. We see in this figure that the simulated robot could crawl up to  $0.3 \text{ m} \cdot \text{s}^{-1}$ . A infant crawls around  $0.18 \text{ m} \cdot \text{s}^{-1}$  [96], so the robot can attain the speed of a real infant. In this figure, we see that increasing  $k_{stance}$  increases the speed of the robot. However, we see that beyond a certain value the speed of locomotion becomes really small, which corresponds to cases where the robot falls. We also note that increasing  $k_{swing}$  for a given  $k_{stance}$  does not lead to a significantly faster crawling. This seems normal since the swing phase is not the longer part of the movement. However the value of  $k_{swing}$  is important for the stability of locomotion, a too slow swing phase leads to cases where the robot falls or the swinging arm touches the ground too early, in the middle of the swing phase.

#### 4.1.5 Conclusion

In this section, we presented an oscillator in which we can independently control the duration of the ascendant and descendant phases. In locomotion control it enables us to set the duration of the swing and the stance phases separately.

We also presented an original way of coupling two oscillators, in order to reproduce the

## 4.2. Inclusion of sensory feedback

---

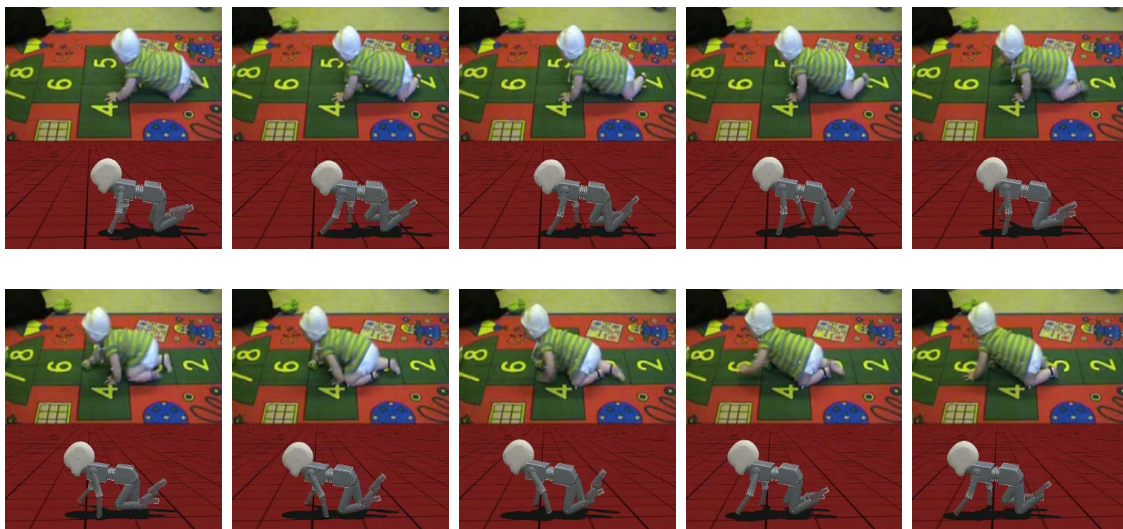


Figure 4.10: We show a sequence of crawling of both a real infant and the simulated robot. There is an interval of 120ms between each picture.

inhibited movement of a limb during the swing phase of the opposite limb. Moreover, we showed that we can use the theory of symmetric coupled cells to construct the architecture of a network of coupled oscillators, given a desired symmetry in the oscillations. This also allowed us to derive the other patterns of oscillations our network could support in a very simple manner, transforming the analytic problem into an algebraic one. The CPG we constructed has several properties that are relevant to robotics: it is stable against perturbations, which is good for sensory feedback integration, and we can easily modulate the pattern in frequency and amplitude.

Finally we showed that our model of CPG matched quite well the experimental data of a single crawling infant and we showed that it could be successfully used to control a simulated humanoid robot.

The oscillator we constructed and the design methodology we followed to build the network of coupled oscillators are general enough to be used in many other applications where rhythmic pattern generation is necessary. While the controller we built in this section is very specific to crawling, we show in the following how the introduced ideas can be used to develop a generic CPG controller for quadruped robots.

## 4.2 Inclusion of sensory feedback

In this section we extend the previous ideas for the control of different gaits and robots. We change the formulation of the oscillator to have a limit cycle that is invariant to the choice of frequency and we present generic coupling architecture for different quadrupedal

gaits. We do not re-use the inhibiting coupling presented in the previous section, since as we already mentioned this was specific to one infant and not observed in more extensive studies and this is not justified for general controllers for quadruped robots. The central aspect of this section is the introduction of sensory feedback at the oscillator level. Interestingly the inclusion of local sensory information is sufficient to greatly improve the performances of the robots.

The following is taken from a conference paper that was originally published in [114] and a journal paper that was to be submitted at the time of writing of this thesis [113].

### 4.2.1 Motivation

In the previous section, 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 [51] 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 section, 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 [7], or by using the entrainment properties of the oscillators [46, 93, 135]. Other feedback pathways have been designed in [46], 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 the presented work 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 [46] 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 and Ghostdog), using several gaits, on different terrains.

### Features of a CPG based controller

Prior to designing a CPG based controller, one must think about the kind of desirable properties the controller should have. First, the controller should be able to generate several gaits and simple parameter change should switch smoothly from one gait to the



## 4.2. Inclusion of sensory feedback

---

other. Indeed it is obvious that different gaits are desirable at different speeds. For example, one will use a walking gait at slow speed, since it is a statically stable gait but at higher speed (when the ratio between stance and step cycle durations become  $< 0.5$ ) a more dynamic gait, such as a bound or a gallop might be interesting.

From the control point of view, the CPG should be as simple as possible and computationally efficient. Indeed on a robot, one would like to use the computational power to do complex tasks on top of the locomotion control. Then we would like the system to generate smooth control policies that can be modulated by simple parameter changes. The CPG should also have a structurally stable limit cycle, in order to allow sensory feedback integration. These feedback pathways are designed to improve the robustness of the locomotion.

Finally we would like the architecture of the CPG to be generic enough and scalable to be applied to different kinds of robots. It should also be flexible in order to allow the addition of higher level controls (e.g. precise foot placement) or reflexes<sup>3</sup> (e.g. the stumbling response reflex when hitting an object during swing). In summary, as we already stated in [114], the required features for the controller are

- Generation of several gaits and smooth gait transitions
- Structural stability of the controller to allow feedback integration
- The generated policies should be smooth and modulated by simple parameter changes
- Feedback integration makes locomotion robust to parameter uncertainty and to unknown environment
- Generic and scalable architecture (application to several robots)
- Framework should allow more complex behavior

### Design assumptions

In order to reduce the design space, we make a few assumptions on the CPG. First we take a few design principles from mammalian locomotion. It is well known that speed of locomotion in animals is controlled by the duration of the stance phase while the swing phase stays almost constant over ranges of speeds and gaits (see Chapter 3). We thus assume that the same might be useful for robots, so we require independent control of these durations in our CPG (this usefulness will be demonstrated in this section).

Sensory information from cutaneous and muscle receptors plays an important role in mammalian locomotion and is strongly coupled to the locomotor CPG [44]. The effect of

---

<sup>3</sup>Here we define reflexes as fast feedforward responses to a specific sensory input, possibly dependent on the state of the CPG.

this feedback is phase dependent and mainly modulates the onset of the different phases of locomotion. Taking this principle for our controller design, we require that the swing phase should end when the corresponding limb touches the ground and the stance phase ends when the corresponding limb does not support anymore the body weight.

From the control point of view, we design the CPG as a network of coupled oscillators. One advantage of oscillators is their intrinsic synchronization capabilities that make coordination of the limbs easy and their structural stability (for feedback integration). We consider that their state space encodes the locomotion policy of the corresponding limb and thus any feedback integration explicitly modifies this state space. We can then view sensory information as having control over the CPG and we would like to emphasize the idea that the CPG and the body it controls are tightly coupled and each one can be considered as the controller of the other.

As a summary we have the following assumptions

- Independent control of swing and stance durations
- CPG encodes the control policies in phase space
- Strong coupling with the mechanical system
- Local sensory feedback modulates the onset (and the end) of swing and stance phases

### 4.2.2 CPG design

In this section we present the method we use to design the CPG controller. The design is done in three steps: first we present a locomotion specific oscillator, then we design generic networks able to support different gaits, and finally we explain the integration of the local sensory feedback.

#### Locomotion specific oscillator

As we explained in the previous section we would like to be able to independently control the duration of the swing and stance phases. We thus need an oscillator in which we can specify the durations of the swing and stance phases. In the following we identify these phases with the ascending and descending phases of the oscillation. In conventional oscillators, we can generally change the frequency of oscillations but this has an effect on both phases. In order to independently control both phases we use a modified Hopf oscillator that has a phase-dependent frequency. It is based on the oscillator we designed in [111] and that we presented in Section 4.1, but normalized such that the limit cycle

## 4.2. Inclusion of sensory feedback

---

has an invariant shape for different frequencies. Its equation is

$$\dot{x} = \alpha(\mu - r^2)x - \omega y \quad (4.15)$$

$$\dot{y} = \beta(\mu - r^2)y + \omega x \quad (4.16)$$

$$\omega = \frac{\omega_{stance}}{1 + e^{-by}} + \frac{\omega_{swing}}{1 + e^{by}} \quad (4.17)$$

where  $r = \sqrt{x^2 + y^2}$ . The resulting system is an oscillator, whose limit cycle is a circle in phase space, the amplitude of oscillations is  $\sqrt{\mu}$  and the instantaneous frequency is  $\omega$   $\text{rad} \cdot \text{s}^{-1}$  (and the frequency of oscillations is  $\simeq \frac{\omega_{swing} + \omega_{stance}}{2}$   $\text{rad} \cdot \text{s}^{-1}$ ). Since  $\omega$  switches from  $\omega_{stance}$  to  $\omega_{swing}$  according to the sign of  $y$ , we can control explicitly the duration of the ascending and descending phases of the oscillator.  $b > 0$  controls the speed of switch between the frequencies (we use  $b = 100$ ).  $\alpha$  and  $\beta$  are positive constants that control the speed of convergence to the limit cycle in the  $x$  and  $y$  directions respectively. We use  $\alpha \ll \beta$  in order to have a fast convergence to the limit cycle in the  $y$  direction and to have a slower convergence in the  $x$  direction. Indeed, since we use  $x$  as the output for our control policy, its velocity ( $\dot{x}$ ) profile should be bounded with a small velocity. In the following we use  $\alpha = 5$  and  $\beta = 50$ .

The design of this oscillator is motivated by the following properties. It has a circular limit cycle, whose radius is independent of the frequency. The output is a sine for  $\omega_{swing} = \omega_{stance}$ . Furthermore, it is structurally stable, so small perturbations cannot destroy its qualitative behavior, which is important if we want to couple oscillators together and to incorporate sensory feedback. Moreover we have complete and independent control on its oscillations amplitude and frequencies. An example of behavior of this oscillator is shown in Figure 4.11.

### Network design

The oscillators in the CPG are coupled such that they generate the desired gaits (i.e. phase differences between the oscillators). Our goal is to create generic networks for different gaits, such that the coupling architecture is independent of the dynamics of the oscillator. In this section we show how we can create generic networks supporting walk, trot, pace and bound gaits.

In order to do so, we use the theory of symmetric coupled cells network developed by Golubitsky and colleagues [50, 51] as we did in Section 4.1. Here we discuss in greater details the approach and extend the previous results. The strength of this theory is that it gives conditions on the coupling architecture of a network such that it can produce patterns with a defined symmetry. Moreover this relies only on algebraic arguments which makes the design easier and scalable to more complex networks. Another interesting point in this theory is that it allows one to calculate other possible solutions in the network by looking only at the symmetries in the architecture. This is really useful in the design

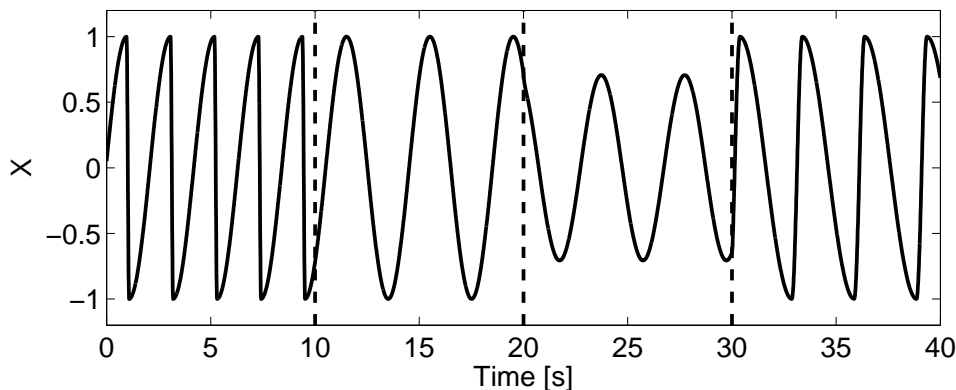


Figure 4.11: Example of behavior of the oscillator of Equations (4.15)-(4.17). The  $x$  variable is shown. At  $t = 0$  we have  $\omega_{swing} = 0.5\pi$ ,  $\omega_{stance} = 8\pi$  and  $\mu = 1$ . At  $t = 10$ , we change  $\omega_{swing} = \omega_{stance} = 0.5\pi$ . At  $t = 20$ , we change  $\mu = 0.5$  and at  $t = 30$ , we set  $\omega_{swing} = 2\pi$ ,  $\omega_{stance} = 0.4\pi$  and  $\mu = 1$ .

of CPGs because we can then make sure that only the desired pattern is stable and understand the possible behavior of the network.

The main idea of the theory is the following: in order to generate patterns (i.e. solutions of the differential equations) possessing certain symmetries, in a network of coupled dynamical systems, one has to design a network possessing the same symmetries.

Let  $A$  define the coupling matrix of a network (in the following we assume that all the cells of the network are identical). We say that the linear transformation  $\gamma$  is a symmetry of the network if  $\gamma A = A$ . It means that the structure of the network is invariant under  $\gamma$ . The set of symmetries of  $A$  form a group. We can see these symmetries as permutations of the cells and coupling that preserves the topology of the network.

There are two types of symmetries for a solution of a differential equation  $x(t)$ . *Spatial symmetries* are linear transformations  $\gamma$  such that  $\gamma x(t) = x(t)$ ,  $\forall t$ . *Spatio-temporal symmetries* are transformations  $\sigma$  such that  $\sigma x(t) = x(t + \psi)$ , they are symmetries of the solution, up to a time shift.

The main result we use is the H/K theorem from Golubitsky and colleagues [50] that we showed in Section 4.1. Then to design a CPG supporting a certain type of gait we can calculate the symmetries of the gait and find the minimal constraints on the network without considering the actual dynamics of the oscillators in the network. Finally one has just to make sure that the conditions of the theorem are met. All this process can be automated since it is a purely combinatorial process.

Since we develop controllers for quadruped robots, we use a network with four nodes. Note however that from the biological perspective it was argued in [52] that it is more likely that animals have at least an eight cells network generating the gaits. In this contribution they suppose that a single network should be able to generate the walk, trot

## 4.2. Inclusion of sensory feedback

---

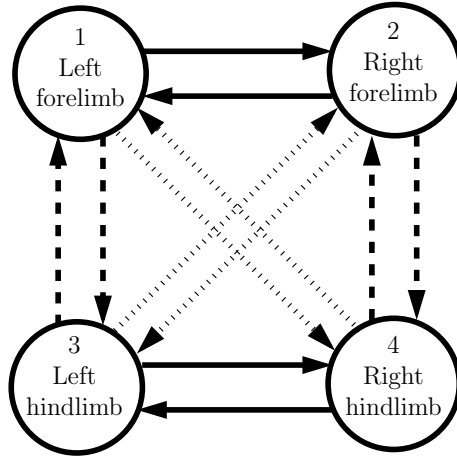


Figure 4.12: Network generating the trot, pace and bound gaits that was used in [114]. The same type of arrows denotes the same coupling function. The new networks, where new symmetries are added are shown in Figure 4.13.

and pace gaits such that trot and pace are not conjugate solutions (since these gaits do not appear to be used by the same animals). For engineering purpose, we do not require to have a single rigid network that generates all the gaits (i.e. the coupling structure between the cells can change when changing the gait) and this is not a problem to have conjugate solutions. Since we do not see for now a good reason from that point of view to use a more complex network and we want to have a CPG that is as simple as possible we use a four cell network for the CPG. However we note that an eight cell network might make more sense if there were antagonist actuators (like artificial muscles) for each joint of the robots, but it is not the case in this study.

If we look at the trot gait where the diagonal limbs move in synchrony (e.g. left fore and right hind legs) and are half a period out of phase with the opposite limbs and if we number the limbs as in Figure 4.13, one spatial symmetry of this gait is  $((14)(23), 0)$  and one spatio-temporal symmetry is  $((12)(34), \frac{1}{2})$ . Here we use permutation notation where the first elements describe how we permute the cells and the last number defines the phase shift induced by the symmetry transformation as a proportion of the period of the solution. In a previous paper [114] and in Section 4.1 we considered the spatio-temporal symmetries of the trot gait generated by  $((12)(34), \frac{1}{2})$  and  $((13)(24), \frac{1}{2})$ . These generators are the same used previously by Golubitsky in [50]. These symmetries generate a group that is isomorphic to  $\mathbb{Z}_2 \times \mathbb{Z}_2$ . For the pace, the generators were  $((12)(34), \frac{1}{2})$  and  $((13)(24), 0)$  and for the bound they were  $((12)(34), 0)$  and  $((13)(24), \frac{1}{2})$ . The corresponding groups are also isomorphic to  $\mathbb{Z}_2 \times \mathbb{Z}_2$ . The resulting network was then the same for the three gaits and is pictured in Figure 4.12.

However it seems that we can some more spatial symmetries to these gaits. For ex-

ample, in the case of the trot gait, the only non trivial spatial symmetry present in the previous group is  $((14)(23), 0)$ . However if the cells are the same and we make no distinctions between the fore and hind legs, then we can also add the following two symmetries  $((14)(2)(3), 0)$  and  $((23)(1)(4), 0)$ . Then the resulting group of spatio-temporal symmetries has now 8 elements and is isomorphic to  $\mathbb{D}_4$ . The group of spatial symmetries is isomorphic to  $\mathbb{Z}_2 \times \mathbb{Z}_2$  and the quotient  $\mathbb{D}_4/\mathbb{Z}_2 \times \mathbb{Z}_2 \simeq \mathbb{Z}_2$  is cyclic (so the conditions of the H/K theorem are fulfilled). We get similar groups of the pace and bound gaits which are all isomorphic to  $\mathbb{D}_4$ . By adding these symmetries we are able to constraint even more the topology of the network and we finally get the 3 distinct networks for trot, pace and bound that are shown in Figure 4.13. We note that the final coupling matrices that we used to generate stable trot, pace and bound in [114] are compatible with the new networks and were indeed already possessing these additional symmetries (except that we did not notice them).

For the case of the walk, there is no non-trivial spatial symmetry and the temporal symmetry  $((1423), \frac{1}{4})$  generates a group that is isomorphic to  $\mathbb{Z}_4$ . The resulting network is shown in Figure 4.13. There exist two types of walk (if we use the standard definition of gaits of Hildebrand [59]), the *diagonal sequence* walk where the limbs swing in the sequence right hind, left fore, left hind and right fore and the *lateral sequence* walk where the limbs swing in the sequence right hind, right fore, left hind, left fore. The network we designed generically supports both gaits.

In this section we are interested in quadruped locomotion and simple intuition on oscillators would have been sufficient to design the previous networks. However, this intuition might become more difficult for CPGs with a higher number of cells (for example for hexapod locomotion), while the method using symmetry offers an algorithmic way to define such networks that is scalable. Moreover we think it is important to provide a framework to design such systems that goes beyond mere intuition.

Furthermore, by calculating the lattice of subgroups of the networks we designed we can calculate the other possible periodic solutions in the network and make sure that only the one that interests us is stable by finding appropriate coupling parameters. In Table 4.1 we show the possible solutions we found for the trot and walk networks with the corresponding group of symmetries. Note that for the trot network, not all the possible combinations of groups are shown since several different combinations lead to the same solution, here we noted only one combination for each solution. In the case of the pace and bound networks, the possible solutions are similar to the one of the trot, up to a permutation of the cells, since their group of symmetries are isomorphic. The pattern of solutions we found are generic patterns (i.e. they exist in general) and do not depend on the dynamics of the cells. The exact form of the pattern (e.g.  $x(t)$ ) will be defined by the internal dynamics and the coupling.

## 4.2. Inclusion of sensory feedback

---

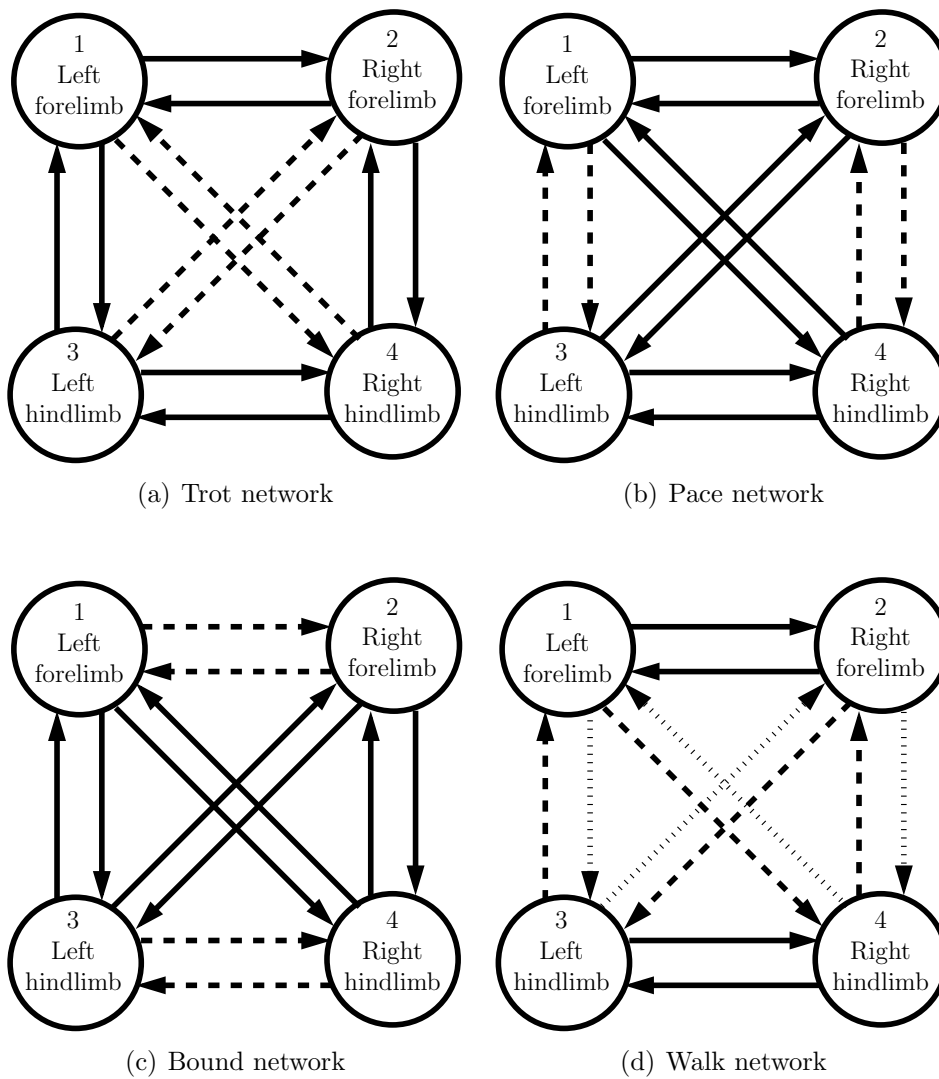


Figure 4.13: Generic networks corresponding to four different gaits. The same type of arrows denote the same type of coupling functions.

Possible solutions in the walk network		
H	K	Pattern of oscillations
$\mathbb{Z}_4$	$I$	$x(t), x(t + \frac{1}{2}), x(t \pm \frac{1}{4}), x(t \pm \frac{3}{4})$
	$\mathbb{Z}_2$	$x(t), x(t), x(t + \frac{1}{2}), x(t + \frac{1}{2})$
	$\mathbb{Z}_4$	$x(t), x(t), x(t), x(t)$
$\mathbb{Z}_2$	$I$	$x(t), x(t + \frac{1}{2}), y(t), y(t + \frac{1}{2})$
	$\mathbb{Z}_2$	$x(t), x(t), y(t), y(t)$
$I$	$I$	$x(t), y(t), z(t), w(t)$

Possible solutions in the trot network		
H	K	Pattern of oscillations
$\mathbb{D}_4$	$\mathbb{D}_4$	$x(t), x(t), x(t), x(t)$
	$\mathbb{Z}_2(\tau) \times \mathbb{Z}_2(\sigma)$	$x(t), x(t + \frac{1}{2}), x(t + \frac{1}{2}), x(t)$
$\mathbb{Z}_4(\varrho)$	$I$	$x(t), x(t \pm \frac{1}{4}), x(t \pm \frac{3}{4}), x(t + \frac{1}{2})$
$\mathbb{Z}_2(\tau) \times \mathbb{Z}_2(\sigma)$	$\mathbb{Z}_2(\tau) \times \mathbb{Z}_2(\sigma)$	$x(t), y(t), y(t), x(t)$
	$I$	$x(t), y(t), y(t + \frac{1}{2}), x(t + \frac{1}{2})$
$\mathbb{Z}_2(\kappa)$	$\mathbb{Z}_2(\kappa)$	$x(t), x(t), y(t), y(t)$
	$I$	$x(t), x(t + \frac{1}{2}), y(t), y(t + \frac{1}{2})$
$\mathbb{Z}_2(\nu)$	$\mathbb{Z}_2(\nu)$	$x(t), y(t), x(t), y(t)$
	$I$	$x(t), y(t), x(t + \frac{1}{2}), y(t + \frac{1}{2})$
$\mathbb{Z}_2(\sigma)$	$\mathbb{Z}_2(\sigma)$	$x(t), y(t), y(t), z(t)$
	$I$	$x(t), y(t), y(t + \frac{1}{2}), z(t)$
$\mathbb{Z}_2(\tau)$	$\mathbb{Z}_2(\tau)$	$x(t), y(t), z(t), x(t)$
	$I$	$x(t), y(t), z(t), x(t + \frac{1}{2})$
$I$	$I$	$x(t), y(t), z(t), w(t)$

Table 4.1: These two tables show the different possible solutions corresponding to the trot and walk networks. For both networks we show the possible pattern of solution for the 4 cells together with the associated group of spatial (K) and spatiotemporal (H) symmetries. For example  $(x(t), y(t), y(t + \frac{1}{2}), z(t))$  means that the solutions for cells 2 and 3 are the same up to a time shift of half a period, while the solutions of cells 1 and 4 are different. For the walk network  $\mathbb{Z}_4$  is the group generated by  $((1423))$ , while  $\mathbb{Z}_2$  is generated by  $((12)(34))$ . For the trot network  $\mathbb{D}_4$  is the full group of symmetries of the gait, for the other groups we show the generators in parentheses, where  $\tau = ((14)(2)(3))$ ,  $\sigma = ((1)(23)(4))$ ,  $\kappa = ((12)(34))$ ,  $\nu = ((13)(24))$  and  $\varrho = ((1243))$ .



## 4.2. Inclusion of sensory feedback

---

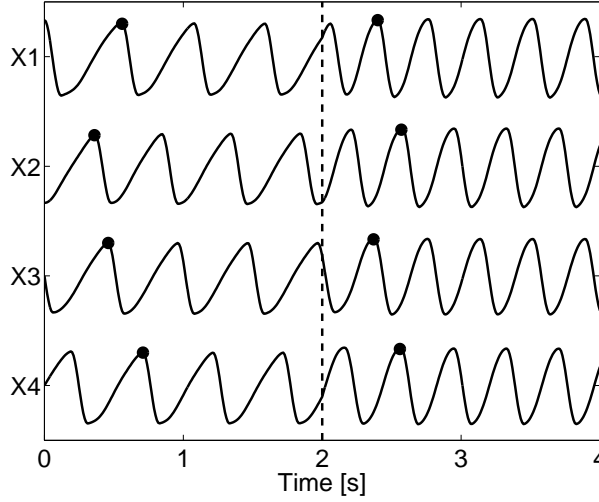


Figure 4.14: Example of a gait transition. At time  $t = 2$  we switch the coupling of the network from the walk matrix to the pace matrix. We also change  $\omega_{stance}$  from  $2\pi$  to  $4\pi$ .  $\omega_{swing} = 8\pi$  for all  $t$ .

Now we couple the oscillators as follows

$$\dot{x}_i = \alpha(\mu - r_i^2)x_i - \omega_i y_i \quad (4.18)$$

$$\dot{y}_i = \beta(\mu - r_i^2)y_i + \omega_i x_i + \sum_j k_{ij} y_j \quad (4.19)$$

$$\omega_i = \frac{\omega_{stance}}{1 + e^{-by_i}} + \frac{\omega_{swing}}{1 + e^{by_i}} \quad (4.20)$$

where the  $k_{ij}$  define a coupling matrix that depends on the chosen gait and that is compatible with the network we designed. These matrices are shown in Figure 4.15 together with an example of the associated pattern of oscillations. The type of gait (in terms of spatio-temporal relations) depends only on the coupling matrices and is not affected by the swing and stance durations. These durations will only affect the period and duty factor of the gait (ratio of stance duration over one cycle duration). It is also very easy to smoothly change the gait by simply changing the coupling matrix, as can be seen in Figure 4.14. We chose the coupling matrices such that only the possible solutions that interests us are stable for each set of matrices. We tested numerically the instability of the other solutions (the ones for which we knew the form of the steady state solution because for cases where solutions are of the form  $(x(t), y(t), z(t), w(t))$  we do not know what is the form of the solution) and from our numerical simulations, only the desired gaits were stable. Figure 4.16 shows an example for the trot network. We must note that for the walk network, both the diagonal and lateral sequence gaits are stable with a bounded region of stability. The gaits from the other networks seem to be globally stable from the numerical experiments we made.

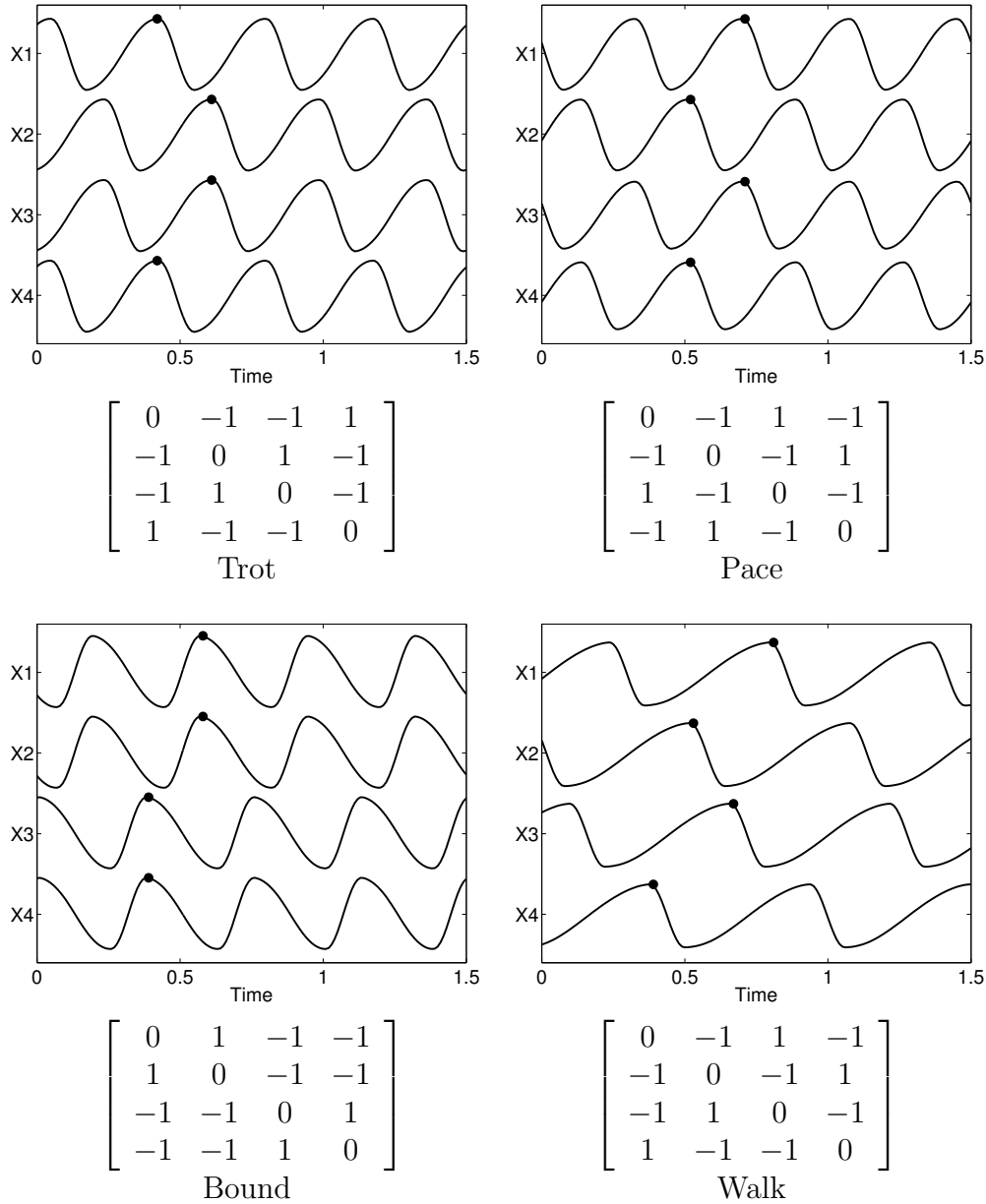


Figure 4.15: Coupling matrices and examples of gait generation for the 4 gaits.  $\omega_{stance} = 2\omega_{swing}$  for the trot and pace gaits.  $\omega_{stance} = 4\omega_{swing}$  for the walking gait and  $\omega_{swing} = 2\omega_{stance}$  for the bound.

## 4.2. Inclusion of sensory feedback

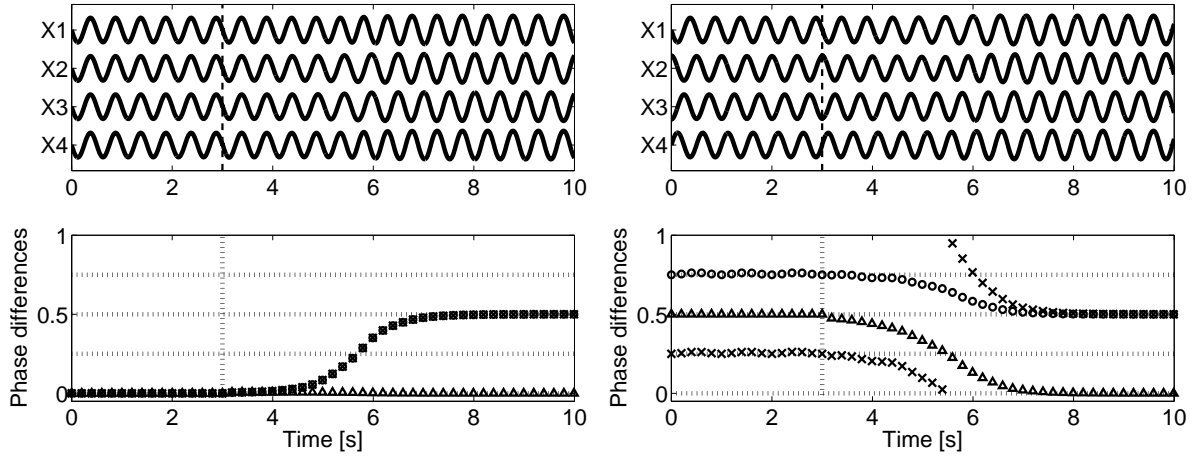


Figure 4.16: These two figures show the existence of two generic solutions of the trot network and their instability, since at  $t = 3$  we perturb the oscillators with a small perturbation (1% of the amplitude of the oscillator). The upper graph shows the behavior of the  $x$  variable of each cell in the network. The lower graph shows the phase differences between the first cell and the second ( $\times$ ), the third ( $\circ$ ) and the fourth ( $\Delta$ ) cell. The initial conditions for the left graph are  $(x(t), x(t), x(t), x(t))$  and for the right graph  $(x(t), x(t + \frac{1}{4}), x(t + \frac{3}{4}), x(t + \frac{1}{2}))$ .

### Local sensory feedback

Now we show the integration of sensory feedback from the pressure sensors located on the extremities of the limbs. We know from mammalian locomotion [44] that the onsets of swing and stance phases are critical and widely sensitive to feedback information from the stretch receptors in the muscles and cutaneous information. The basic modulation of the onset of each phases works as follows: at the end of swing, the limb starts stance only when it has touched the ground, if it touches the ground before than expected then swing phase is shortened, otherwise it continues to some stop position until it touches the ground. For the stance to swing modulation, transition is accelerated if the limb does not support the body anymore and is delayed while it support a significant amount of body weight.

In our system we want to integrate local information from the pressure sensors such that each oscillator can modulate the generation of swing and stance accordingly. Since we want to explicitly modify the state space according to the desired behavior we rewrite

the equation of one oscillator integrating a control  $u_i$  that accounts for the feedback.

$$\dot{x}_i = \alpha(\mu - r^2)x_i - \omega_i y_i \quad (4.21)$$

$$\dot{y}_i = \beta(\mu - r^2)y_i + \omega_i x_i + \sum_j k_{ij}y_j + u_i \quad (4.22)$$

$$\omega_i = \frac{\omega_{stance}}{1 + e^{-by_i}} + \frac{\omega_{swing}}{1 + e^{by_i}} \quad (4.23)$$

We add the control on the  $y_i$  variable for two principal reasons. First because the feedback will be discontinuous (touch information is not continuous) and we use  $x_i$  as a control policy, then on the second state variable, by integration we are sure to have a continuous policy and velocity profile. Second because if we look at the phase space of the oscillator, the swing and stance phases can be discriminated by the sign of  $y_i$  as it is shown in Figure 4.17, thus if we can control this state variable, we can control the transition between the two phases.

The goal now is to determine  $u_i$  such that the oscillator has a behavior that resembles the one that we see in mammals. There are two aspects to integrate, one is a fast transition from one phase to the other, the second is a stop before the transition.

**Fast transition** We want a fast transition from one phase to the other in two cases. During stance we want fast transition to swing if the limb does not support anymore the body. During swing, as fast transition to stance happens if the limb touches the ground before the end of swing.

To accelerate the transition, we need to make  $y_i$  go to 0 as fast as possible. Thus we choose  $u_i = F$  with  $F$  high enough. Its sign depends on the direction of transition (positive for swing to stance, negative for stance to swing). When we use this control, we can approximate  $\dot{y}_i \simeq F$  since  $F$  will be much bigger than the other terms of the differential equation. Thus the delay before transition will be approximately  $\frac{y(t_{transition})}{F}$  seconds, where  $t_{transition}$  denotes the time at which transition starts.

One could argue that it would be easier to directly reset  $y$  to 0, but the purpose of doing it this way is that we can explicitly control the delay before transition. Of course we could reset after some delay, but one would need to do that explicitly outside of the equations.

**Stop before transition** We want the oscillator to stop at a limit point in two cases. During stance when the limb still supports a significant amount of the body weight and during swing when the limb has not yet touched the ground.

We want the oscillator to stop at the limit position  $x = \pm\sqrt{\mu}$  with  $y = 0$ , so we require  $\dot{x} = \dot{y} = 0$ . We obtain this with  $u_i = -\omega_i x_i - \sum k_{ij}y_j$ . We can then control the stopping position by controlling  $\mu$ .

## 4.2. Inclusion of sensory feedback

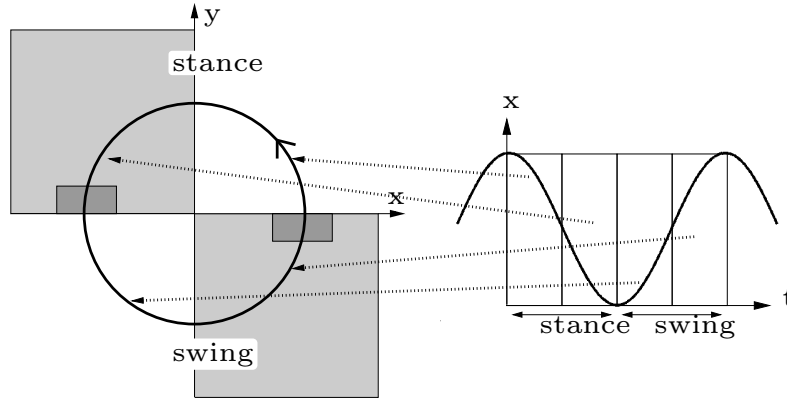


Figure 4.17: Phase space of an oscillator (left figure) with its activation zone for the feedback (light gray for transition and dark gray for the stop controls). Correspondence with the  $x$  variable of the oscillator is shown on the right.

To understand the stability of such a fixed point, we linearize the equation with the new control at  $x = \pm\sqrt{\mu}$ ,  $y = 0$ . The Jacobian matrix at these points is

$$Df = \begin{bmatrix} -2\alpha\mu & -\omega_i \\ 0 & 0 \end{bmatrix} \quad (4.24)$$

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 point is tangent to the  $Ox$  axis (it is in fact the  $Ox$  axis). The flow cannot cross the  $x$  axis and converges to the desired fixed points. However when it has crossed the  $x$  axis it cannot come back and goes towards the other fixed point. The activation of the feedback for an oscillator depends on its phase and on the pressure sensors information of the corresponding limb. The activation regions of the feedback are shown in Figure 4.17. The resulting phase space of the oscillator is shown in Figure 4.18. In summary the feedback goes as

$$u_i = \begin{cases} -\text{sign}(y_i)F & \text{fast transitions} \\ -\omega_i x_i - \sum k_{ij} y_j & \text{stop before transition} \\ 0 & \text{otherwise} \end{cases} \quad (4.25)$$

### 4.2.3 Experimental setup

In this section we explain how we map the policies generated by the CPG to the different joints of a quadruped robot. Then we give a description of the robots we use to perform our experiments.

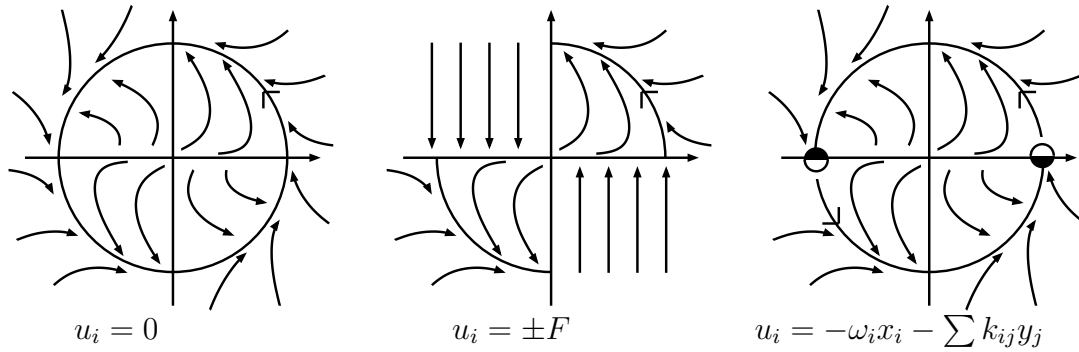


Figure 4.18: Schematic oscillator phase plot for the different types of feedback.

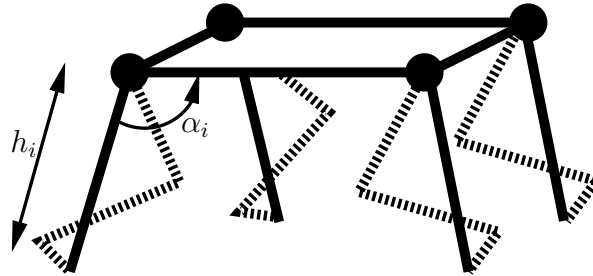


Figure 4.19: Schematic of a quadruped robot where the abstract limbs are represented by a segment defined by a length  $h_i$  and an angle  $\alpha_i$  relative to the body. The dashed segments represent a possible representation of the real limbs corresponding to this abstraction, with the detailed kinematic chains.

### Generation of the trajectory policy

So far, the CPG we presented generates a policy that coordinates the different limbs of a quadruped robot and modulates the transitions between the different phases of each limb according to local sensory information (in our case the pressure sensors located at the extremities of the limbs).

For different robots, the kinematic chain of the limbs can be very different and we would like to have a controller that generates a unified control independently of these differences. We consider an abstract representation of the limbs of a quadruped robot, as it is shown in Figure 4.19. For each abstract limb we can control two parameters, the length of the limb and its angle relative to the body (currently we control locomotion only in the sagittal plane).

The  $x_i$  variable of the CPG controls the angle  $\alpha_i$  as

$$\alpha_i = A_i x_i + B_i \quad (4.26)$$

where  $A_i$  defines the amplitude of oscillations of  $\alpha_i$  and  $B_i$  is the set point around which

## 4.2. Inclusion of sensory feedback

---

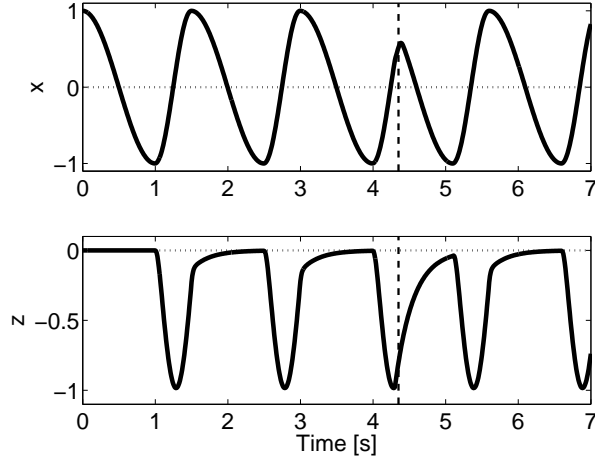


Figure 4.20: We show an example of the evolution of  $z$ , defined by Equation (4.27), together with the evolution of the  $x$  variable of the corresponding oscillator. At  $t = 4.3$ , we simulate a fast transition from swing to stance (vertical dashed line). Here the parameters used are  $K_1 = 30$ ,  $K_2 = 100$ ,  $\omega_{swing} = 2\pi$  and  $\omega_{stance} = \pi$ .

$\alpha_i$  oscillates.

We then need to control the height  $h_i$  of the limb. The simple solution for this controller could be to have  $h_i$  constant during stance phase and  $h_i$  decreasing at the beginning of swing and increasing again at the end. We need  $h_i$  to be continuous and if swing is shortened because of feedback it should then go to the maximum height in a controllable time. A possible controller for this variable is the following second order critically damped switching linear system

$$\ddot{z}_i = \begin{cases} -K_1 \dot{z}_i - \frac{K_1^2}{4} z_i & \text{if } y_i > 0 \\ -K_2 \dot{z}_i + \frac{K_2^2}{4} (y_i - z_i) & \text{else} \end{cases} \quad (4.27)$$

with the output

$$h_i = h_{max} + C_i z_i \quad (4.28)$$

where  $K_1$  and  $K_2$  are positive constants,  $h_{max}$  is the height of the limb during stance and  $C_i$  is the amplitude elevation of the limb during swing. During swing,  $h_i$  follows the movement of  $y_i$  which makes a half sine wave and during stance  $h_i \rightarrow h_{max}$ .

In the following we use the two variables  $\alpha_i$  and  $h_i$  to control each limb of the robot, the appropriate angles of the different joints of the robots are set using inverse kinematics. We control all the robots in position with high gain PID controllers, since this is their standard mode of operation. We note however that it is not a satisfactory (and definitive) solution and a force controller would be more interesting, in particular to deal with the intermittent contacts of the limbs with the ground.

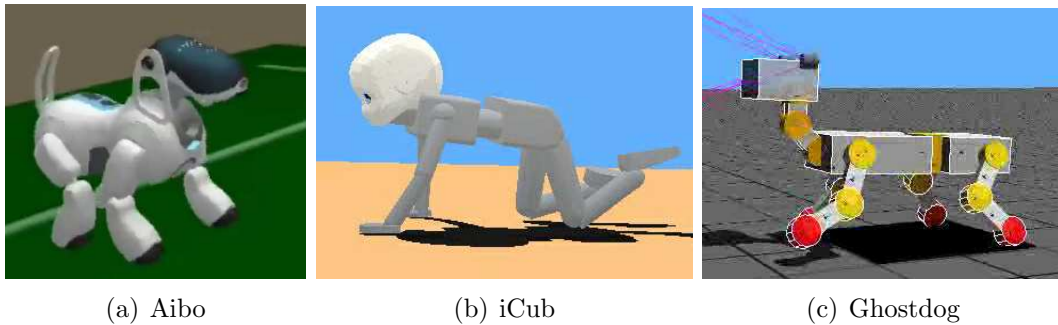


Figure 4.21: The three different robots used in the experiments.

### Controlled robots

We tested our controller on three different robots in simulation: the Aibo robot, the iCub humanoid and the ghostdog robot. These robots have very different mechanical properties. The simulator we used is Webots [90], a software for the physics simulation of robots based on ODE [2], an open source library for the simulation of rigid body dynamics. The software is distributed with a library of different robot simulations including the ones we use here. These simulations are calibrated to match as close as possible the real robots.

The Aibo robot is the well known dog-like robot from Sony. Each limb is composed of two segments, there are two joints at the hip level to control flexion/extension and abduction/adduction and one joint at the knee level. There are pressure sensors at the extremities of the limbs.

The iCub robot is the open source humanoid robot [141] with the dimensions of a two year old child that we described in Chapter 2. As an infant like robot it is able to crawl on its hands and knees, we use it in this configuration. It has 53 degrees of freedom. The joints that we use are the 3 degrees of freedom at the shoulder and hip level (flexion/extension, adduction/abduction and rotation) and one degree of freedom at the knee and elbow joints. There are pressure sensors on the hands and knees. For the knees we do not control the height of the limb ( $h = \text{constant}$ ), since there is only one segment between hip and knee joints.

The previous two robots are quite rigid robots with no compliance. The third one, the ghostdog robot (GD) is an imaginary robot that was developed by Cyberbotics for the Webots software and that has some passive compliance. Each limb has two segments with two degrees of freedom in the sagittal plane (one at the hip level and one at the knee). There is a linear spring in series with the knee joint that gives some compliance and passive dynamics to the robot. Although it is an abstract model, it resembles the Puppy robot [64] but has more degrees of freedom. A robot having a similar dynamics was just designed [120]. A picture of each robots is shown in Figure 4.21.

The parameters that can be changed are the amplitude of oscillations  $A_i$  of  $\alpha_i$ , its



## 4.2. Inclusion of sensory feedback

---

set position  $B_i$ ,  $h_{max}$ ,  $C_i$ ,  $\omega_{swing}$  and  $\omega_{stance}$ . For all the experiments we fixed all the parameters (which are different for the three robots) but  $\omega_{swing}$  and  $\omega_{stance}$  since from various test we made, the results seemed to be robust to the choice of the other parameters. Moreover our focus in this section is to explore the basic properties of the controller rather than the optimal parameter set for a given robot. Of course for other applications, it might be interesting to optimize these parameters.

$h_{max}$  is set to be close to the maximum limb height and  $C_i$  is not crucial as long as the limb can move forward without touching the ground. The amplitude  $A_i$  of  $\alpha_i$  has an influence on the speed of locomotion but is limited by the maximum torques available in the joints of the robot. The offset  $B_i$  has an influence on the stability of the robot on slopes but it turned out that for variations of this parameter we obtained the same qualitative results.

### 4.2.4 Experiments and results

In this section we present results from various experiments in order to show the effectiveness of the approach and the basic properties of the controller, namely the strong linear correlation between speed and  $\omega_{stance}$  (that can thus serve as a parameter to control speed), the robustness to parameter choice and the increased stability of the robot.

#### Flat ground experiments

We first made flat ground experiments with a systematic exploration on the parameters  $\omega_{swing}$  and  $\omega_{stance}$ . Our goal in these experiments is to understand the influence of the feedback in the performances of the robot. Because the feedback is stronger than the inter-oscillator coupling, we also would like to understand to what extent the coupling between the oscillators is necessary. Another goal of these experiments is to explore how  $\omega_{swing}$  and  $\omega_{stance}$  affect locomotion speed.

We made three different types of experiments, one with the CPG without feedback (Experiment A), the second with feedback but without coupling between the oscillators<sup>4</sup> (Experiment B) and the third with feedback and coupling between the oscillators (Experiment C). Each trial was evaluated during 40s. The initial posture of the robot is a standing posture with  $\alpha_i = \frac{\pi}{2}$  rad. We initialize the CPG such that its state is compatible with the state of the robot. We chose this posture in order to test the transient behavior of the robot, before going to steady state locomotion. We tested the Aibo with a walking gait, the iCub with a trot gait and the GD with a bound gait.

**Aibo robot** In Figure 4.22 we show the speed of locomotion of the Aibo depending on the different parameters. The robot never falls for the whole range of parameters (this

---

<sup>4</sup>Indirect coupling still exists through the mechanics of the robot.

	Aibo		iCub		Ghostdog	
	$\omega_{swing}$	$\omega_{stance}$	$\omega_{swing}$	$\omega_{stance}$	$\omega_{swing}$	$\omega_{stance}$
Exp. A (coupling, no feedback)	<b>0.45</b>	<b>0.51</b>	<b>0.28</b>	<b>0.88</b>	<b>0.33</b>	<b>0.30</b>
Exp. B (no coupling, feedback)	0.1	<b>0.57</b>	0.13	<b>0.84</b>	0.17	<b>0.69</b>
Exp. C (coupling, feedback)	0.03	<b>0.67</b>	0.24	<b>0.8</b>	<b>0.28</b>	<b>0.56</b>

Table 4.2: Correlation between speed of locomotion and  $\omega_{swing}$  and  $\omega_{stance}$ . We used a Spearman correlation test. The numbers in bold are correlations with a corresponding p-value  $< 0.01$ .

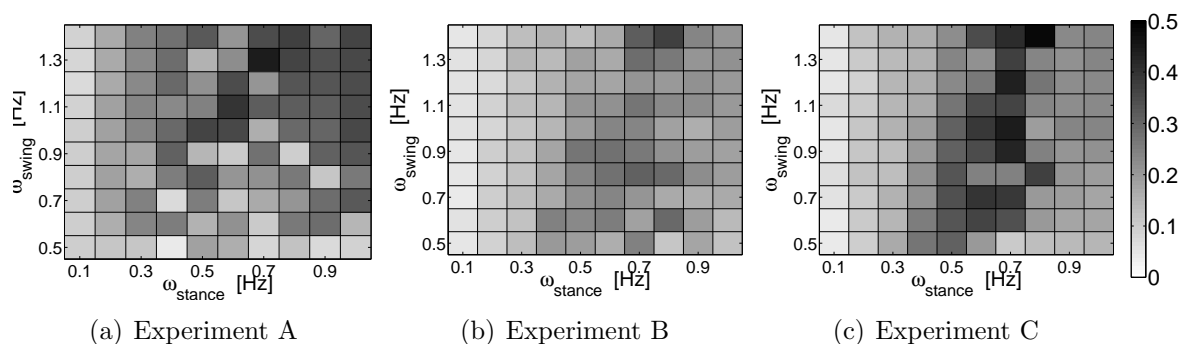


Figure 4.22: We show the locomotion speed of the Aibo for different values of  $\omega_{stance}$  and  $\omega_{swing}$ . The values of the parameters are shown in Hz (in Equation (4.17) they are expressed in  $\text{rad} \cdot \text{s}^{-1}$ ). The locomotion speed is expressed in  $\text{body length} \cdot \text{s}^{-1}$ .

was expected since the Aibo robot is intrinsically very stable). The maximum speed of locomotion for the Experiment A is  $0.44 \text{ body length} \cdot \text{s}^{-1}$ . For Experiment B it is  $0.37 \text{ body length} \cdot \text{s}^{-1}$  which is 16% slower. The best performance is attained in Experiment C, where the CPG has coupling and feedback. In this experiment speed of locomotion can attain  $0.48 \text{ body length} \cdot \text{s}^{-1}$  which is 9% faster than Exp. A. The speed performances of the robot in Exp. B are much smaller than in the two other experiments. For Experiments A and C the speed performances are similar but we notice that for small values of  $\omega_{swing}$  the speed degrades a lot in Exp. A while it is not the case in Exp. C. In that case the speed decreases for too high values of stance duration. It seems that it comes from the fact that the servos of the robot attain their limit (the performances in trajectory following of the motors degrade). It is not the case for Exp. A because the average frequency of the trajectories sent to the robot are lower than in Exp. C (because the fast transitions shorten the cycle durations). In all experiments we see a strong linear correlation between speed of locomotion and  $\omega_{stance}$ , which is similar to animal studies, while the correlation with  $\omega_{swing}$  is statistically significant only in the case of the control without feedback (see Table 4.2).

## 4.2. Inclusion of sensory feedback

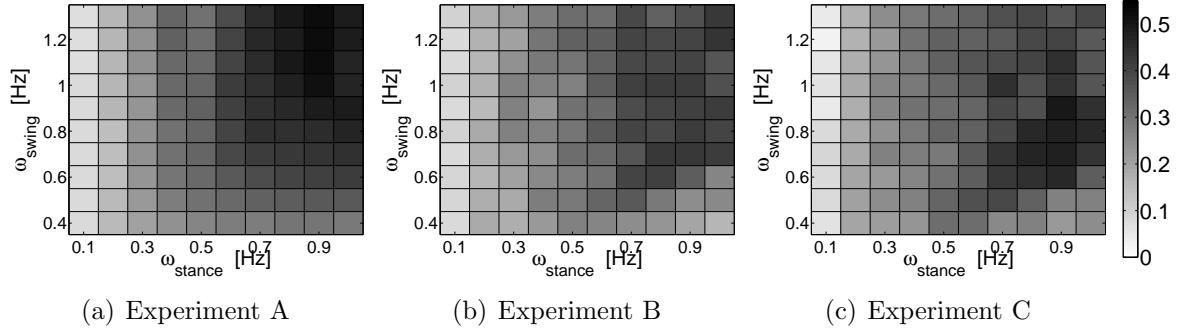


Figure 4.23: We show the locomotion speed of the iCub for different values of  $\omega_{stance}$  and  $\omega_{swing}$ . The values of the parameters are shown in Hz (in Equation (4.17) they are expressed in  $\text{rad} \cdot \text{s}^{-1}$ ). The locomotion speed is expressed in  $\text{body length} \cdot \text{s}^{-1}$ .

**iCub robot** The speed of locomotion of the iCub robot as a function of the parameters  $\omega_{swing}$  and  $\omega_{stance}$  is shown in Figure 4.23. The robot never falls for the parameter set (this was also expected since the crawling iCub is quite stable). The maximum speed of locomotion for the Experiment A is  $0.52 \text{ body length} \cdot \text{s}^{-1}$ , while it is  $0.43 \text{ body length} \cdot \text{s}^{-1}$  for Experiment B (17% slower) and  $0.49 \text{ body length} \cdot \text{s}^{-1}$  for Experiment C (6% slower than in Exp. A). The speeds in parameter space are smoother than in the case of the Aibo and it is especially visible for Experiment A. In most parameter configurations the speed of locomotion of the robot controlled by the CPG without coupling (Exp. B) is smaller than in the other experiments. For Exp. A and C the differences of speed depend on the parameters and they do not attain their maximum speed at the same location in parameter space. As for the Aibo robot, there is a strong linear correlation between speed of locomotion and  $\omega_{stance}$  for each experiment while there is a statistically significant correlation with  $\omega_{swing}$  only in Experiment A (see Table 4.2)

**Ghostdog** Figure 4.24 shows the speed of locomotion of the GD robot as a function of  $\omega_{swing}$  and  $\omega_{stance}$ . First we notice that the robot falls for certain parameter choices in the three experiments. It is Experiment B that the robot falls the most, while it is in Experiment C that the robot falls in the least cases. We note here that the GD robot is not intrinsically stable, mainly due to its passive dynamics and mechanical structure (long legs compared to body width). The maximum speed of locomotion of the robot is  $1.48 \text{ body length} \cdot \text{s}^{-1}$  in the case of Experiment A and  $1.72 \text{ body length} \cdot \text{s}^{-1}$  in the two other experiments, which is an increase of 16% in locomotion speed. From the speed plots, it is clear that feedback increases drastically the speed performances for most parameter combinations while the coupling improves the stability of the robot. There is also a strong linear correlation between locomotion speed and  $\omega_{stance}$  as can be seen in Table 4.2 but there is also a correlation with  $\omega_{swing}$  in the two experiments with coupling (Exp. A and

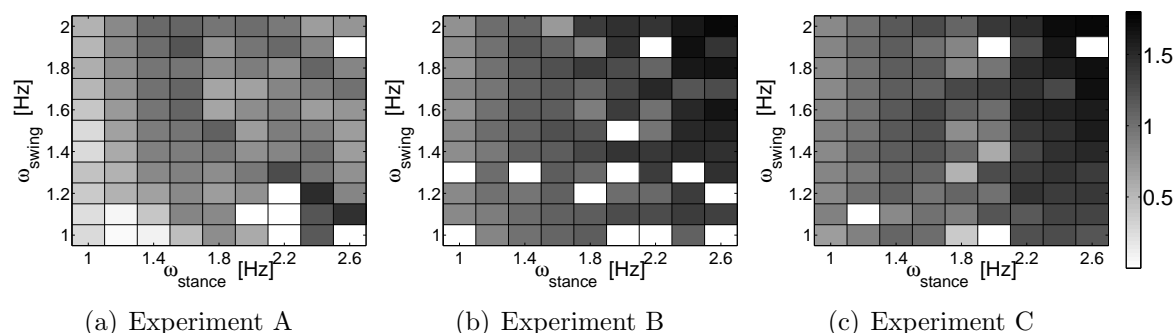


Figure 4.24: We show the locomotion speed of the Ghostdog for different values of  $\omega_{stance}$  and  $\omega_{swing}$ . The values of the parameters are shown in Hz (in Equation (4.17) they are expressed in  $\text{rad} \cdot \text{s}^{-1}$ ). The locomotion speed is expressed in  $\text{body length} \cdot \text{s}^{-1}$ . White squares show the experiments where the robot falls.

C). When feedback is enabled (Exp. B and C), however, this correlation is much lower than the one with  $\omega_{stance}$ .

We can draw several conclusions from these experiments. First there is a strong linear correlation between speed of locomotion and  $\omega_{stance}$  while the correlation with  $\omega_{swing}$  is always smaller and not always significant. It seems that we need to set  $\omega_{swing}$  to a value high enough to have good locomotion (at least in the case without feedback with the Aibo and Ghostdog). Therefore this justifies our previous choice of independent control of these two values in a specially designed oscillator, it is then a good idea to fix  $\omega_{swing}$  and use  $\omega_{stance}$  to control speed.

Although the controller without feedback has good performances in many parameter configurations with the different robots, it seems that  $\omega_{swing}$  should be small enough for good performances. When the swing duration is too long then a swinging limb will touch the ground before it has finished the swing and will then continue pushing forward and create undesirable friction with the ground. If the swing is too short, then it will start the stance phase before the limb hits the ground, which can lead to unstable behaviors. When feedback is enabled, this does not happen anymore and the state of the CPG is always coherent with the state of the robot. Another problem with the system without feedback, related to the coherence between the dynamic state of the robot and the CPG is at onset of locomotion. Indeed, without feedback, the commands sent by the CPG are not always compatible with the state of the robot, this leads to high movements of the body (possibly unstable movements) and to hard limb contacts with the ground.

Since the use of feedback tightly couples the controller with the robot, what can be the utility of having coupling between the oscillators? From our experiments, we saw that without coupling the performances of the robot were not as good as when the oscillators are coupled, either in term of speed of locomotion (Aibo and iCub) or stability (GD).

## 4.2. Inclusion of sensory feedback

---

It seems that the coupling (which forces the robot to adopt a desired gait) gives some drive to the robot to move faster and to keep a coordination between the limbs that is efficient for stable locomotion (if we consider that gaits give some kind of meaningful coordination pattern for locomotion). The implicit coupling between the limbs through the robot mechanics is not sufficient.

The CPG without coupling and the CPG without feedback can be seen as two extremities of a continuum of possible controllers and we see that a good controller needs both aspects: the feedback to have coherence between the dynamic state of the robot and the controller and the coupling to keep good coordination between the limbs. However it is still an open question to know what should be the importance of the coupling strength to balance feedback effects. We can guess that it would depend on the type of gait used (i.e. fast running gaits or slow walking gaits).

In terms of stability, we measured for each robot the vertical displacement of its center of mass and the pitch and roll angles of the body of the robots (data not shown). It turns out that sensory feedback decreases the amplitude of the vertical displacement for the three robots, the most obvious decrease being in the case of the GD (standard deviation of vertical displacement is 20% lower in the case of feedback than without feedback). However, when measuring the roll and pitch angles the differences between controllers with feedback and without depend on the parameters used and in all the cases there is no significant differences that would show that one controller makes the robot more stable than the other. For the iCub and Aibo robot, since they are very stable intrinsically, we cannot say much. However for the GD, from our experiments the robot falls less when feedback and coupling is enabled. In order to assess that the feedback system increases the stability of the robots, we make experiments in non flat terrains.

### Non flat terrains

We made tests on terrains with a slope, to test the effect of a transition from two flat surfaces of different slopes and to test the effect of changing the dynamics of the robot (by changing the orientation of gravity) on the controller. The slopes are ranging from 0 to 15 degrees and are ascending and descending slopes. The simulation's world is composed of 3 planes, the robot starts on a terrain with a slope of 0 degrees, one meter further there is a plane with a constant slope (either increasing or decreasing) which is 5 meters long. At the end of the slope a new surface with 0 degree slope starts. We tested both the complete controller with feedback and the controller without feedback on the three robots with different  $\omega_{swing}$  and  $\omega_{stance}$  parameters on these terrains.

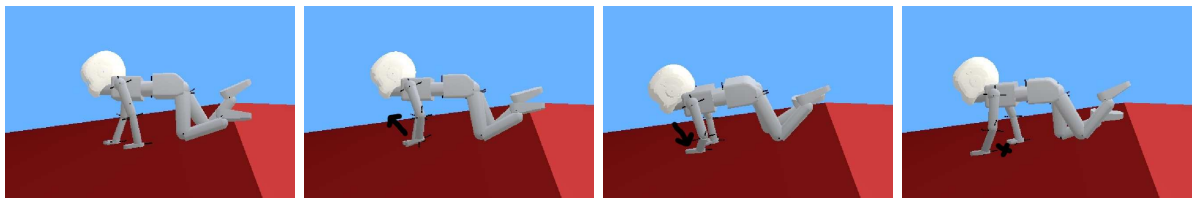
For the Aibo and iCub robots, we do not see much differences between the controller with feedback and the one without in terms of stability. Indeed both robots are very stable. However, there is a qualitative difference between the controllers. Indeed the controller without feedback is not always coherent with the state of the robot and that makes some differences. For example on a descending slope, the fore limbs of the iCub

touch the ground before the end of swing in some parameter configurations, however the limb continues to move forward and it makes the robot slip on the ground and creates undesirable internal torques and friction. Figure 4.25 shows such an example. We see on these figures that when there is not feedback the state of the CPG is not coherent with the information from the force sensors. In the case of a CPG with feedback, the state of the CPG is always coherent with the state of the robot, it then reduces the overall forces exerted on the limbs right after touch down. One problem that appears in both controllers when climbing up slopes is that the robots tend to slip on the floor and turn. It seems that it comes from the fact that these robots are very rigid and when several feet are on the ground, it creates rigid kinematic chains that make the robots slip. In this thesis we kept the controller as simple as possible to understand the basic properties of the CPG (which mainly generates the inter-limb coordination). We think that this problem can be solved on a lower level of control, where we could for example use the redundancy of the robots to implement a whole body controller and use the CPG on top of it.

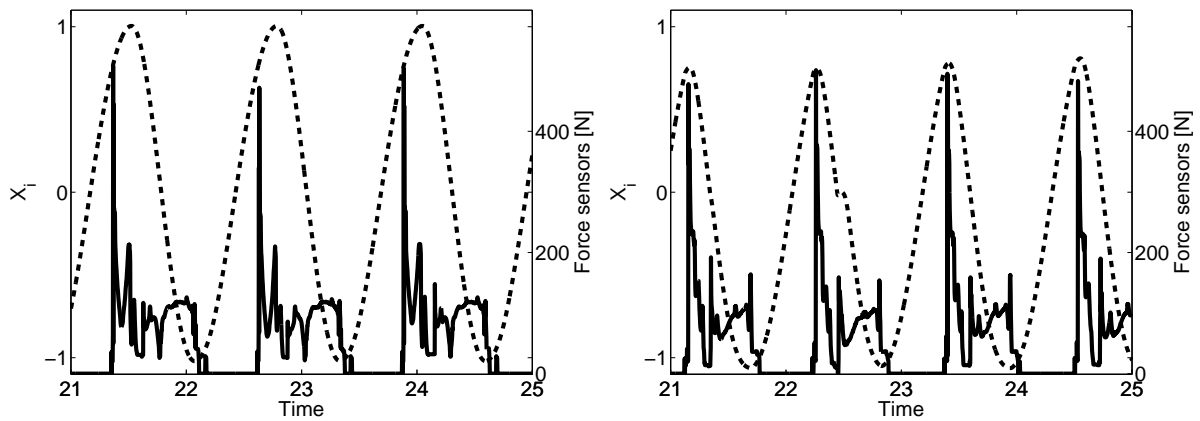
We do not have these problems with the GD robot because the linear springs add compliance that automatically compensate for such constraints. We obtained very interesting results for this robot. Since the robot has passive dynamics and is not intrinsically very stable, on ascending slopes, the effect of feedback is really visible. For a slope of 5 degrees the controller without feedback has really poor performances, since the limbs do not touch the slope at the right time the robot slips and turns (this behavior is very different from the one we observed in the Aibo and iCub robots). In comparison the feedback controller has a very good performance. In Figure 4.26(a), we have plotted the trajectories of the robot for the different trials. When the slope goes to 10 degrees the performances of the controller without feedback are really poor (Figure 4.26(b)). The problem is that the CPG is not at all coordinated with the limbs of the robots on the slope and thus the robot slips on the slope and cannot climb up. In the case with feedback we do not have this problem. The robot is not going always straight, but it is mainly due to the fact that the movements of the limbs are not symmetric because of the feedback. A steering controller would be sufficient to make the robot go up straight, while it would not work for the controller without feedback since it would not correct the coordination of the limbs. The effects are even more obvious in the case of slopes of 15 degrees (Figure 4.26(c)). In that case the feedback controllers start to have problems, and the robot often falls on its side, but still it manages to climb quite a lot of the slope and in some configurations it manages to climb the complete slope, while it never goes up in the case without feedback. In this case we also see the limitations of the controller with feedback for stabilizing the robot and we would need some feedback to also compensate for the rolling motion of the robot. For descending slopes both controllers work well, although with the controller with feedback the state of the CPG is always coherent with the dynamics of the robot (as in the case of the iCub and Aibo).

## 4.2. Inclusion of sensory feedback

---



(a) Typical swing phase without feedback



(b) Force sensor and CPG state (no feedback)

(c) Force sensor and CPG state (feedback)

Figure 4.25: In the upper figure we show a typical swing phase of the iCub robot on a slope when there is not feedback, we notice that after touch down, the limb continues to move forward. The lower figures show the relation between the CPG outputs and the force sensors in the case of a CPG with feedback (right graph) and one without (left graph).

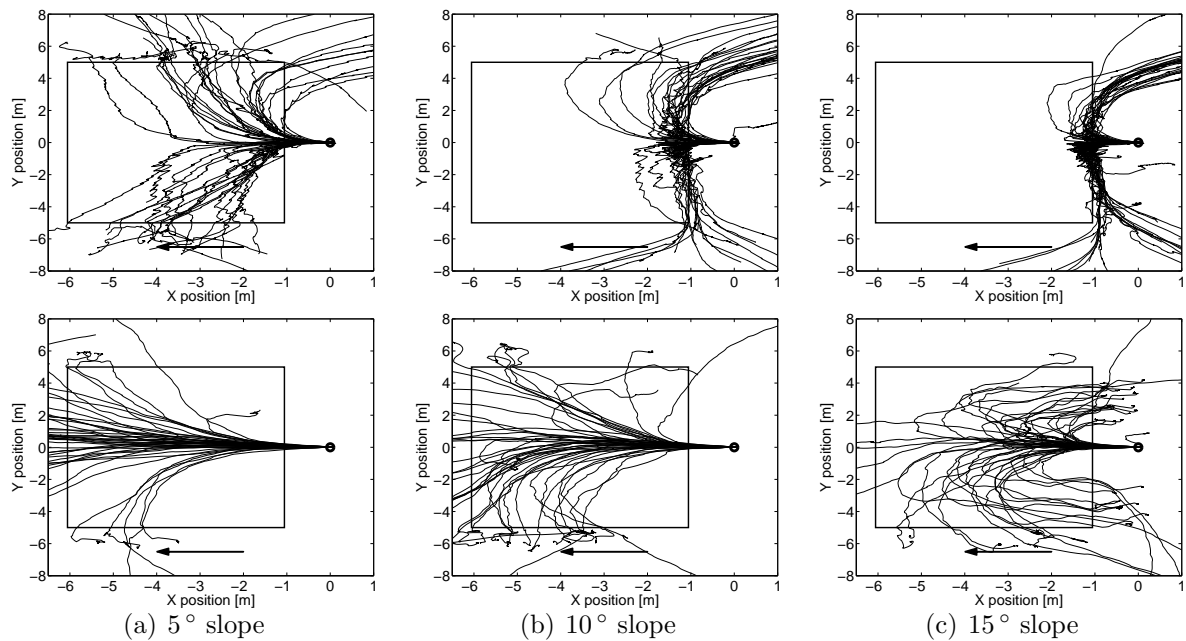


Figure 4.26: Trajectories of the GD robot on different slopes with feedback (lower graphs) and without feedback (upper graphs). We made tests for configurations corresponding to parameters in the range  $2.8\pi < \omega_{swing} < 3.6\pi$  and  $2\pi < \omega_{stance} < 5.2\pi$  which corresponds to the best parameter sets for both controllers with and without feedback. The black circles show the initial position of the robot, the square shows the slope on the terrain and the arrow shows the climbing direction.



### 4.3. Conclusion

---

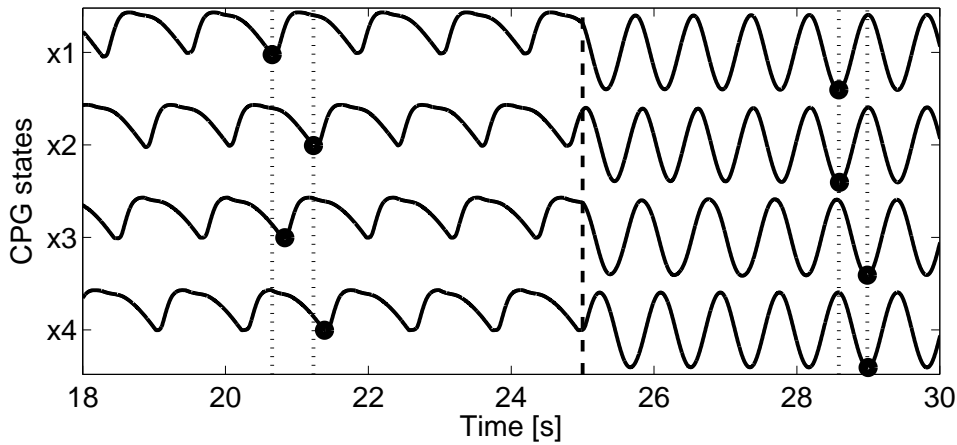


Figure 4.27: Example of an automatic gait transition for the Ghostdog robot with a CPG with feedback. The CPG is coupled to generate a walk. At  $t = 25$  we change  $\omega_{stance}$  from  $0.8\pi$  to  $2.4\pi$ . During the whole experiment  $\omega_{swing} = 2.4\pi$ .

#### Gait transitions

The choice of the gait mainly depends on the properties of the robot, indeed the bound gait would not be possible for the iCub and Aibo robots since they are very rigid and cannot adopt a fast gait with possibly flying phases. For these robots, we successfully tested the walk and trot gaits. For the GD, we successfully tested the walk, trot, pace and bound gaits. However these gaits are not stable for every parameter combination (the bound gait is the gait that has the biggest region of working parameters). It seems normal since different gaits should be used at different speeds. An interesting effect of the feedback controller, when a parameter set is chosen is that it can automatically switch from one gait to another even if the CPG couplings do not specify this gait. The feedback can then force the CPG to adopt a gait that is more adapted to the dynamics of the robot. In Figure 4.27 we show an example of such an automatic gait transition (without changing the CPG couplings). The GD robot starts in a walking gait, the CPG is coupled to produce a walk, but at time  $t = 25$ s we suddenly decrease the stance duration (thus increasing the speed of locomotion) and we see that automatically the gait of the robot changes to a bound gait. We also notice that during the walk gait, the robot makes small steps and after the transition the steps are bigger. These automatic gait transitions are observed in all robots.

### 4.3 Conclusion

In this chapter we presented a method to design Central Pattern Generators. We designed an oscillator for which we can independently control the duration of the ascending and

descending phases (i.e. swing and stance durations). Numerical experiments showed the importance to decouple both phases, since stance duration can be used as a control parameter for locomotion speed while swing duration seems more important for good locomotion performances (at least in the case where we do not use feedback) and should be kept constant at a sufficiently low value (i.e.  $\omega_{swing}$  should be high).

Then we proposed generic networks of coupled dynamical systems to generate different gait patterns. By using the theory of symmetric coupled cell networks, the problem was reduced to a purely combinatorial one. We were also able to simply calculate the other possible patterns of oscillations in these networks (and showed that only the desired ones were stable). These networks can be used with any types of oscillators and the method can be used to calculate constraints on networks for any legged robot.

We introduced sensory feedback in the oscillators to modulate the onset of stance and swing. The interest of using our method is that we can explicitly control the behavior of the system. The idea of adding sensory feedback in the CPG to modulate the transitions between swing and stance phases is closely related to the work of Kimura and colleagues in their Tekken robot [46, 75, 76, 77]. Their approach is very much biologically inspired. They use a Matsuoka oscillator [85], which is an oscillator composed of two mutually coupled neuron-like systems, as a building block for their CPG and they include sensory feedback and reflexes to their controller. The main differences between their approach and ours are the following. First their controller includes reflexes (i.e. stumbling response and rolling motion corrective reflexes) which is a direction that we have not yet explored in this thesis. From that point of view, their controller is more advanced. Second they use an oscillator that is more biologically plausible but the drawback is that it is much more complex than the one we use, it has 4 state variables and we cannot control its amplitude, stance and swing frequency as easily as we do. Third the inclusion of feedback pathways is made from an engineering perspective in our approach and we control all the parameters while they use a more biological way of including feedback and the behavior of the system is not so simple to understand. Finally we insist on the importance to have independent control of swing and stance durations to control speed of locomotion, while they do not address this question. It is interesting to compare both works because it shows two effective locomotion controllers based on similar inspirations from biology but designed from two different perspectives.

In our controller, feedback made the locomotion of the robots more robust to parameter choices and uncertain environments. It is interesting to note that we so far use only local information at the limb level to modulate the behavior of the oscillators but it is sufficient to produce stable locomotion on different terrains. It seems that coherent behavior between the CPG and dynamical state of the robot is sufficient to improve locomotion. The results were particularly obvious in the case of the GD robot which had passive dynamics. This remark may emphasize the importance of compliant robots for locomotion control since compliance can simplify greatly the control. Moreover, CPGs seem particularly well adapted to control this kind of robots since they can exploit their

### 4.3. Conclusion

---

intrinsic dynamics. This observation supports previous claims on the usefulness of robots with passive dynamics such as [15, 64].

One interesting aspect for this controller is that it is extremely simple, it is composed of 4 first order differential equations for each limb and uses very simple sensory feedback, thus it is computationally very efficient and could be easily implemented on simple micro controllers in a robot (and this could be done on distributed hardware).

The proposed framework also allows the integration of more complex behaviors, for example it was shown in [30, 31] that it is possible to do precise hand placement by adding discrete movement primitives on top of this CPG.

However it is clear that many problems still exist. For example for rigid robots, kinematic chains created by several limbs touching the ground could be problematic. One possible solution to this issue would be to use a force control framework instead of high gain PID controllers, for example whole body control methods as the ones proposed in [128] or [62]. Another problem is to improve the control of stability, especially to control the rolling motion of the robot, as we have seen for the GD robot.

Future work will include solving the previously mentioned issues as well as going to real world experiments. In order to do so, a robot resembling the GD robot was recently designed [120] and the CPG without feedback was successfully used to control it. We will investigate the inclusion of reflex pathways (e.g. stumbling response reflex) and more global sensory information (e.g. vestibular information) in the CPG and we will also test the controller on more difficult terrains in order to assess the stability of the locomotion.



---

## CHAPTER 5

# ADAPTIVE FREQUENCY OSCILLATORS

---

In this chapter we present our work on Adaptive Frequency Oscillators (AFOs). It is a general mechanism that we developed to transform an oscillator into an oscillator that can adapt its parameters to learn the frequency of any periodic signal. A key concept in this mechanism is that we transform the parameter to adapt into a new state variable of the adaptive oscillator. The adaptation is then part of the dynamics of the oscillator. This illustrates well the concept of having no separation between the learning algorithm and the learning substrate that we explained in Chapter 2. CPG-based controllers can benefit from this mechanism in two ways, first it can be used to tune the controller to the resonant frequency of the robot and second we can build programmable CPGs that can learn specific limit cycle trajectories.

The genesis of this mechanism and its applications comes from a very fruitful collaboration with Dr. Jonas Buchli. Everything started during my Master thesis conducted at the BIRG under the supervision of Professor Auke Ijspeert and Jonas Buchli (who was a PhD student in the laboratory at that time). One goal of my project was to find a mechanism such that a Hopf oscillator could adapt its frequency to the frequency of a periodic signal. I came up with a discrete map version of frequency adaptation [108]. Then Jonas developed the continuous version of the adaptation mechanism in [16], which is the actual version that we discuss in this chapter. He used this version to propose an adaptive controller able to find the resonant frequency of a mechanical system. It was successfully used for the locomotion control of robots with passive dynamics (see for example [14, 15, 17, 16]). I then joined the BIRG as a PhD student and the collaboration with Jonas led to the development of many aspects of the theory of AFOs. All the work presented here are my personal contributions to this theory.

The next two sections present the theoretical work on AFOs, the third one an application to signal processing from a dynamical systems perspective. In the fourth section we show how this mechanism can be used to build programmable CPGs that encode into a limit cycle any periodic pattern.

## 5.1 Frequency adaptation mechanism

This section presents the general mechanism for frequency adaptation. We show the convergence of the frequency adaptation for harmonic oscillators and weak coupling and we generalize the mechanism to more complex oscillators, such as relaxation types and strange attractors. This work was originally published in [110].

### 5.1.1 Motivation

Oscillators are used increasingly in sciences, for both modeling and engineering purposes. They are well suited for applications that involve synchronization with periodic signals. Models of Josephson junctions [134], lasers, central pattern generators (CPGs) [27, 52, 80, 138], associative memories [13, 99] or beat perception [37, 82] are a few examples that show the importance of oscillators in modeling and control.

Oscillator models are interesting because of their synchronization capabilities, either with other oscillators or with external driving signals. However, since they traditionally have a fixed intrinsic frequency, two main limitations arise. First their synchronization properties are limited in the sense that they can synchronize only with signals with close enough frequencies (they have a finite entrainment basin). Although this entrainment basin depends on the coupling strength with the signal to be synchronized to and can be made arbitrarily wide (at least for simple types of oscillators) this entrainment basin will always be finite. Second, they have no memory of past interactions, if the signal to which they were synchronized disappears, they return to their original frequency of oscillation.

Consequently when one wants to model systems that have unlimited synchronization capabilities and where past interactions (i.e. memory) plays an important role, these models are not well adapted. People have postulated that some biological oscillators have a mechanism to adapt their intrinsic frequencies, for example to explain the synchronization phenomena of some species of fireflies [42] or to explain how the neural pattern generators that control the locomotion of animals can adapt to a body that changes dramatically during the development of the animal. Moreover, for engineering applications, one would like to have flexibility in setting the parameters of the oscillator to have synchronization with zero phase delay without wondering about the entrainment basins. Continuous interactions with signals of the environment would change the parameters of the oscillator such that they correspond to the mode of operation of the engineered system (e.g. the resonant frequency of a mechanical system).

Some recent studies, however, concentrate on developing dynamic plasticity for oscillators, so they can learn and synchronize with a wider range of frequencies, without having to tune the parameters by hand [4, 13, 42, 95, 97, 98]. But these attempts are so far limited to very simple classes of oscillators, equivalent to phase oscillators, mainly because this is the only class of oscillators that can be analytically studied and for which convergence can be proved, when adding adaptivity to the system. Adaptive relaxation

## 5.1. Frequency adaptation mechanism

---

oscillators were also developed to model rhythm perception [37]. These oscillators are able to adapt their frequencies to synchronize with external input. But these input signals are simple and reduce to periodic pulse trains.

In this section, we present the general mechanism of adaptive frequency oscillators. First we show the convergence of the adaptive Hopf oscillator and then we generalize the adaptive rule for more complex oscillators so they can learn the frequencies of, and synchronize with, any rhythmic input signal. An interesting property of our method is that we go beyond phase-locking of oscillations. We add plasticity to the system, in the sense that the system can change its own parameters in order to learn the frequencies of the periodic input signals. So the range of frequencies that can be learned is not limited and after learning the oscillator continues to oscillate at the learned frequency, even if the input signal disappears. We call our adaptive mechanism<sup>1</sup> *dynamic Hebbian learning* because it shares similarities with correlation-based learning observed in neural networks [73].

One major aspect of our approach is that an oscillator learns the frequency of any periodic input, without any signal processing. It means that an oscillator can adapt its frequency to any kind of periodic, or even pseudo-periodic, input. The process is completely dynamic, and does not require the specification of time windows or similar free parameters as it is often the case in signal processing algorithms. The whole learning process and the frequency extraction from the input is totally embedded in the dynamics of the system. Another interesting property of the method is that we can directly apply it to many kinds of oscillators, for example relaxation oscillators or strange attractors. An oscillator, perturbed by a periodic signal  $F$ , is described by the general equations

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

with  $\omega$  some parameter that has a monotonic relation with the frequency of the oscillations (not necessarily linear). We introduce a learning rule for this parameter

$$\dot{\omega} = \pm \epsilon F \frac{y}{\sqrt{x^2 + y^2}}$$

The sign depends on the direction of rotation of the limit cycle in the  $(x, y)$  phase space. This general adaptation rule works for many different oscillators,  $\omega$  will converge to a value such that one frequency component of the oscillator and one of the input  $F$  match. We discuss this general learning rule in the following.

In Section 5.1.2, we first present the adaptive learning rule with a simple Hopf oscillator and show the convergence and the stability of the whole system for weak coupling. Then, in Section 5.1.3, we present some numerical simulations, to show that the oscillator can adapt its frequency to the frequency of any kind of periodic or pseudo-periodic

---

<sup>1</sup>In this chapter, we use adaptation and learning as synonyms

signals. Finally, in order to demonstrate the generality of our method, we construct, in Section 5.1.4, an adaptive Van der Pol oscillator which we discuss in details. We also present examples of frequency adaptation with an adaptive Rayleigh oscillator, an adaptive Fitzhugh-Nagumo oscillator and an adaptive Rössler system.

### 5.1.2 Learning frequencies with a Hopf oscillator

In this section, we introduce and discuss the learning rule for frequency adaptation in oscillators. To keep discussion as simple as possible, we use a Hopf oscillator to discuss our learning method, because its phase evolution is simple to describe. Generalization to more complex oscillators will be presented in further sections. We first present the model, then we show the convergence of the adaptive dynamical system.

#### Model description

**The Hopf oscillator** The dynamics of the Hopf oscillator is governed by the following differential equations

$$\dot{x} = (\mu - r^2)x - \omega y + \epsilon F \tag{5.1}$$

$$\dot{y} = (\mu - r^2)y + \omega x \tag{5.2}$$

Where  $r = \sqrt{x^2 + y^2}$ ,  $\mu > 0$  controls the amplitude of the oscillations and  $\omega$  is the intrinsic frequency of the oscillator. It means that without perturbations (when  $\epsilon = 0$ ), the system is oscillating at  $\omega \text{ rad} \cdot \text{s}^{-1}$ . This oscillator is coupled with a periodic force  $F$ . When the force is zero, the system has an asymptotically stable harmonic limit cycle, with radius  $\sqrt{\mu}$  and frequency  $\omega$ . As the limit cycle of the Hopf oscillator is structurally stable, small perturbations around its limit cycle ( $\epsilon > 0$ ) do not change the general behavior of the system. It means that the limit cycle will still exist, only its form and time-scale will change. Structural stability assures that this change is close to identity.

As we are mainly interested in the phase dynamics, we rewrite the system in polar coordinates. We set  $x = r \cos \phi$  and  $y = r \sin \phi$ . Equations (5.1) and (5.2) transform into

$$\dot{r} = (\mu - r^2)r + \epsilon F \cos \phi \tag{5.3}$$

$$\dot{\phi} = \omega - \frac{\epsilon}{r} F \sin \phi \tag{5.4}$$

It is well known that when the oscillator has its intrinsic frequency  $\omega$  close to one frequency component of the periodic input, it will phase-lock (this phenomena is also called entrainment) [103]. It means that the oscillations synchronize with the frequency of the periodic input. The maximum distance between the intrinsic frequency of the oscillator and the periodic input that still permits phase-locking depends directly on the coupling strength. The stronger the coupling is, the larger the entrainment basin. Outside this



## 5.1. Frequency adaptation mechanism

---

basin, the oscillator is still influenced by the coupling but does not synchronize. If the periodic input has several frequency components, then several entrainment basins will appear. Phase-locking will be possible with each frequency component. Outside the basin, the oscillator will have tendency to accelerate or decelerate, according to the term  $F \sin \phi$ , in average the oscillator will tend to oscillate at a frequency which is between the intrinsic frequency of the oscillator and the frequency of the input. In the case of multi-frequency inputs, these oscillations will be influenced in a similar manner.

**Adaptive dynamical system** Now we can build our adaptation rule by using the influence of the external perturbation on the activity of the oscillator. The adaptation rule will be a dynamical system of the form

$$\dot{\omega} = f(\omega, r, \phi, F) \quad (5.5)$$

In the following we motivate the concrete choice of the adaptation rule by reasoning about the effects of a perturbation in a geometric way in the phase space of the dynamical system. This provides insights into our choice of the learning rule. In further sections, we will show more rigorously that this reasoning is appropriate and leads to the desired behavior.

To get a good grasp on the effects of perturbations on a limit cycle system (i.e. an oscillator) it is helpful to look at the limit cycle in the phase space representation. In the phase space all perturbations have a direction, i.e. they can be represented as a vector  $\vec{P}$  in that space.

Due to the stability properties of a limit cycle system a perturbation can in the long term only affect the phase of the oscillator. The phase is marginally stable whereas the system is damped perpendicularly to the limit cycle. This means that the phase point always returns to the limit cycle, but it can be phase shifted. In other words the system after a singular perturbation will forget all the perturbation's influence except its influence on the phase.

Especially in a small neighborhood of the limit cycle a small perturbation can only affect the phase strongly if it perturbs the oscillator in the direction tangential to the limit cycle. The perturbations perpendicular to the limit cycle are damped out. The domain where this assumption is valid depends on the coupling of phase and radius. While for the Hopf oscillator this assumption is valid for a very large neighborhood, the neighborhood can be very small for other oscillators, e.g. oscillators with strongly bent isochrones.

To discuss the influence of the perturbation on the phase in this neighborhood, let us introduce a coordinate system with its origin on the phase point. The first base vector  $\vec{e}_r$  is chosen perpendicular to the limit cycle, while the second base vector  $\vec{e}_\phi$  is chosen tangential to the limit cycle (cf. Fig. 5.1). Thus, this coordinate system rotates with the phase point along the limit cycle. In order to know the influence  $p_\phi = |\vec{p}_\phi|$  of the

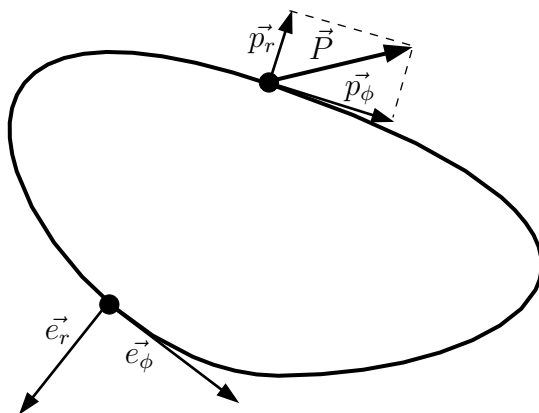


Figure 5.1: We illustrate the coordinate system in which synchronization is most naturally discussed. The figure shows an arbitrary limit cycle. The system is strongly damped in direction perpendicular to the limit cycle  $\vec{e}_r$  and marginally stable in direction tangential to the limit cycle  $\vec{e}_\phi$ . This is the reason for the structurally stable limit cycle in the first place and allows for a resetting of the phase on the other hand. Note that the 2-dimensional representation is always valid for discussing a limit cycle since there exists always a 2 dimensional manifold which contains the limit cycle. Refer to text for a discussion of the perturbation  $\vec{P}$ .

perturbation on the phase it is sufficient to project  $\vec{P}$  on  $\vec{e}_\phi$

$$p_\phi = \vec{P} \cdot \vec{e}_\phi \quad (5.6)$$

Thus, depending on the external perturbation *and* the state of the oscillator (i.e. the position of the point on the limit cycle) the perturbation accelerates the phase point or slows it down. If the perturbation is a periodic signal, this results in an average acceleration or deceleration depending on the frequency difference. This effect, if the frequency of the oscillator and the external frequency are close, leads to well known phase-locking behavior. Thus, the influence carries the information needed to adjust to the frequency of the external perturbation. Consequently, if we take this same effect to tune the frequency of the oscillator (on a slower time scale) the frequency should evolve toward the frequency of the perturbation. Therefore, the effect of  $f(\omega, r, \phi, F)$  on  $\omega$  has to be the same as the effect of the perturbation on the phase, thus, (in average) driving  $\omega$  toward the frequency of the perturbation.

While the discussion here is valid for limit cycles of any form and in any dimension, in the case of the Hopf oscillator and the perturbation as chosen in Eqs.(5.3) and (5.4) it is evident that  $p_\phi = \frac{\epsilon}{r} F \sin \phi$ . We chose accordingly

$$\dot{\omega} = -\epsilon F \sin \phi \quad (5.7)$$

## 5.1. Frequency adaptation mechanism

---

which corresponds in Cartesian coordinates to

$$\dot{\omega} = -\epsilon F \frac{y}{\sqrt{x^2 + y^2}} \quad (5.8)$$

The adaptation of  $\omega$  happens on a slower time scale than the evolution of the rest of the system. This adaptation time scale is influenced by the choice of  $\epsilon$ . Note that the  $r$  variable is dropped because we do not want a learning rule which is scaled by the amplitude of the oscillations. With this rule, the oscillator will adapt to the frequency of any input signal. As in applications most signals will be non-harmonic, i.e. they have several frequency components, the oscillator will adapt to one of these components, generally the closest to the intrinsic frequency of the oscillator. We must also note that it is required to keep the oscillator coupled with the input, because it is the evolution of  $\phi(t)$ , i.e. change of frequency correlated with  $\dot{\omega}$ , that enables adaptation in Equation (5.7). The convergence of this adaptive oscillator (Eqs (5.3), (5.4) and (5.7)) in the general case of multi-frequency inputs is shown in the next section.

### Proof of convergence with the Hopf oscillator

In this section we show the stability of the adaptive Hopf oscillator, but we will see in next sections that the results we derive in this section can also justify convergence for other types of oscillators. The new dynamical system we study is the one composed of the oscillator and its learning rule for the frequency (Eqs. (5.3), (5.4) and (5.7)). As long as  $\omega > 0$ , because of structural stability, the behavior of the oscillator (Eqs. (5.3) and (5.4)) is known, so we just have to show that  $\omega$  converges to the desired input frequency.

We use perturbation methods (cf.[74]) to discuss the convergence of the system. The solution of the system  $\{r(t), \phi(t), \omega(t)\}$  can be written as a perturbation series, with  $\epsilon < 1$

$$r(t) = r_0 + \epsilon r_1 + \epsilon^2 r_2 + \epsilon^3 R_r \quad (5.9)$$

$$\phi(t) = \phi_0 + \epsilon \phi_1 + \epsilon^2 \phi_2 + \epsilon^3 R_\phi \quad (5.10)$$

$$\omega(t) = \omega_0 + \epsilon \omega_1 + \epsilon^2 \omega_2 + \epsilon^3 R_\omega \quad (5.11)$$

with initial conditions  $r_0(t_0) = r_0$ ,  $\phi_0(t_0) = 0$  and  $\omega_0(t_0) = \omega_0$  independent of  $\epsilon$ . Here,  $r_i$ ,  $\phi_i$  and  $\omega_i$  are functions of time and  $R_r$ ,  $R_\omega$  and  $R_\phi$  are small residues of the order  $\epsilon^3$ . Which means there exists a constant  $k$  such that  $R_i < k$ , generally  $k$  is small. The following proof will hold under the hypothesis that  $k \ll 1$ , numerical simulations in Section 5.1.3 will confirm this hypothesis. We could also expand the perturbation series to higher order: the finer the approximation is, the wider the time interval valid for the approximation. But we will show that convergence appears on the time-scale associated with the second order approximation. By inserting Eqs. (5.9), (5.10) and (5.11) into Eqs.

(5.3), (5.4) and (5.7), and by observing that

$$\begin{aligned} \sin(\phi_0 + \epsilon\phi_1 + \epsilon^2\phi_2 + \epsilon^3R_\phi) &= \sum_{k=0}^{\infty} \frac{(-1)^k (\phi_0 + \epsilon\phi_1 + \epsilon^2\phi_2 + \epsilon^3R_\phi)^{2k+1}}{(2k+1)!} \\ &= \sin\phi_0 + \epsilon\phi_1 \cos\phi_0 + O(\epsilon^2) \end{aligned} \quad (5.12)$$

and similarly that

$$\cos(\phi_0 + \epsilon\phi_1 + \epsilon^2\phi_2 + \epsilon^3R_\phi) = \cos(\phi_0) - \epsilon\phi_1 \sin(\phi_0) + O(\epsilon^2) \quad (5.13)$$

we can identify the terms corresponding to each  $\epsilon^n$  and derive the following differential equations

$$\dot{r}_0 = (\mu - r_0^2)r_0 \quad (5.14)$$

$$\dot{\phi}_0 = \omega_0 \quad (5.15)$$

$$\dot{\omega}_0 = 0 \quad (5.16)$$

$$\dot{r}_1 = \mu r_1 - 3r_1 r_0^2 + F \cos\phi_0 \quad (5.17)$$

$$\dot{\phi}_1 = \omega_1 - \frac{1}{r_0}(r_1 \dot{\phi}_0 - r_1 \omega_0 + F \sin\phi_0) \quad (5.18)$$

$$\dot{\omega}_1 = -F \sin\phi_0 \quad (5.19)$$

$$\dot{r}_2 = \mu r_2 - 3r_2 r_0^2 - r_2 r_1^2 - F\phi_1 \cos\phi_0 \quad (5.20)$$

$$\dot{\phi}_2 = \omega_2 - \frac{1}{r_0}(r_1 \dot{\phi}_1 - r_1 \omega_1 + r_2 \dot{\phi}_0 - r_2 \omega_0 + F\phi_1 \cos\phi_0) \quad (5.21)$$

$$\dot{\omega}_2 = -F\phi_1 \cos\phi_0 \quad (5.22)$$

with initial conditions  $r_0(t_0) = \sqrt{\mu}$ ,  $\phi_0(t_0) = 0$ ,  $\omega_0(t_0) = \omega_0$  and  $r_i(t_0) = \phi_i(t_0) = \omega_i(t_0) = 0$ ,  $\forall i = 1, 2$ . We consider that the unperturbed system ( $i = 0$ ) has already converged to the limit cycle and that at time  $t_0$ , there is no perturbations. We have to solve Equations (5.16), (5.19) and (5.22) to construct an approximate solution of Equation (5.7) and thus show the convergence properties of the adaptation rule  $\omega$ . The behavior of the two other state variables is already known since the Hopf oscillator has a structurally stable limit cycle. In order to solve these equations we also have to solve Equations (5.14), (5.15) and (5.18). The error of the approximation will be of order  $O(\epsilon^3)$  and will hold for some time interval  $[t_0, t_0 + \sigma]$ . The solutions of Equations (5.14)-(5.16) are straightforward

$$r_0(t) = \sqrt{\mu} \quad (5.23)$$

$$\phi_0(t) = \omega_0(t - t_0) \quad (5.24)$$

$$\omega_0(t) = \omega_0 \quad (5.25)$$

## 5.1. Frequency adaptation mechanism

---

To solve the other equations, we first rewrite the periodic input as its complex Fourier series

$$F(t) = \sum_{n=-\infty}^{\infty} A_n e^{in\omega_F t} \quad (5.26)$$

Where  $\omega_F$  is the frequency of the input. We now consider the case where  $\omega_0 \neq n\omega_F$ ,  $\forall n \in \mathbb{N}$ , which means that at the beginning the system is not synchronized with any frequency component of the periodic input  $F$ . We then get

$$\begin{aligned} \dot{\omega}_1 &= - \left( \sum_{n=-\infty}^{\infty} A_n e^{in\omega_F t} \right) \sin(\omega_0(t - t_0)) \\ &= - \sum_{n=-\infty}^{\infty} A_n \frac{e^{i(n\omega_F + \omega_0)t - i\omega_0 t_0} - e^{i(n\omega_F - \omega_0)t + i\omega_0 t_0}}{2i} \end{aligned} \quad (5.27)$$

Which solves into

$$\omega_1(t) = \frac{1}{2} \sum_{n=-\infty}^{\infty} A_n \left( \frac{- (e^{i(n\omega_F - \omega_0)t + i\omega_0 t_0} - e^{in\omega_F t_0})}{(n\omega_F - \omega_0)} + \frac{(e^{i(n\omega_F + \omega_0)t - i\omega_0 t_0} - e^{in\omega_F t_0})}{(n\omega_F + \omega_0)} \right) \quad (5.28)$$

and

$$\dot{\phi}_1 = \omega_1 + \frac{\dot{\omega}_1}{\sqrt{\mu}} \quad (5.29)$$

which solves into

$$\begin{aligned} \phi_1(t) &= \frac{\omega_1(t)}{\sqrt{\mu}} + \frac{1}{2} \sum_{n=-\infty}^{\infty} A_n \left( \frac{(e^{i(n\omega_F + \omega_0)t - i\omega_0 t_0} - e^{in\omega_F t_0})}{i(n\omega_F + \omega_0)^2} + \right. \\ &\quad \left. \frac{2\omega_0(t - t_0)e^{in\omega_F t_0}}{n^2\omega_F^2 - \omega_0^2} - \frac{(e^{i(n\omega_F - \omega_0)t + i\omega_0 t_0} - e^{in\omega_F t_0})}{i(n\omega_F - \omega_0)^2} \right) \end{aligned} \quad (5.30)$$

By combining Equations (5.25) and (5.28), we have a first order approximation  $\omega(t) = \omega_0 + \epsilon\omega_1(t) + \epsilon^2 R_\omega$ . This approximation is a periodic solution with mean equal  $\omega_0$ . Nevertheless, this first order approximation does not show any adaptation of  $\omega(t)$ . This seems normal, since we argued before that the learning takes place on a larger time-scale than the perturbation (which is of order  $\epsilon$ ). We now derive the second order approximation to show that learning appears on the associated time-scale. As we are interested in the second order form of  $\omega$ , we now solve Equation (5.22)

$$\begin{aligned} \dot{\omega}_2 &= - \left( \sum_{m=-\infty}^{\infty} A_m e^{im\omega_F t} \right) \left( \frac{e^{i\omega_0(t-t_0)} + e^{-i\omega_0(t-t_0)}}{2} \right) \phi_1(t) \\ &= - \frac{1}{2} \left( \sum_{m=-\infty}^{\infty} A_m (e^{i(m\omega_F + \omega_0)t - i\omega_0 t_0} + e^{i(m\omega_F - \omega_0)t + i\omega_0 t_0}) \right) \phi_1(t) \end{aligned} \quad (5.31)$$

By expanding the equation we find a sum of simpler terms that can be easily integrated

$$\omega_2 = \int_{t_0}^t \left( \frac{1}{4} \sum_{m,n \in \mathbb{Z}} A_m A_n (E_1 + E_2 + E_3 + E_4 + E_5 + E_6) \right) \quad (5.32)$$

where

$$\begin{aligned} E_1 &= e^{i((m+n)\omega_F + 2\omega_0)t - 2i\omega_0 t_0} \left( \frac{-1}{\sqrt{\mu}(n\omega_F + \omega_0)} - \frac{1}{i(n\omega_F + \omega_0)^2} \right) \\ E_2 &= e^{i((m+n)\omega_F - \omega_0)t + 2i\omega_0 t_0} \left( \frac{1}{\sqrt{\mu}(n\omega_F - \omega_0)} + \frac{1}{i(n\omega_F - \omega_0)^2} \right) \\ E_3 &= e^{i(m\omega_F + \omega_0)t + i(n\omega_F - \omega_0)t_0} \left( \frac{-2\omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} - \frac{4n\omega_F\omega_0}{i((n\omega_F)^2 - \omega_0^2)^2} \right) \\ E_4 &= e^{i(m\omega_F - \omega_0)t + i(n\omega_F + \omega_0)t_0} \left( \frac{-2\omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} - \frac{4n\omega_F\omega_0}{i((n\omega_F)^2 - \omega_0^2)^2} \right) \\ E_5 &= e^{i(m+n)\omega_F t} \left( \frac{2\omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} + \frac{4n\omega_F\omega_0}{i((n\omega_F)^2 - \omega_0^2)^2} \right) \\ E_6 &= (e^{i\omega_0(t-t_0)} + e^{-i\omega_0(t-t_0)}) \left( \frac{-2\omega_0}{(n\omega_F)^2 - \omega_0^2} \right) e^{i(m\omega_F t + n\omega_F t_0)} (t - t_0) \end{aligned}$$

Prior, we postulated that  $\omega_0 \neq n\omega_F, \forall n \in \mathbb{N}$ , consequently, the integration of  $E_1, E_2, E_3$  and  $E_4$  gives periodic functions with zero mean. The integration of  $E_6$  gives a function oscillating with some frequency but with its amplitude varying because of the  $t$  term, the average contribution of this function is zero. The integration of  $E_5$  is more interesting because when  $n = -m$ , the exponential disappears and we have a constant instead. Thus when integrating we will find linear terms. For the case  $m \neq -n$ , after integration, we find a periodic function with zero mean. Therefore,  $\omega_2(t)$  is composed of a periodic function  $\tilde{\omega}_2(t)$  with zero mean and a deviation  $D_\omega(t)$ .

$$\omega_2(t) = \tilde{\omega}_2(t) + D_\omega(t) \quad (5.33)$$

where

$$\begin{aligned} D_\omega(t) &= \int_{t_0}^t \frac{1}{4} \sum_{\substack{n \in \mathbb{Z} \\ m = -n}} A_n A_m \left( \frac{2\omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} - \frac{4n\omega_F\omega_0}{i((n\omega_F)^2 - \omega_0^2)^2} \right) \\ &= \int_{t_0}^t \left( \frac{-A_0}{2\sqrt{\mu}\omega_0} + \sum_{n \in \mathbb{N}^*} \frac{A_n \bar{A}_n \omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} \right) \\ &= \left( \frac{-A_0}{2\sqrt{\mu}\omega_0} + \sum_{n \in \mathbb{N}^*} \frac{|A_n|^2 \omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} \right) (t - t_0) \end{aligned} \quad (5.34)$$

## 5.1. Frequency adaptation mechanism

---

Then, the solution of  $\omega(t)$  in a neighborhood of  $t_0$  is

$$\omega(t) = \omega_0 + \epsilon\omega_1(t) + \epsilon^2\tilde{\omega}_2(t) + \epsilon^2D_\omega(t) + O(\epsilon^3) \quad (5.35)$$

The solution is composed of small oscillations of amplitude much smaller than  $\epsilon$  around  $\omega_0$  and a slight deviation  $\epsilon^2D_\omega(t)$ . This deviation term determines how the frequency converges to the input frequency. It can also be used to predict the basins of attraction for inputs with several frequency components (cf. Section 5.1.3). For an input signal that has only one frequency in its spectrum, the deviation is obviously done in the direction of this frequency, since  $D_\omega(t) > 0$  when  $\omega_F > \omega_0$  and  $D_\omega(t) < 0$  otherwise. As this approximation is valid for any  $\omega_0$  and any  $t_0$ , i.e. the point in time when we make the approximation is not important, the oscillator will always, in average, change its frequency in the direction of the input frequency. For more complex signals with more than one frequency component, because of the  $(n\omega_F)^2 - \omega_0^2$  term in  $D_\omega$ , the system will just change its frequency according to the distance between its intrinsic frequency  $\omega_0$  and the frequency components of the input. The amplitudes  $A_n$  of the frequency components will also influence this convergence, in the sense that the more intensity a frequency component has, the more it will attract  $\omega(t)$ . Section 5.1.3 shows examples of such convergence. We must also note that the zero frequency (the mean of the periodic signal) can also influence the convergence because of the  $A_0$  term. Thus, if the input signal has a non-zero mean,  $\omega$  could eventually converge to 0 if  $A_0$  has a stronger influence than the other frequency components. In this case, the limit cycle of the Hopf oscillator would bifurcate into a fixed point.

We still have to discuss the case  $\omega_0 = n\omega_F$  for a given  $n \in \mathbb{N}$ . In this case, the oscillator is synchronized with one frequency component of the perturbation. Thus,  $\omega(t)$  oscillates and deviates from  $n\omega_F$ . Then there are two cases, either the deviation becomes an attraction as soon as  $\omega_0 \neq n\omega_F$  and the intrinsic frequency of the oscillator is always staying in a small neighborhood of  $n\omega_F$ . Or  $\omega(t)$  diverges from this frequency and gets attracted by another frequency component of the input signal, with stronger amplitude.

We notice that  $\epsilon$  controls both the amplitude of oscillations around  $n\omega_F$  and the learning rate of the system (proportional to  $\epsilon^2$ ). So the faster the learning is, the higher the error of adaptation will be. But as  $\epsilon < 1$ , the error of adaptation is bounded and small (of the order of  $\epsilon$ ).

So we have shown that the learning rule makes the frequency converge to a frequency component of the input signal, for any initial conditions  $(t_0, \omega_0)$ . The attracting frequency component depends on its distance to the intrinsic frequency of the oscillator and its intensity. The proof is global because we did not make any assumption on the initial condition for  $\omega$  and on the neighborhood of the attracting frequencies.

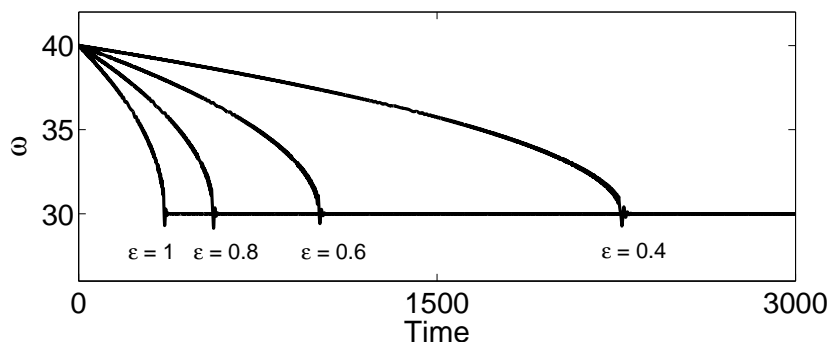


Figure 5.2: Plot of the evolution of  $\omega$  for four different values of  $\epsilon$ . Here we set  $\mu = 1$ ,  $x(0) = 1$  and  $y(0) = 0$ , the perturbing force is  $F = \cos(30t)$ . For every value of  $\epsilon$ , we see that  $\omega$  converges to 30, which is the frequency of the input signal. Therefore, the system is able to learn the frequency of the input signal. We also notice that  $\epsilon$  controls the convergence rate, the higher it is, the faster the system learns.

### 5.1.3 Numerical simulations

The goal of this section is to study the behavior of the learning dynamical system with numerical simulations. First we give a simple example of adaptation of the oscillator receiving a simple periodic signal as input. Then we confirm the proof of Section 5.1.2 by calculating the second order approximation error for a simple example. We also use the analytic results to predict the behavior of the system when varying several parameters. Finally, we show that the system can adapt to pseudo-periodic signals.

#### Simple example of learning

First of all, we want to show a simple example of how the system works and discuss the influence of the learning rate  $\epsilon$ . The adaptive Hopf oscillator is composed of the perturbed Hopf oscillator

$$\dot{x} = (\mu - r^2)x - \omega y + \epsilon F \quad (5.36)$$

$$\dot{y} = (\mu - r^2)y + \omega x \quad (5.37)$$

and of the adaptive frequency learning rule

$$\dot{\omega} = -\epsilon F \frac{y}{\sqrt{x^2 + y^2}} \quad (5.38)$$

Here we use a simple cosine signal  $F = \cos(30t)$  as input, with  $\mu = 1$  and initial conditions  $r(0) = 1$ ,  $\phi(0) = 0$  and  $\omega(0) = 40$ . We integrate the system numerically for several values of  $\epsilon$ , the results of the simulations are shown in Figure 5.2. In this figure,



## 5.1. Frequency adaptation mechanism

---

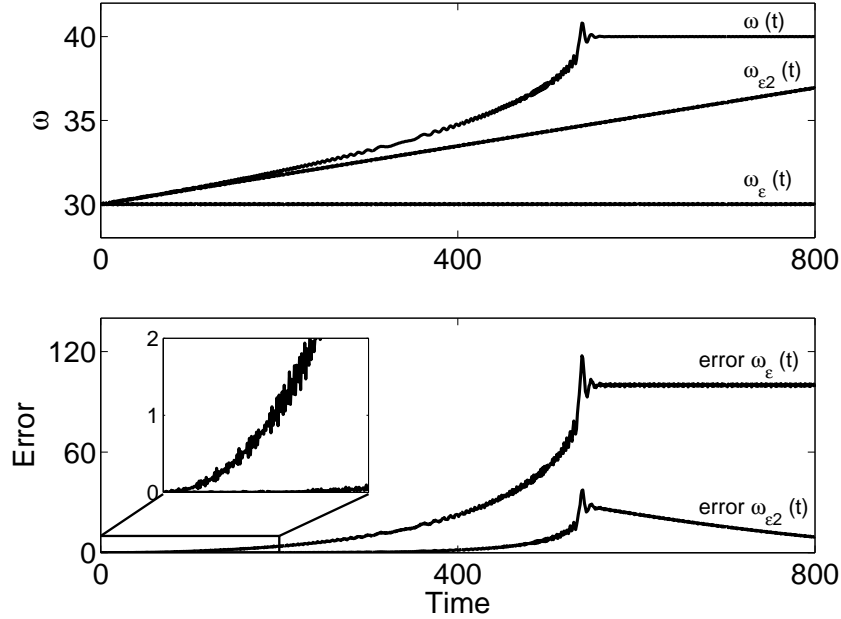


Figure 5.3: Results of the simulation of the first and second order approximations. For a simple input, here  $F = \sin(40t)$ ,  $\epsilon = 0.9$ , initial conditions are  $t_0 = 0$ ,  $w_0 = 30$ . The upper figure shows the evolution of the  $\omega$  variable for the initial dynamical system (Eq. (5.38)), the first order approximation  $\omega_\epsilon(t)$  and the 2nd order approximation  $\omega_{\epsilon^2}(t)$ . The lower figure shows quadratic errors between the initial system and the 2 approximations, for the evolution of  $\omega$ .

we can see that the oscillator adapts its intrinsic frequency to the frequency of the input signal. We also see that  $\epsilon$  controls the adaptation rate of the system, the higher  $\epsilon$  is, the faster the learning.

### Error evaluation of the analytic approximation for a simple perturbing force

In Section 5.1.2, we derived an approximate solution of the learning dynamical system, in order to show its convergence. The error of this approximation is bounded by some constant  $k$ . We now evaluate numerically the error of the approximation, for a simple sinusoidal input, in order to show that this constant is really small and that the hypothesis made for proving convergence holds. We set  $F = \sin(\omega_F t)$ ,  $t_0 = 0$ ,  $\mu = 1$ . Then we can

Time	Maximum Error $\omega_\epsilon$	Maximum Error $\omega_{\epsilon^2}$
0s	0	0
0.001s	$5.18e^{-13}$	$1.70e^{-19}$
0.01s	$4.91e^{-7}$	$1.15e^{-12}$
0.1s	0.0053	$6.30e^{-11}$
1s	0.0114	$1.85e^{-7}$
10s	0.0340	$4.25e^{-4}$

Table 5.1: This table summarizes the maximum errors of the simulation for the first and second order approximations discussed from Figure 5.3

derive an approximate solution of  $\omega(t)$  using Equations (5.25),(5.28) and (5.32).

$$\omega_0(t) = \omega_0 \tag{5.39}$$

$$\omega_1(t) = -\frac{1}{2(\omega_F - \omega_0)} \sin((\omega_F - \omega_0)t) + \frac{1}{2(\omega_F + \omega_0)} \sin((\omega_F + \omega_0)t) \tag{5.40}$$

$$\begin{aligned} \omega_2(t) = & \frac{\sin(2\omega_0 t)}{16\omega_0(\omega_F - \omega_0)} - \frac{\sin(2\omega_F t)}{16\omega_F(\omega_F - \omega_0)} - \frac{\sin(2(\omega_F - \omega_0)t)}{16(\omega_F - \omega_0)^2} + \frac{t}{8(\omega_F - \omega_0)} \\ & - \frac{t}{8(\omega_F + \omega_0)} + \frac{\sin(2(\omega_F + \omega_0)t)}{16(\omega_F + \omega_0)^2} - \frac{\sin(2\omega_0 t)}{16\omega_0(\omega_F + \omega_0)} + \frac{\sin(2\omega_F t)}{16\omega_F(\omega_F + \omega_0)} \\ & + \frac{\cos(2\omega_F t) - 1}{16\omega_F(\omega_F - \omega_0)^2} + \frac{\cos(2\omega_0 t) - 1}{16\omega_0(\omega_F - \omega_0)^2} + \frac{\cos(2(\omega_F - \omega_0)t) - 1}{16(\omega_F - \omega_0)^3} \\ & - \frac{\cos((\omega_F + \omega_0)t) - 1}{4(\omega_F - \omega_0)^2(\omega_F + \omega_0)} - \frac{\cos((\omega_F - \omega_0)t) - 1}{4(\omega_F - \omega_0)^3} \\ & - \frac{\cos(2(\omega_F + \omega_0)t) - 1}{16(\omega_F + \omega_0)^3} - \frac{\cos(2\omega_F t) - 1}{16\omega_F(\omega_F + \omega_0)^2} + \frac{\cos(2\omega_0 t) - 1}{16\omega_0(\omega_F + \omega_0)^2} \\ & + \frac{\cos((\omega_F + \omega_0)t) - 1}{4(\omega_F + \omega_0)^3} - \frac{\cos((\omega_F - \omega_0)t) - 1}{4(\omega_F + \omega_0)^2(\omega_F - \omega_0)} \end{aligned} \tag{5.41}$$

We can now numerically evaluate the errors of the approximations of order 1,  $\omega_\epsilon(t) = \omega_0 + \epsilon\omega_1(t)$ , and of order 2,  $\omega_{\epsilon^2}(t) = \omega_0 + \epsilon\omega_1(t) + \epsilon^2\omega_2(t)$ . The upper plot of Figure 5.3 shows the result of this simulation. First of all, we clearly see that the dynamical system correctly learns the frequency of the input signal. In this figure we also plotted the function  $\omega_\epsilon(t)$  and  $\omega_{\epsilon^2}(t)$ , we clearly see that the second order approximation is really better than the first and explains the behavior of the system on a larger time scale. Actually, it explains very well the convergence process of the learning dynamical system. We see that the learning appears on a coarser time scale than the oscillations of the system. In the lower plot, we see the square error between the original system and the approximations. We clearly see that the 2nd order approximation follows the

## 5.1. Frequency adaptation mechanism

---

real system for quite a long time. Table 5.1 summarizes the maximum square error of the approximations. It must be noted that numerical integration of the dynamical system is done with an embedded Runge-Kutta-Fehlberg(4,5) algorithm, with absolute and relative errors of  $10^{-6}$ . As a matter of fact, errors below this value cannot be taken as significant errors. Obviously, the first order approximation diverges rapidly, at 0.1s of simulation, the error is becoming really significant. On the other hand, the second order approximation is really good still after 10s. These results validate the hypothesis of the approximation methods and so, the analytic proof. It also emphasizes the fact that learning takes place on a larger time-scale than the perturbations on the oscillator and its oscillations. Consequently, the adaptive Hopf oscillator has two distinct time scales. The finer one describes the perturbation on the oscillator and its oscillations. Learning takes place on the coarser one.

### Predicting learning with multi-frequency inputs

When learning frequency of multi-frequency input signals, we might expect the system to converge to one of the frequency components of the input. But how can we calculate the range of initial frequencies for which the adaptive oscillator will converge to a specific frequency component of the input? While proving the convergence of the system, we derived a deviation equation, Equation (5.34), that describes the deviation from the initial intrinsic frequency,  $\omega_0$ , of the oscillator

$$D_\omega(t) = \left( \frac{-A_0}{2\sqrt{\mu}\omega_0} + \sum_{n \in \mathbb{N}} \frac{|A_n|^2 \omega_0}{\sqrt{\mu}((n\omega_F)^2 - \omega_0^2)} \right) (t - t_0) \quad (5.42)$$

We saw that this equation is depending on the initial frequency of the system  $\omega_0$ , the frequency components of the periodic input  $n\omega_F$  and their amplitude  $A_n$ . Thus, for a given input signal, we can calculate the values of  $\omega_0$  for which the function is equal to zero  $\forall t$ . These zeros give the intervals of convergence, the dynamical system converging towards the frequency components located in the same interval as  $\omega_0$ .

For example consider the following input

$$F = 0.2 \sin(20t) + 0.5 \sin(30t) + 0.3 \sin(40t) \quad (5.43)$$

The main frequency of this signal is  $\omega_F = 10$ . The amplitude of the frequency component are  $A_2 = \frac{0.2}{2i}$ ,  $A_3 = \frac{0.5}{2i}$ ,  $A_4 = \frac{0.3}{2i}$  and  $A_i = 0, \forall i \in \mathbb{N} \setminus \{2, 3, 4\}$ . Thus we only have to find the roots of the following equation

$$\frac{0.2^2 \omega_0}{4(20^2 - \omega_0^2)} + \frac{0.5^2 \omega_0}{4(30^2 - \omega_0^2)} + \frac{0.3^2 \omega_0}{4(40^2 - \omega_0^2)} = 0 \quad (5.44)$$

The solutions of this equation are 0 and  $\pm \sqrt{\frac{717 \pm \sqrt{134089}}{0.76}}$ . As we are working with frequencies  $> 0$  we have the following bounds  $\omega_{down} \simeq 21.3598$  and  $\omega_{up} \simeq 37.8233$ . Thus

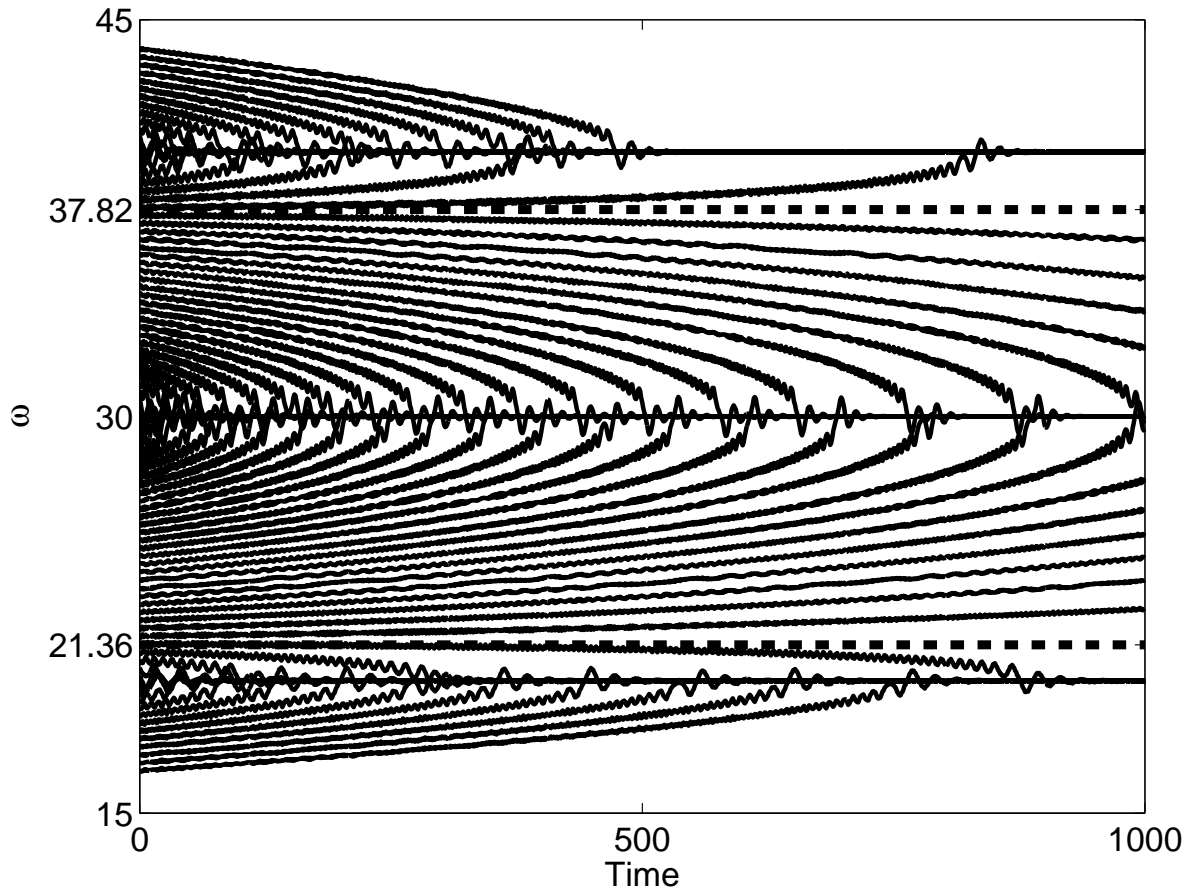


Figure 5.4: In this figure, we plotted  $\omega(t)$  for several initial conditions,  $\omega_0$ . The periodic input is Equation (5.43),  $\epsilon = 0.9$ . The dotted lines indicate the boundary between the different basins of attraction, corresponding to the different frequency components of the input, that were predicted analytically.

## 5.1. Frequency adaptation mechanism

---

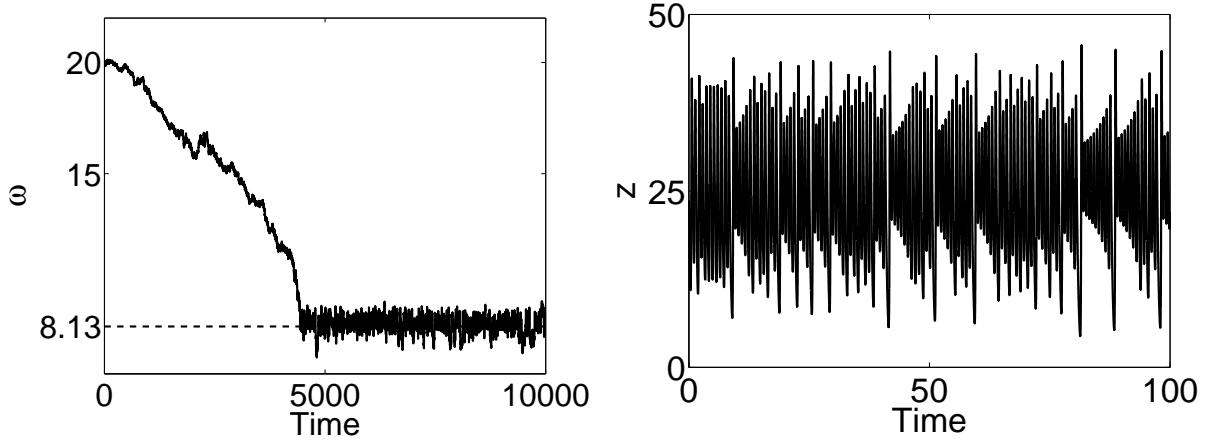


Figure 5.5: The left plot of this figure represents the evolution of  $\omega(t)$  when the adaptive Hopf oscillator is coupled to the  $z$  variable of the Lorenz attractor. The right plot represents the  $z$  variable of the Lorenz attractor. We clearly see that the adaptive Hopf oscillators can correctly learn the pseudo-frequency of the Lorenz attractor. See the text for more details.

we must expect to have convergence to 20, 30 or 40 when  $\omega_0 \in [0, \omega_{\text{down}}]$ ,  $[\omega_{\text{down}}, \omega_{\text{up}}]$ ,  $[\omega_{\text{up}}, \infty]$  respectively. With some uncertainty at the limit of the intervals, because of the oscillations of order  $\epsilon$  that can make the system switch from one interval to the other. Figure 5.4 shows this behavior, the horizontal dotted lines mark the bounds. Convergence corresponds to what we predicted.

### Learning pseudo-period of chaotic signals

We showed convergence for periodic signals, but we argue that even pseudo-periodic signals can be used as input for the learning dynamical system. In order to show this fact, we present the result of learning, when coupled to a chaotic pseudo-periodic signal. We couple the oscillator with the  $z$ -variable of the Lorenz system [133], whose equation is

$$\dot{x} = -\sigma x + \sigma y \quad (5.45)$$

$$\dot{y} = -xz + rx - y \quad (5.46)$$

$$\dot{z} = xy - bz \quad (5.47)$$

Where  $\sigma = 10$ ,  $r = 28$  and  $b = \frac{8}{3}$  (parameters for which the system produces a strange attractor). The Fourier spectrum of the  $z$ -variable indicates two major frequency components (data not shown). The first one at frequency 0 ( $A_0$  in the Fourier series), because the average of  $z$ ,  $\langle z \rangle \neq 0$ , and the second one at  $\sim 1.3\text{Hz}$ . As the zero frequency component has a really strong amplitude compared to the other and we do not want adaptation

to this frequency, we center the  $z$ -variable before coupling to the oscillator. Otherwise,  $\omega$  converges to 0 and the oscillations disappear. Indeed the basin of attraction corresponding to frequency  $\sim 1.3\text{Hz}$  is not very wide and  $\omega$  gets kicked out of it because of the chaotic nature of the input. Thus the input for coupling we use is  $F = z - \langle z \rangle$ .

Figure 5.5 shows the result of the learning process. After convergence,  $\langle \omega \rangle \simeq 8.13 \text{ rad} \cdot \text{s}^{-1}$  which corresponds to an intrinsic frequency of the oscillator of  $\sim 1.29 \text{ Hz}$ . Thus our adaptive dynamical system has learned the pseudo-frequency of the strange attractor. As this is not a strictly periodic signal,  $\omega(t)$  oscillates, following the constantly changing pseudo-frequency of the attractor.

This experiment enforces the idea that our adaptive dynamical system is able to learn the frequency of any periodic, or pseudo-periodic signal. It learns a frequency component of the input, even if the signal is really noisy or if the frequency is not strictly defined.

### 5.1.4 Generalization to non-harmonic oscillators

In previous sections, we presented an adaptive Hopf oscillator able to learn the frequency component of a periodic signal. The goal of this section is to show how we can easily apply our adaptive rule to non-harmonic oscillators like relaxation oscillators. The problem with such oscillators is that they have two time scales (slow buildup and fast relaxation) so it is difficult to treat them analytically to show convergence of the adaptive rule. In this section, we discuss in details the case of the Van der Pol oscillator, then we show results of the adaptive rule with the Rayleigh oscillator, the Fitzhugh-Nagumo oscillator and the Rössler system.

#### An adaptive Van der Pol oscillator

**The Van der Pol oscillator** The Van der Pol is a classical example of relaxation oscillator and is often used in biological modeling, for example to model CPGs for quadrupedal locomotion [26]. Its equation is

$$\ddot{x} + \alpha(x^2 - p^2)\dot{x} + \omega^2 x = 0 \tag{5.48}$$

Here  $\alpha$  controls the degree of nonlinearity of the system (the relaxation part),  $p$  the amplitude of the oscillations and  $\omega$  mainly influences the frequency of the oscillations. In this study we set the amplitude of oscillations to  $p = 1$ . We rewrite the system in a 2-dimensional form and perturb it in the direction of  $x$  as we did in Section 5.1.2

$$\dot{x} = y + \epsilon F \tag{5.49}$$

$$\dot{y} = -\alpha(x^2 - 1)y - \omega^2 x \tag{5.50}$$

Because of the relaxation property of the oscillator, the frequency spectrum contains, in addition to the frequency of the oscillations, an infinite number of frequency components.

## 5.1. Frequency adaptation mechanism

---

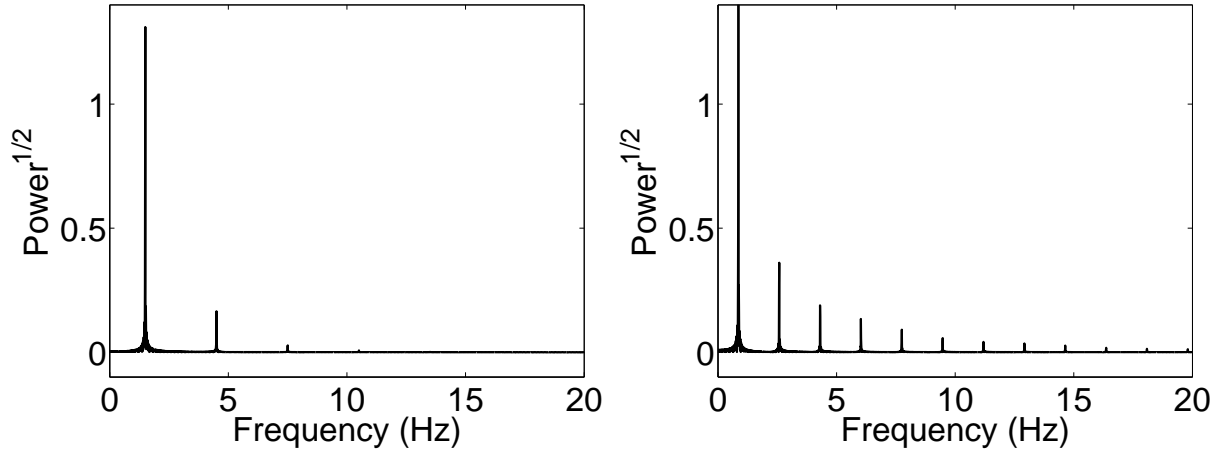


Figure 5.6: Frequency spectrum of the Van der Pol oscillator, both plot with  $\omega = 10$ . The left figure is an oscillator with  $\alpha = 10$  and on the right the nonlinearity is higher  $\alpha = 50$ . On the y-axis we plotted the square root of the power intensity, in order to be able to see smaller frequency components.

They are all multiples of the frequency of the oscillations and have smaller intensities. The nonlinear part of the system, whose importance is driven by the  $\alpha$  variable, influences the intensity of these components. It means the higher  $\alpha$  is, the more intensity high frequency components have. The frequency of the oscillations are mainly defined by  $\omega$ , but  $\alpha$  also influences this frequency. In fact an increase of the nonlinear term  $\alpha$  tends to slow the oscillator down.

Figure 5.6 shows the frequency spectrum of the  $x$  variable for two different values of  $\alpha$ . We clearly see that the intensities of the fast frequency components increase as  $\alpha$  increases. We also observe that the oscillator gets slower when  $\alpha$  increases (the peaks shift to the left). But still  $\omega$  is a good control parameter of the frequency of the system.

The complexity of the frequency spectrum of such oscillators complicates learning. Indeed, according to the initial conditions (i.e. according to the distance between the frequency of the periodic force and the main frequency of the oscillator), the oscillator may learn different frequencies and synchronize one of its higher frequency components to the input, instead of adapting its main frequency.

**The adaptive dynamical system** The adaptive rule we introduced in this section dynamically changes the parameter that mainly controls the frequency of the oscillations. Thus, in this case we will make the  $\omega$  parameter a dynamical system. Before discussing adaptation, we want to discuss the locations of the entrainment basins in function of  $\omega$ , in order to understand how the adaptive rule will work. The entrainment basins are the regions of frequencies where the oscillator phase-lock with an input signal [103].

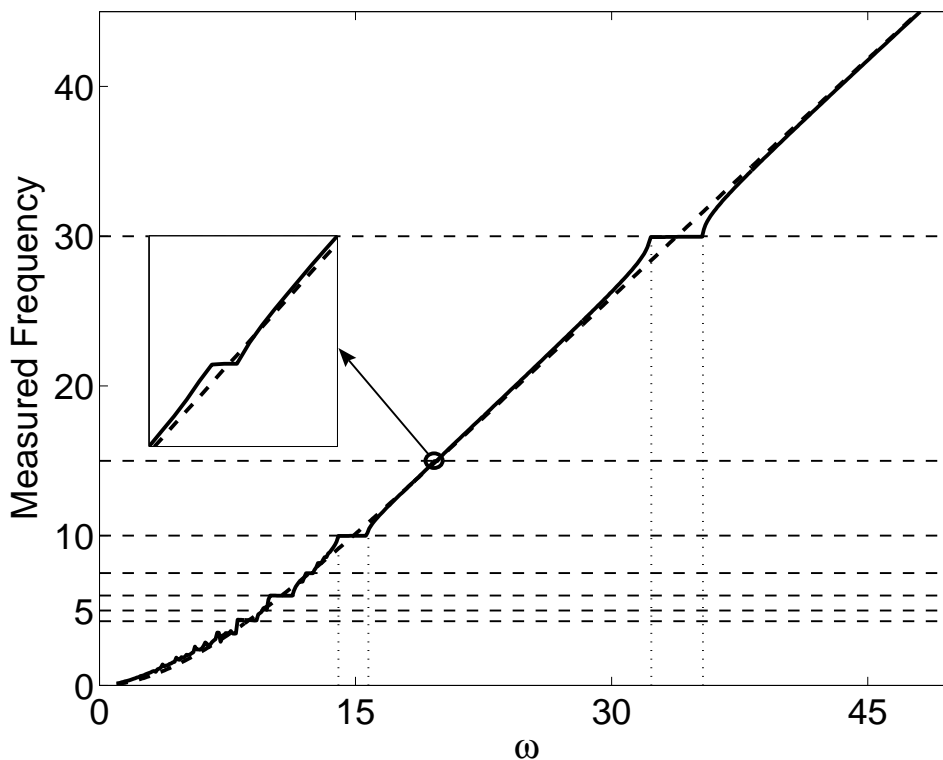


Figure 5.7: Plot of the frequency of the oscillations of the Van der Pol oscillator according to  $\omega$ . Here  $\alpha = 50$ . There are 2 plots, in dotted line the oscillator is not coupled and in plain line the oscillator is coupled to  $F = \sin 30t$ . The strength of coupling is  $\epsilon = 2$ . We clearly see basins of phase-locking, the main one for frequency of oscillations 30. The other major basins appears each  $\frac{30}{n}$  (dotted horizontal lines). We also notice small entrainment basins for some frequencies of the form  $\frac{30p}{q}$ . For a more detailed discussion of these results refer to the text.



## 5.1. Frequency adaptation mechanism

---

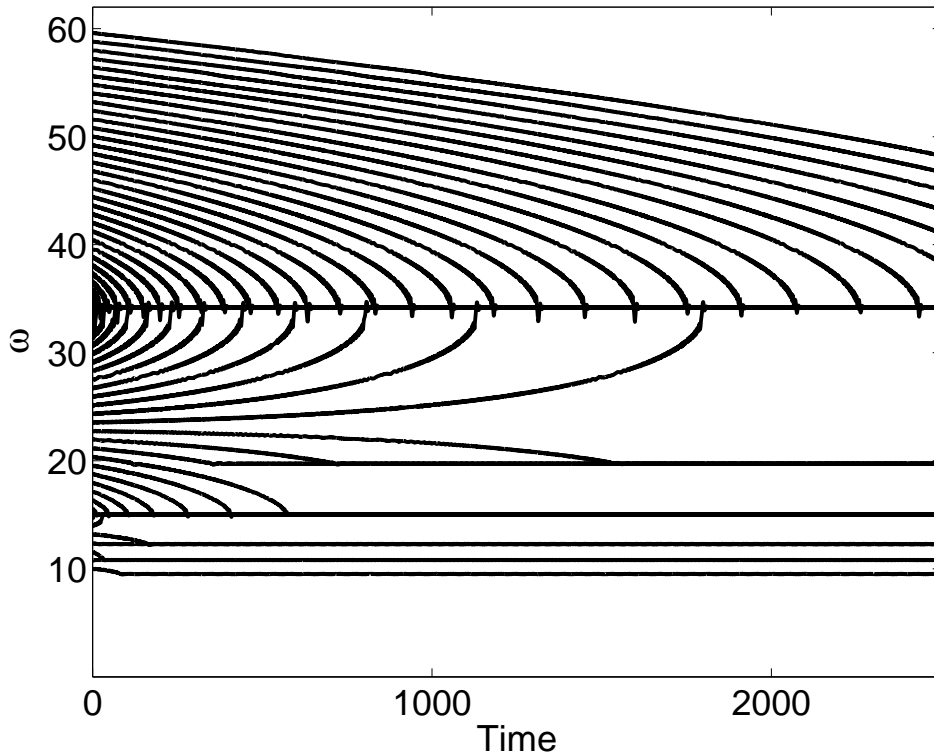


Figure 5.8: This figure shows the convergence of  $\omega$  for several initial frequencies. The Van der Pol oscillator is perturbed by  $F = \sin(30t)$ , with coupling  $\epsilon = 0.7$ ,  $\alpha = 50$ . We clearly see that the convergence directly depends on the initial conditions and as expected the different kinds of convergence correspond to the several entrainment basins of Figure 5.7.

Figure 5.7 shows the entrainment basins of a Van der Pol oscillator with high nonlinear component  $\alpha = 50$ , which is forced by a periodic signal  $\sin(30t)$ . As expected, we see phase locking at frequency of oscillations 30, with an entrainment basin of  $\omega \in [32, 35]$ . We also explained that the oscillator may phase lock its higher frequency components, as these frequency components are equally spaced, one should expect phase lock for fractions of the frequency of the perturbing force. In this case, for example, we see phase locking at frequencies of oscillations  $\frac{30}{2}$ ,  $\frac{30}{3}$  and  $\frac{30}{4}$ .

This figure may become even more complex if the input signal has several frequency components. We would see entrainment basins every time a frequency component of the oscillator is close enough to any frequency component of the external signal. Then, when using our adaptive rule, one should expect convergence to any entrainment basins, depending on the initial conditions. Therefore, the oscillator might adapt its higher frequency components to the frequency of the input.

We now discuss the learning rule we introduced in Section 5.1.2, applied to the Van

der Pol oscillator. We just change the sign of Equation (5.7). This is justified because when looking to the limit cycle of the Van der Pol oscillator, we see that it is rotating in the opposite direction of the Hopf oscillator limit cycle. So the learning rule is

$$\dot{\omega} = \epsilon F \frac{y}{\sqrt{x^2 + y^2}} \quad (5.51)$$

We do not give an analytical proof of convergence for the Van der Pol oscillator because to use perturbation methods, as we did for the Hopf oscillator, we need to know the solution of the unperturbed Van der Pol oscillator, but to the best of our knowledge, only implicit solutions are known [36] and thus such a proof is beyond the scope of this section. But the general behavior of the system should be qualitatively the same, because of the linear coupling on the oscillator. Lets rewrite Equations (5.49) and (5.50) into polar coordinates

$$\dot{r} = \epsilon F \cos \phi + (1 - \omega^2)r \cos \phi \sin \phi + \alpha r^3 \sin^4 \phi \quad (5.52)$$

$$\dot{\phi} = -\omega^2 \cos^2 \phi - \sin^2 \phi + \alpha r^2 \sin^3 \phi \cos \phi - \frac{\epsilon F}{r} \sin \phi \quad (5.53)$$

Even if the phase evolution is more complex than for the Hopf oscillator, the interaction between the phase of the oscillator  $\phi$  and the perturbation  $F$  is of the same kind. Indeed, we clearly identify the same  $-\frac{\epsilon F}{r} \sin \phi$  terms for the phase of both oscillators (Eqs. (5.52), (5.53) and Eqs. (5.3), (5.4)). So we can expect the same deviation of  $\omega$  and therefore, the same convergence properties.

Now that we discussed the different expected behaviors, we present a series of experiments in order to confirm our predictions and the functionality of the adaptive dynamical system.

**Numerical confirmation** We predicted that the adaptive Van der Pol oscillator will either adapt its frequency of oscillations or one of its higher frequency component to the frequency of the input. In order to show this, we study convergence of  $\omega$  for different initial conditions, when the oscillator is coupled with a simple sinus input ( $F = \sin(30t)$ ). Figure 5.8 shows the result of the simulation.

When the initial condition  $\omega_0 > 23$ , we clearly see that  $\omega$  converges to 34 which corresponds to a frequency of oscillations of  $30 \text{ rad} \cdot \text{s}^{-1}$ . In this case the oscillator is correctly adapting its frequency to the frequency of the input. For lower values of  $\omega_0$ , we see convergence to other frequencies, corresponding to the entrainment basins of Figure 5.7. We can conclude that the adaptive rule is changing  $\omega$  in order to get one frequency component of the oscillator to the same frequency than the input signal. In fact,  $\omega$  is falling into the nearest entrainment basin. Therefore, we see how useful entrainment basins studies are to understand the dynamics of the adaptive oscillator.

Moreover, even if there is not a direct relation between  $\omega$  and the frequency of the oscillations, the adaptive learning rule can appropriately tune  $\omega$  so that the frequency

## 5.1. Frequency adaptation mechanism

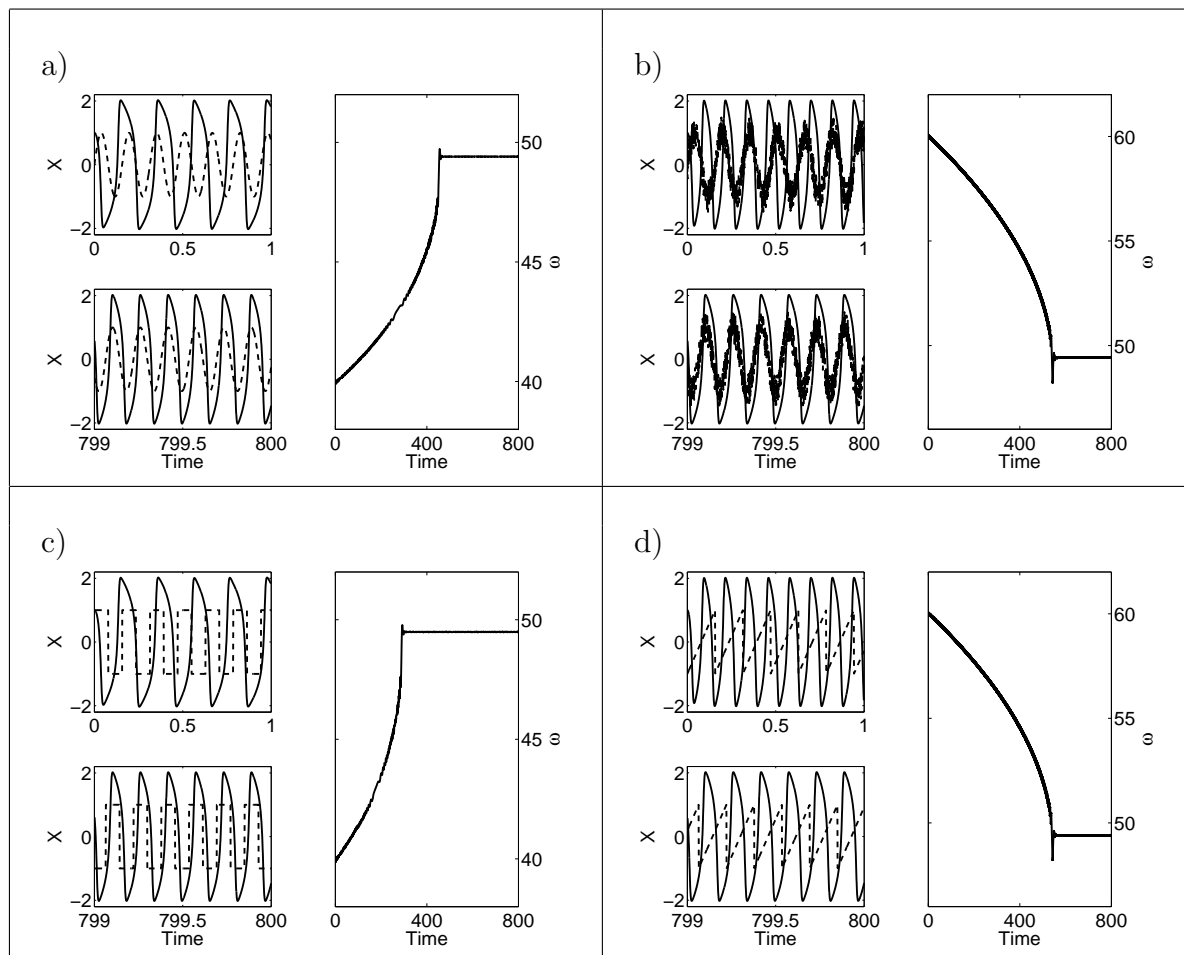


Figure 5.9: We show the adaptation of the Van der Pol oscillator to the frequency of various input signals: (a) a simple sinusoidal input ( $F = \sin(40t)$ ), (b) a sinusoidal input with uniformly distributed noise ( $F = \sin(40t) + \text{uniform noise in } [-0.5, 0.5]$ ), (c) a square input ( $F = \text{square}(40t)$ ) and (d) a sawtooth input ( $F = \text{sawtooth}(40t)$ ). For each experiment, we set  $\epsilon = 0.7$  and  $\alpha = 100$  and we show three plots. The right one shows the evolution of  $\omega(t)$ . The upper left graph is a plot of the oscillations,  $x$ , of the system, at the beginning of the learning. The lower graph shows the oscillations at the end of learning. In both graphs, we also plotted the input signal (dashed). In each experiment,  $\omega$  converges to  $\omega \simeq 49.4$ , which corresponds to oscillations with a frequency of  $40 \text{ rad} \cdot \text{s}^{-1}$  like the input and thus the oscillator correctly adapts its frequency to the frequency of the input.

of oscillations (or one of the other frequencies of the oscillator) are the same than the frequency of the input signal. Figure 5.9 shows the result of the adaptation of the oscillator for various input signals. From these experiments, we see that  $\omega$  converges to a value that corresponds to a correct adaptation of the frequency of the oscillations to the frequency of the input. In each experiment, we see that after learning, the Van der Pol oscillator and the input signal are oscillating at the same frequency.

The adaptive Van der Pol oscillator demonstrates how to generalize our adaptive rule to complex oscillators. But, an increase in the complexity of the frequency spectrum of an oscillator also generates side effects, like adaptation toward synchronization of the higher frequency components of the oscillator and the frequency of an input signal. Thus, when using highly nonlinear oscillators, one should always know the kind of frequency spectrum it has, in order to be able to predict the behavior of the oscillator. Even if we cannot analytically prove the convergence of our model, by numerically calculating the positions of the entrainment basins of the oscillator when perturbed, we are able to predict the behavior of the system in a quite powerful way.

In this section, we also discussed a very important property of the adaptive learning rule. Although, the parameter we tune has not a linear relation with the frequency of the oscillator, as it is often the case in highly nonlinear oscillators, the adaptive oscillator is able to correctly adapt this parameter and find the appropriate frequency of oscillations. It seems that a monotone relation between the frequency of the oscillations and the parameter we tune is sufficient for frequency adaptation.

### Other examples of adaptive oscillators

In this section, in order to show the generality of the adaptive rule, we present experimental results with three other oscillators. We build an adaptive Rayleigh oscillator, an adaptive Fitzhugh-Nagumo oscillator and an adaptive Rössler system.

The construction of the adaptive dynamical system is straightforward. The main task is to identify in each oscillator the parameter that mostly influences the frequency of the oscillations. Then, we only have to make this parameter a dynamical system in the same way we did for the Hopf or the Van der Pol oscillator. The right column of Figure 5.10 gives the resulting equations for each oscillator.

In order to demonstrate the frequency adaptivity of these modified oscillators, we made experiments for each oscillator. The results of the experiments are summarized in Figure 5.10. In these experiments, the oscillators were perturbed by a simple sinusoidal input and each oscillator was able to adapt its  $\omega$  parameter in order to learn the frequency of the input. Moreover, although the parameters controlling the frequency in each oscillator are not linearly related to the frequency of the oscillations, the adaptive rule is able to correctly find the correct value for the  $\omega$  parameter to learn the desired frequency.

## 5.1. Frequency adaptation mechanism

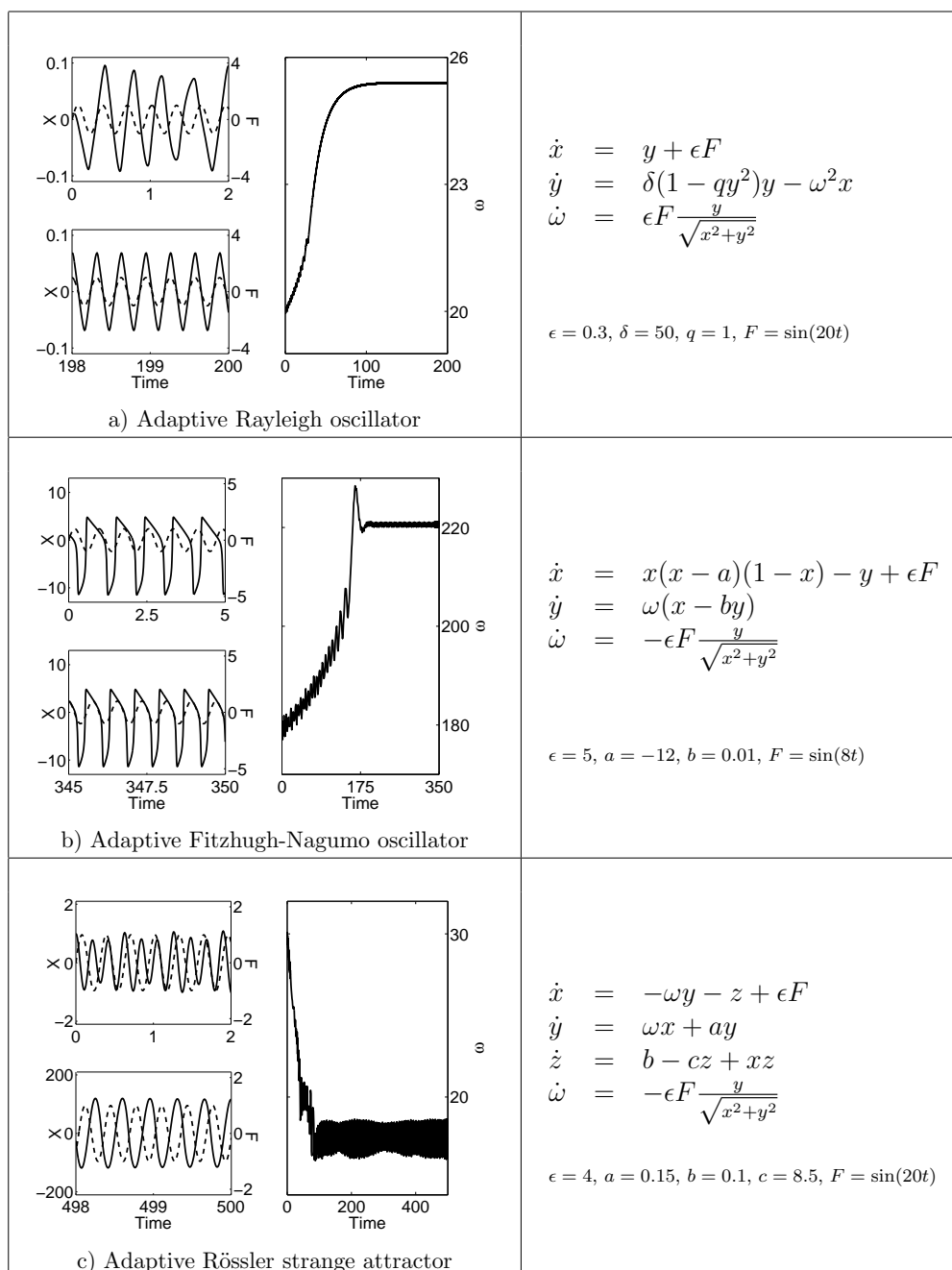


Figure 5.10: We show results for several adaptive oscillators. For each oscillator, we give its equation in the right column,  $\omega$  corresponding to the adaptive parameter. We also specify the values of the different parameters used in the experiments. In the left column we plotted results of the experiment. Each figure is composed of 3 plots. The right one is a plot of the evolution of  $\omega$ . The left ones are plots of the oscillations (the  $x$  variable) and of the input signal  $F$  (dashed line), before (upper figure) and after (lower figure) adaptation.

### 5.1.5 Discussion

Fields such as control of autonomous robots or signal processing may need models of plastic dynamical systems to adapt to a constantly changing environment. Moreover, plasticity in nonlinear oscillators might become an important aspect in modeling adaptive processes, as for example in biology where adaptivity and memory are major properties of living systems. The learning rule presented in this section is a step towards a general framework of plastic dynamical systems, which are systems for which learning is embedded in their dynamics and not an offline optimization process.

The evolution of the parameter controlling the frequency of the adaptive oscillators that we discussed can be viewed as the correlation between the phase of the oscillator and the input signal. So we defined a type of correlation-based learning for periodic functions. In neurobiology, correlation based-learning rules are known as Hebbian learning [73], hence we call our rule *dynamic Hebbian learning* to highlight its correlation properties. Possible relevance to biology has to be investigated in further research.

The construction of adaptive oscillators that we presented is simple, and general enough to be applied to non-harmonic oscillators and not only to phase oscillators. The adaptive rule is general for an oscillator, perturbed by a signal  $F(t)$ , with general equation

$$\begin{aligned}\dot{x} &= f_x(x, y, \omega) + \epsilon F(t) \\ \dot{y} &= f_y(x, y, \omega)\end{aligned}\tag{5.54}$$

with  $\omega$  influencing the frequency of the oscillations. We have the general learning rule

$$\dot{\omega} = -\epsilon F \frac{y}{\sqrt{x^2 + y^2}}\tag{5.55}$$

Only the sign in front of  $F$  may change according to the orientation of the flow of the oscillator in the phase space. In this sense we generalize the concept of learning presented by Nishii in [97, 98], in which learning rules were only derived for phase oscillators. Nevertheless, in addition to frequency adaptation, Nishii also derived learning rules for coupling strength in populations of oscillators, which is an issue we do not address here.

The learning rule we presented is not rigid and can be modified. For instance, for the Hopf oscillator, a change in the learning rule in Eq. (5.7), from  $\sin\phi$  to  $\cos\phi$  or any combination of periodic functions will not change the convergence properties. This would only correlate the force to more complex periodic functions instead of  $\sin\phi$ . Intuitively, the proof of convergence should give the same results, since the learning part of the approximation (Equation (5.34)) depends on the conjugate symmetry of the complex Fourier series of the input signal, which is true for every real input signal.

The mathematical analysis given in this section leads to a better comprehension of the learning process, which takes place on a coarser time scale than the oscillations of the system. This proof also allows us to predict what the oscillator would learn in the case of multi-frequency inputs. Nevertheless, we only give a proof for the adaptive Hopf oscillator

## 5.2. Properties of adaptive phase oscillators

---

and even if we numerically show that more complex adaptive oscillators can be designed, a general rigorous proof for a larger class of oscillators is still missing. Constructing such a proof is a very difficult task.

A major feature of our learning rule is that the oscillator can extract the frequency of any input signal without any explicit signal processing (Fourier transform) or any explicit time window or similar parameters. All the processing is embedded in the dynamics of the oscillator. We also showed that the system can learn frequencies from really noisy signals or from pseudo-periodic signals, like a signal from the Lorenz strange attractor. The adaptive rule is also valid to tune parameters that do not control linearly the frequency of the oscillations. A monotonic, possibly nonlinear, relation between the frequency of oscillations and the adapted parameter is sufficient for correct adaptation of the parameter as we showed for the case of relaxation oscillators. In this case, the system is able to correctly find a value that produces oscillations at the same frequency as the input signal.

Dynamic Hebbian learning for adaptive oscillators has an important implication in the design of CPG models. Actually, coupled nonlinear oscillators are often used for modeling CPGs [27, 52, 80, 138], but the coupling has to be defined by hand and this is a non-trivial task. By using adaptive oscillators, one could build CPGs that can dynamically adapt their frequencies and consequently, create a desired pattern of oscillations. For instance, in the Section 5.3.2 we show how a population of adaptive oscillators can implement some kind of dynamic Fourier transform. Furthermore, one can imagine using this adaptation mechanism to model various processes where self-synchronization is observed.

## 5.2 Properties of adaptive phase oscillators

In this section we continue our analysis of adaptive frequency oscillators. In particular we show that for strong coupling convergence is exponential and we extend the mechanism to include an explicit relaxation time.

### 5.2.1 Motivation

In the previous section we proposed a general mechanism to transform an oscillator into an adaptive frequency oscillator (i.e. an oscillator that can adapt its parameters to learn the frequency of an input signal) [16, 110]. This mechanism is generic enough to be applied to a large class of oscillators with a wide range of driving signals. This mechanism goes beyond mere synchronization since it works for any initial frequencies (infinite basin of attraction) and if the input signal disappears ( $F(t) = 0$ ), the new frequency stays encoded in the system. Moreover it automatically tracks changes in the frequency of the input (for non-stationary signals) and after synchronization the phase delay is 0. This mechanism was successfully applied in adaptive control, where it was able to automatically tune a controller to the resonant frequency of a legged robot with passive dynamics and to track

changes in this frequency (for example when the gait changed) [15, 16, 17].

Numerical simulations show that this mechanism is also working for strong coupling  $K \gg 1^2$  and that the higher the coupling, the faster the convergence to the frequency. After convergence, the frequency parameter still oscillates around the correct frequency value and its amplitude increases with coupling. But it seems that this amplitude is bounded when  $K \rightarrow \infty$  as well as its speed of convergence. However, an analytic characterization of these properties is still missing and this is one of the goals of this section.

The goal of this section is to present a detailed description of the properties of adaptive phase frequency oscillators, in order to understand the fundamental behavior of the adaptation mechanism. These results could then serve as a basis to analyze more complex adaptive frequency oscillators. We show that for strong coupling ( $K \rightarrow \infty$ ) the convergence of  $\omega$  is in average exponential. We then introduce a parameter that allows one to control explicitly the relaxation time associated to this exponential convergence. As one can expect from the Fourier uncertainty relationship, this relaxation time is intrinsically related to the final resolution in frequency. We also extend our results to any  $K$ .

## 5.2.2 Strongly coupled adaptive frequency phase oscillator

In the following we derive our results using a phase oscillator, even though we used Hopf oscillators in our previous section. We justify this because for strong coupling it turns out that the frequency to which the oscillator converges is slightly different than the expected frequency. This behavior is mainly due to the interaction of the adaptation mechanism with the radius of the oscillator (in phase space) and is not related to the fundamental frequency adaptation process (see Section 5.2.3 for a more detailed explanation). Thus we use a phase oscillator to exhibit the fundamental frequency adaptation mechanism.

### Harmonic perturbation

We first analyze the system when perturbed by a simple harmonic signal, in further sections we extend this to more complex signals. Let an adaptive phase oscillator, i.e. a phase oscillator strongly coupled to a periodic input with the adaptation rule for its frequency

$$\dot{\phi} = \omega - KF \sin \phi \tag{5.56}$$

$$\dot{\omega} = -KF \sin \phi \tag{5.57}$$

where  $F = \cos(\omega_F t)$  and  $K$  high enough. We look at the differences  $\omega_d = \omega - \omega_F$  and  $\phi_d = \phi - \omega_F t$  in order to be able to do a fixed point analysis in the following. We then

---

<sup>2</sup>Starting from here we replace  $\epsilon$  with  $K$  because we will discuss the strong coupling case.



## 5.2. Properties of adaptive phase oscillators

---

get

$$\dot{\phi}_d = \omega_d - \frac{K}{2}(\sin \phi_d + \sin(2\omega_F t + \phi_d)) \quad (5.58)$$

$$\dot{\omega}_d = -\frac{K}{2}(\sin \phi_d + \sin(2\omega_F t + \phi_d)) \quad (5.59)$$

We rewrite the system into time-invariant and time dependent components and look at the time-dependant components as a perturbation, adding a factor  $\lambda$

$$\dot{x} = f(x) + \lambda g(x, t) \quad (5.60)$$

where  $x = \begin{pmatrix} \phi_d \\ \omega_d \end{pmatrix}$ ,  $f(x) = \begin{pmatrix} \omega_d - \frac{K}{2} \sin \phi_d \\ -\frac{K}{2} \sin \phi_d \end{pmatrix}$  and  $g(x, t) = -\frac{K}{2} \sin(2\omega_F t + \phi_d) \begin{pmatrix} 1 \\ 1 \end{pmatrix}$  and where  $\lambda$  is the strength of the perturbation ( $\lambda = 1$  in the original system). We write the Taylor series expansion of  $x(t, \lambda)$  around  $\lambda = 0$  to isolate the time-invariant terms

$$x(t, \lambda) = x(t, 0) + \sum \frac{\partial^n x(t, \lambda)}{\partial \lambda^n} \Big|_{\lambda=0} \frac{\lambda^n}{n!} \quad (5.61)$$

In the following, we first study the unperturbed system,  $x(t, 0)$  using singular perturbation theory and then we investigate the effect of the higher order terms in the Taylor series.

**Solving  $x(t, 0)$**  The time invariant system that we first need to solve is

$$\dot{\phi}_d = \omega_d - \frac{K}{2} \sin \phi_d \quad (5.62)$$

$$\dot{\omega}_d = -\frac{K}{2} \sin \phi_d \quad (5.63)$$

rescaling  $\Omega = \omega_d \frac{2}{K}$  we get the singular perturbation problem

$$\epsilon \dot{\phi}_d = \Omega - \sin \phi_d \quad (5.64)$$

$$\dot{\Omega} = \sin \phi_d \quad (5.65)$$

where  $\epsilon = \frac{2}{K} \ll 1$ . To solve this system, we use the singular perturbation theory presented in [74]. The system has two distinct time scales, Equation (5.64) is varying rapidly while Equation (5.65) varies on a slower time scale. In order to solve this system of equations, we first solve an auxiliary system, taking for Equation (5.64)  $\epsilon = 0$  and  $\Omega$  constant. We get

$$\phi_d = \sin^{-1} \Omega \quad (5.66)$$

This solution is valid only for  $\Omega < 1$  which corresponds to the case where the frequency of the phase oscillator of Equation (5.62) (without any frequency adaptation) has entered in its entrainment basin. Injecting this solution into equation (5.65) we get

$$\Omega(t) = \Omega_0 e^{-t} \quad (5.67)$$

From this we solve, what is called in [74], the boundary layer equation defined by

$$\frac{d\tilde{\phi}_d}{d\tau} = \Omega - \sin(\sin^{-1} \Omega + \tilde{\phi}_d) \quad (5.68)$$

where  $\Omega$  is kept fixed.  $\tilde{\phi}_d = 0$  is an exponentially stable fixed point of the system and the solution is

$$\tilde{\phi}_d(\tau) = 2 \tan^{-1} \left( \frac{1}{\Omega} (1 - \sqrt{\Omega^2 - 1} \tan(\frac{-\tau \sqrt{\Omega^2 - 1}}{2})) \right) \quad (5.69)$$

We can now use Theorem 11.2 of [74] that tells us that there exists a positive  $\epsilon^*$  such that  $\forall t_0 \geq 0$  and  $0 < \epsilon < \epsilon^*$  the previous singular perturbation problem (Eqs. (5.64)-(5.65)) has a unique solution on  $[0, \infty]$  and that  $\Omega(t, \epsilon) - \Omega_0 e^{-t} = O(\epsilon)$  and  $\phi_d(t, \epsilon) - \sin^{-1}(\Omega_0 e^{-t}) - \tilde{\phi}_d(\frac{t}{\epsilon}) = O(\epsilon)$ . Moreover  $\forall t_1 > t_0$ , there is a  $\epsilon^{**}$  such that  $\phi_d(t, \epsilon) - \sin^{-1}(\Omega_0 e^{-t}) = O(\epsilon)$  holds uniformly for  $t \in [t_1, \infty]$  whenever  $\epsilon < \epsilon^{**}$ . Relating these results to the original system of Equations (5.62)-(5.63), it means that we can find sufficiently high coupling  $K$  such that in the region where  $\omega_d \frac{2}{K} < 1$  (when the oscillator enters its entrainment basin) we have for any  $t > 0$

$$\phi_d(t) = \sin^{-1}(\omega_d(t)) + O\left(\frac{2}{K}\right) \quad (5.70)$$

$$\omega_d(t) = \omega_d(0)e^{-t} + O\left(\frac{2}{K}\right) \quad (5.71)$$

It is an interesting result since it shows that when  $t \rightarrow \infty$  we eventually get  $\phi_d = 0$  which means that we have synchronization of the phases and  $\omega_d = 0$  which means that the correct frequency is learned. Moreover it shows that for the frequency adaptation, convergence is exponential with relaxation time 1.

**Higher order terms of the Taylor series** We now calculate the  $\frac{\partial^n x(t, \lambda)}{\partial \lambda^n}|_{\lambda=0}$  terms in the Taylor series expansion (Eq. (5.61)). For  $n = 1$  we have

$$x_\lambda(t, \lambda) = \int \left( \frac{\partial f}{\partial x} + \lambda \frac{\partial g}{\partial x} \right) \frac{\partial x}{\partial \lambda} + g(x, s) ds \quad (5.72)$$

where  $x_\lambda = \frac{\partial x}{\partial \lambda}$ . Its time derivative at  $\lambda = 0$  is

$$\dot{x}_\lambda = A(t)x_\lambda + \frac{K}{2}b_1(t) \quad (5.73)$$

where we have

$$A(t) = \frac{\partial f}{\partial x}|_{x(t,0)} = \begin{bmatrix} -\frac{K}{2} \cos(\phi_d(t, 0)) & 1 \\ -\frac{K}{2} \cos(\phi_d(t, 0)) & 0 \end{bmatrix} \quad (5.74)$$

## 5.2. Properties of adaptive phase oscillators

---

and

$$b_1(t) = g(x, t)|_{x(t,0)} = -\sin(\phi_d(t, 0) + 2\omega_F t) \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad (5.75)$$

which can be approximated (at any precision by increasing  $K$ ) using Equation (5.70)

$$A(t) \simeq \begin{bmatrix} -\frac{K}{2} \sqrt{1 - \frac{2\omega_d(0)}{K}} e^{-t} & 1 \\ -\frac{K}{2} \sqrt{1 - \frac{2\omega_d(0)}{K}} e^{-t} & 0 \end{bmatrix} \quad (5.76)$$

and

$$b_1(t) \simeq -\sin(\sin^{-1}(\omega_d(0)e^{-t}) + 2\omega_F t) \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad (5.77)$$

Generally we see that higher order partial derivatives of  $x$  by  $\lambda$  have a time derivative that has the form of

$$\dot{x}_{\lambda^n} = A(t)x_{\lambda^n} + \frac{K}{2}b_n(t, x_{\lambda}, \dots, x_{\lambda^{n-1}}) \quad (5.78)$$

where  $b_n$  is  $2\omega_F$  periodic in  $t$ . We also notice that  $b_n$  is made of polynomial combinations of  $x_{\lambda^k}$ ,  $k < n$ , such that for each monomial, the sum of the degrees of each  $x_{\lambda^k}$  times  $k$  is lower or equal to  $n - 1$ . The convergence and boundedness of the Taylor series depends on the behavior of the matrix  $A(t)$  when forced by the periodic functions  $b_n$ . We study the properties of this matrix next.

**Properties of  $A(t)$**  First we can notice that the eigenvalues of  $A(t)$  are always negative for  $t > 0$ , thus the linear systems defined by (5.78) are BIBO-stable and inputs  $b_n$  make the  $x_{\lambda^n}$  converge to some periodic function of frequency  $2\omega_F$  after some transient. Moreover, this matrix converges exponentially fast to

$$\lim_{t \rightarrow \infty} A(t) = A_{\infty} = \begin{bmatrix} -\frac{K}{2} & 1 \\ -\frac{K}{2} & 0 \end{bmatrix} \quad (5.79)$$

Approximating  $A(t)$  by  $A_{\infty}$  we find the Laplace transform of  $x_{\lambda^n}$  valid for some  $t > 0$  (after the transient) as

$$X_{\lambda^n}(s) = (Is - A_{\infty})^{-1}CB_n(s) \quad (5.80)$$

where  $C = \begin{bmatrix} \frac{K}{2} & 0 \\ \frac{K}{2} & 0 \end{bmatrix}$ . Thus the corresponding transfer function  $H(s) = \frac{X_{\lambda^n}(s)}{B_n(s)}$  is

$$H(s) = \begin{bmatrix} \frac{K}{2} \frac{s+1}{s^2 + \frac{K}{2}(s+1)} & 0 \\ \frac{K}{2} \frac{s}{s^2 + \frac{K}{2}(s+1)} & 0 \end{bmatrix} \quad (5.81)$$

We see that the gain when  $K \rightarrow \infty$  is independent of  $K$  and equal to  $\left(\frac{1}{\sqrt{1+\omega^2}}\right) < 1$ . Taking the previous observation on the structure of the  $b_n$  we can write that for  $\lambda = 1$

$$\|x_{\lambda^n}\|_{\lambda=0} \leq \|b_n\| \leq d_n \|b_1\| \quad (5.82)$$

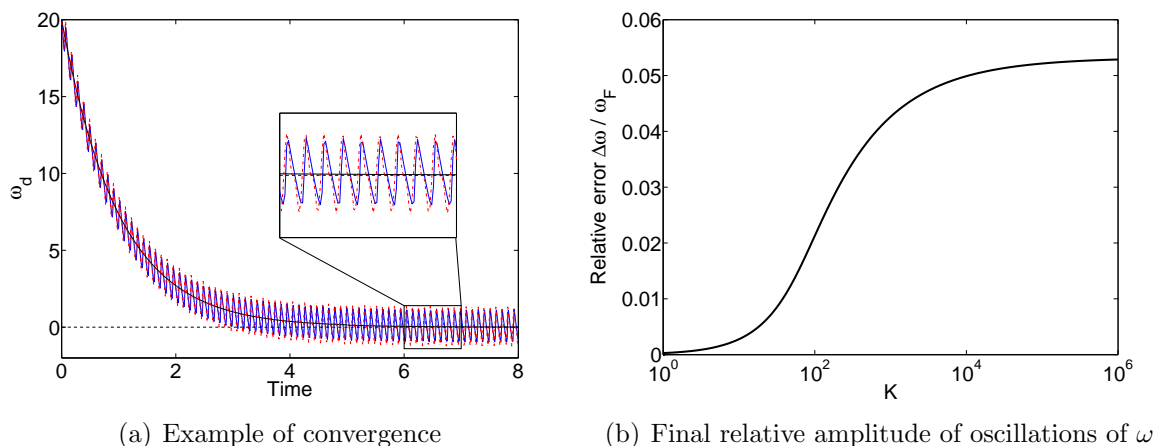


Figure 5.11: (a) We plot  $\omega_d$  for the adaptive frequency phase oscillator (in blue), the approximated system up to the second order term of the Taylor series (in dashed red) and the exponential convergence (in black). We used  $\omega_F = 30$ ,  $\omega(0) = 50$  and  $K = 1000$ . We see the good match between the approximations and the original system. (b) This figure shows the final relative amplitude of oscillations of  $\omega$  after convergence of the original system as a function of  $K$ . In this experiment we used  $\omega_F = 30$ .

where  $d_n$  is a number that cannot grow faster than  $n!$ . Thus we can now bound the Taylor series around  $\lambda = 1$  and get

$$\left\| \sum x_{\lambda^n} \Big|_{\lambda=0} \frac{1}{n!} \right\| \leq \sum \frac{d_n \|b_1\|}{n!} \quad (5.83)$$

We then see that the series is bounded and converges absolutely. The solution of Equations (5.56)-(5.57) for strong coupling  $K$  can then be written as

$$\dot{\phi}(t) = \sin^{-1}(\omega_d(t)) + P_\phi(t) + O\left(\frac{2}{K}\right) \quad (5.84)$$

$$\dot{\omega}(t) = \omega_F + (\omega(0) - \omega_F)e^{-t} + P_\omega(t) + O\left(\frac{2}{K}\right) \quad (5.85)$$

where  $P_\phi(t)$  and  $P_\omega(t)$  are  $2\omega_F$  periodic perturbations that have a maximum bounded amplitude independent of the coupling strength  $K$  when  $K \rightarrow \infty$ . In Figure 5.11(a) we show the behavior of the frequency variable ( $\omega_d$  of the original system (Eqs. (5.62)-(5.63)) together with the approximation from the singular perturbation problem (Eq. (5.71)) and the approximation from the Taylor series of order 2. We see very well the exponential convergence of the system in average and also that the first two terms of the Taylor series explains most of the oscillating behavior. Figure 5.11(b) shows how the final amplitude of oscillations ( $\Delta\omega$ ) around the frequency  $\omega_F$  changes as a function of the coupling strength. We clearly see that this amplitude is bounded when  $K \rightarrow \infty$ .

## 5.2. Properties of adaptive phase oscillators

---

### Control of the relaxation time

The convergence of frequency is exponential with relaxation time 1. From the previous analysis, it is then easy to choose an arbitrary relaxation time  $\tau$  for the exponential convergence by transforming the system as

$$\dot{\phi} = \frac{\omega}{\tau} - KF \sin \phi \quad (5.86)$$

$$\dot{\omega} = -KF \sin \phi \quad (5.87)$$

It can be seen as a rescaling of frequency or equivalently as a change in the frequency resolution (the frequency of interest is now  $\frac{\omega}{\tau}$ ). Performing the same analysis as before we can see that the convergence will be of order  $e^{-\frac{t}{\tau}}$ .

Another way of considering this change in frequency resolution is to make the change of variable  $\Omega = \frac{\omega}{\tau}$

$$\dot{\phi} = \Omega - KF \sin \phi \quad (5.88)$$

$$\dot{\Omega} = -\frac{K}{\tau} \sin \phi \quad (5.89)$$

and we see that we can consider the frequency rescaling as a different coupling strength for the  $\dot{\phi}$  and  $\dot{\Omega}$  equations. Theoretically we could converge as fast as possible if we set  $\tau \rightarrow 0$ , however this control of relaxation time does not come for free, as we will see next.

### Tradeoff between fast convergence and precision

In addition to exponential convergence there is a periodic oscillation  $P_\omega(t)$  that is conserved after convergence. Since this function is the weighted sum of the  $x_\lambda^n$ , its amplitude  $\Delta_\omega$  will be related to the frequency response of the transfer function  $H(s)$  associated to  $x_\lambda^n$  given by Equation (5.81).

We are interested in the relative amplitude  $\frac{\Delta_\omega}{\omega_F}$  and for relaxation time  $\tau$ , we can rewrite the magnitude of the frequency response of  $\omega_\lambda^n$  relatively to the converged frequency  $\omega_F$  for  $K \rightarrow \infty$  as

$$\frac{\|H_\omega(2\omega_F, \tau)\|}{\omega_F} = \frac{2}{\sqrt{1 + 4(\omega_F\tau)^2}} \quad (5.90)$$

from this equation, we see that in terms of error of convergence, changing the relaxation time is the same as changing the frequency of the input, i.e. doubling  $\tau$  will yield the same relative error as doubling  $\omega_F$ . From this analysis, we can expect that  $\frac{\Delta_\omega}{\omega_F}$  will increase as  $\omega_F$  decreases and that it will also increase when decreasing the time constant  $\tau$  (i.e. increasing speed of convergence).

In order to evaluate the error of convergence, another measure of interest is the spread of  $\omega$  around the converged frequency  $\omega_F$ . We define this as the standard deviation of

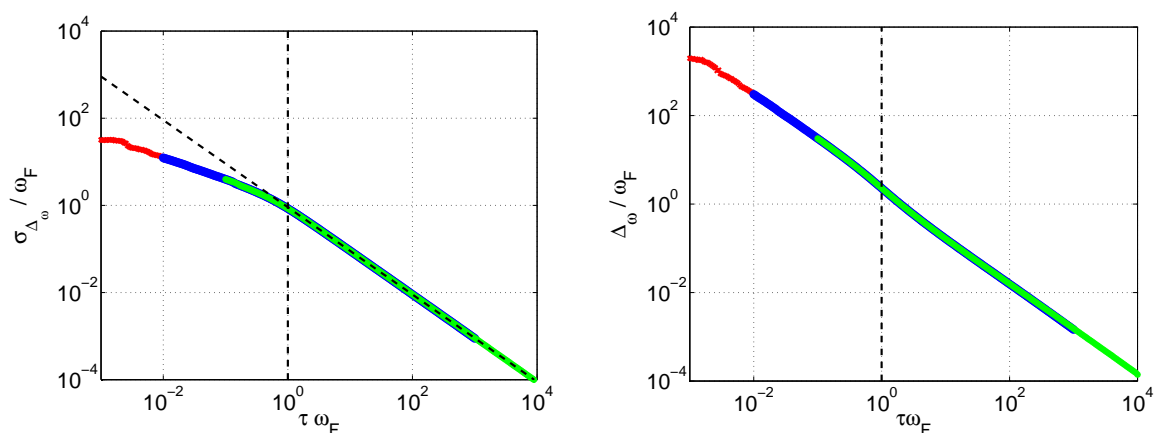


Figure 5.12: Left: relative standard deviation  $\sigma_{\Delta_\omega}$  of the  $\omega$  variable around the converged frequency as a function of  $\tau\omega_F$ . The diagonal dashed line shows the linear approximation for these values for  $\tau\omega_F > 1$ . Right: amplitude of the oscillations of  $\omega$  after convergence. Note the log scale on the two graphs. In this experiment, we used  $K = 10^7$  and  $\tau = 0.1, 1$  and  $10$  for the red, blue and green lines respectively.

$\omega$  after convergence  $\sigma_{\Delta_\omega}$ . This measure is the kind of measure that is used in signal processing to measure the relationship between time and frequency resolution.

We made experiments to measure both quantities  $\Delta_\omega$  and  $\sigma_{\Delta_\omega}$  relatively to  $\omega_F$  for different values of  $\tau$ , with  $K$  sufficiently high. Figure 5.12 shows the results of the experiments.

The first observation is that we get exactly the same results if we either change  $\omega_F$  or  $\tau$  in the same manner (i.e. the graphs for several  $\tau$  superpose perfectly if we use  $\tau\omega_F$  as the abscisse). This confirms what we predicted from Equation (5.90).

The second observation is that for  $\tau\omega_F < 1$ ,  $\sigma_{\Delta_\omega}$  becomes more than 100% of the converged frequency  $\omega_F$ , and the amplitude of oscillations are also much higher than 100% of  $\omega_F$ . These observations just show that it is not possible to have a good resolution on  $\omega_F$  if the time window defined by  $\tau$  is smaller than  $\omega_F^{-1}$  (i.e. we cannot converge with a small error faster than the input signal oscillates).

The third observation is the two linear relations between the scaled frequency  $\tau\omega_F$  and the relative amplitude and standard deviations. In the case of the standard error, the linear relation that interests us is the one for  $\tau\omega_F > 1$  (indeed we notice that there is an inflexion around that point). Linear regression on the data gives us the two relations

$$\ln\left(\frac{\Delta_\omega}{\omega_F}\right) = -1.07 \ln(\tau\omega_F) + 0.8036 \quad (5.91)$$

$$\ln\left(\frac{\sigma_{\Delta_\omega}}{\omega_F}\right) = -1.0006 \ln(\tau\omega_F) - 0.103 \quad (5.92)$$

## 5.2. Properties of adaptive phase oscillators

---

for the second relation (Eq. (5.92)) regression was done on the data set such that  $\tau\omega_F > 1$ . For this relation, if we approximate the slope with  $-1$  (since it is included in the confidence interval of the regression) and we then get

$$\sigma_{\Delta\omega}\tau \simeq 0.9021 \quad \text{for } \tau\omega_F > 1 \quad (5.93)$$

Note that it was not possible to assume the slope to be  $-1$  for the equation involving  $\Delta\omega$ , since it was not included in the confidence interval of the regression.

The result of Equation (5.93) is quite remarkable since it provides an equality relating the spread of  $\omega$  around  $\omega_F$  after convergence with the relaxation time (or time window) associated with the exponential convergence. It thus shows that these two quantities are closely related, even if at first sight they seem to measure two different processes.

If we relate this observation to what is known in signal processing, we can see  $\tau$  as an implicit time window for our system and  $\sigma_{\Delta\omega}$  as a frequency window and there is a relation between the two such that the area of the window in time-frequency space is constant (when  $K \rightarrow \infty$ ). We can then see  $\tau\sigma_{\Delta\omega}$  as an equivalent of an Heisenberg box for adaptive frequency oscillators.

### Generalization to finite coupling and more complex inputs

So far we have shown that the convergence of frequency was exponential, with relaxation time  $\tau$  only in cases where  $K$  is high enough and for sine waves as input signals. In this section we extend the results to any value  $K$  and to more complex input signals.

**Exponential convergence for finite  $K$**  In Section 5.2.2, we showed that frequency adaptation was exponentially fast. The singular perturbation problem that led to this conclusion has a solution only when the initial frequency of the oscillator is located in its entrainment basin (Eq. (5.62) without frequency adaptation) and then all our results were derived supposing that  $K$  was high enough. Now we conjecture that the important hypothesis for exponential convergence is the presence of the oscillator's frequency in its entrainment basin and that it is independent of the value of  $K$ . In the case of the original system (Eqs. (5.56)-(5.57)), the entrainment basin cannot be calculated explicitly but we can evaluate it numerically.

We performed simulations of an adaptive phase oscillator perturbed by a cosine input and evaluated its entrainment basin (without frequency adaptation) for a large range of  $K$  (between 1 to  $10^3$ ) and then the region of exponential convergence (with frequency adaptation). It turned out that these two regions match very well (data not shown), even for small coupling ( $K < 10$ ). In Figure 5.13 we show an example of such convergence, it is obvious that exponential convergence starts when the oscillator enters its entrainment basin, before convergence is slower.

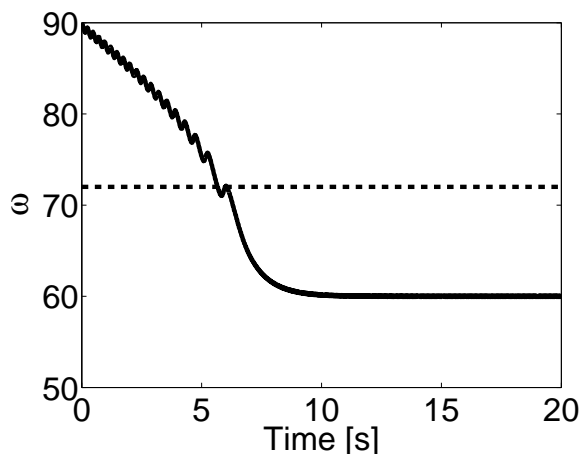


Figure 5.13: Example of convergence of  $\omega$  for small coupling ( $K = 20$ ). The input signal is  $F(t) = \cos(60t)$ ,  $\omega(0) = 90$ . The vertical dashed line shows the limit of the entrainment basin, we notice that convergence becomes exponential after the frequency of the oscillator enters in it.

**The case of signals with discrete spectra** We know from our previous contributions [19, 110] and from the previous section that if the input has a more complicated spectrum,  $\omega$  converges to one of the frequency components of the spectrum. We showed in [110] that given an input signal  $F(t) = A_0 + \sum_n A_n \cos(\omega_n t + \psi_n)$ , the basins of attraction corresponding to the frequencies  $\omega_i$  present in the input signal are delimited approximately by the solutions of equation

$$\frac{-A_0}{2\omega(0)} + \sum_n \frac{|A_n|^2 \omega(0)}{((n\omega_n)^2 - \omega(0)^2)} = 0 \quad (5.94)$$

This result is valid for small  $K$ . The frequency to which  $\omega$  converges depends on the initial frequency of the oscillator and the energy content  $A_n$  of each frequency present in the spectrum of  $F(t)$ . In the case of stronger coupling, the previous equation to delimit the region of convergence is not valid any more.

In order to characterize the convergence of  $\omega$  for periodic and non periodic inputs with discrete spectra, we numerically evaluated the behavior of the adaptive frequency phase oscillator for different types of input, different values of coupling and initial conditions for  $\omega$ . For each experiment, we evaluated the entrainment basins of the oscillator for a given input without frequency adaptation together with the convergence behavior of  $\omega$  when adaptation was activated. Typical results for periodic and non periodic signals with discrete spectra are shown in Figure 5.14.

From the figures, we notice that the regions of exponential convergence matches roughly the entrainment basins, as long as these entrainment basins contain the frequency to which they correspond. It must be noted that since the type of convergence



## 5.2. Properties of adaptive phase oscillators

---

(if it is exponential or not) is evaluated numerically, the delimitation of the region of exponential convergence is not exact, since in addition to the oscillations due to the frequency component to which the oscillator converges there are oscillations coming from the other frequency components. It may explain why this region exceeds a bit the regions of entrainment in our numerical simulations.

In the case of the non periodic signal (Figure 5.14(b)), the dark gray region corresponds to the case where the frequency still converges to the frequency  $30\sqrt{2}$ , but the final oscillations around this frequency are mixed with sudden jumps out the region of the frequency (then the frequency comes back to normal oscillations). This phenomena becomes more visible as coupling increases and as the entrainment basin of this frequency gets bended until the moment where the entrainment basin does not contain anymore the frequency  $30\sqrt{2}$ , then the oscillator converges to a frequency that does not correspond in average to one of the frequencies of the input.

Albeit adaptation of frequency is different from mere synchronization, it turns out that the structure of the entrainment basins is critical in the convergence of the adapted frequency. First, convergence is possible only if the entrainment basin contains the corresponding frequency. Second, when  $\omega$  enters an entrainment basin where convergence is possible, convergence is exponential.

### Tracking changing frequencies

An adaptive frequency oscillator is also able to track a time-varying frequency (i.e. non-stationary input signals). Since the average convergence of the frequency to the input frequency is exponential, with relaxation time  $\tau$  we can describe it with the following differential equation

$$\dot{\omega} \simeq \omega_F - \tau^{-1}\omega \quad (5.95)$$

If we assume that  $\omega_F$  changes with time, this equation corresponds to a low-pass filter with cutoff frequency  $\tau^{-1}$  rad · s<sup>-1</sup>. It means that the oscillator will only be able to correctly track changing frequencies such that  $\dot{\omega}_F < \tau^{-1}$ . Experimental results are shown in Section 5.3.2.

### 5.2.3 Error of convergence for the adaptive Hopf oscillator

We show in the following that due to interaction with the radius of the Hopf oscillator, the adaptation mechanism makes the system converge to a frequency that is smaller than the expected frequency. This error in convergence is only related to the fact that there is an interaction between the adaptation mechanism and the radius of the oscillator and does not relate to the fundamental properties of the adaptation mechanism. It justifies our choice of phase oscillators in this section instead of Hopf oscillators (as previously used in [19, 109, 112]) to understand the properties of the adaptation mechanism.

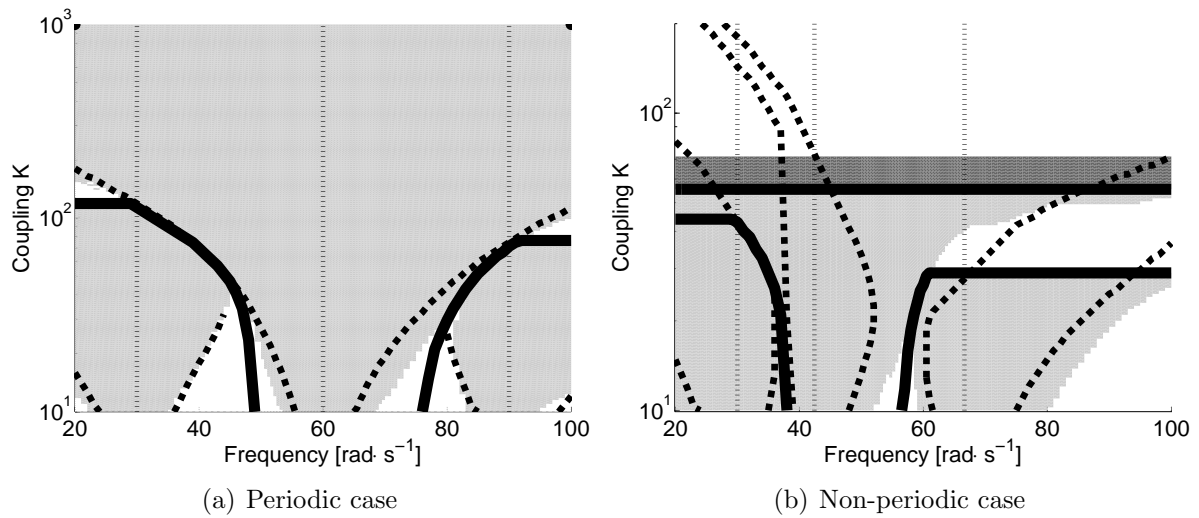


Figure 5.14: These figures shows the entrainment basins of a phase oscillator (in dashed line) for two different inputs, the vertical dotted lines represent the frequency components of the forcing signal. The light gray area represents the region where there is exponential convergence of the frequency adaptation. The thick black lines separate the region of convergence (i.e. towards which frequency component the oscillator goes). The left graph shows result for a periodic signal  $F(t) = 1.3 \cos(30t + 0.4) + \cos(60t) + 1.4 \cos(90t + 1.3)$ , the right graph shows results for a non periodic signal  $F(t) = 1.3 \cos(30t) + \cos(30\sqrt{2}) + 1.4 \cos(\frac{30\pi}{\sqrt{2}})$ . See the text for discussion of the results and an explanation of the dark gray zone of the right graph.

## 5.2. Properties of adaptive phase oscillators

---

The adaptive frequency Hopf oscillator has the following equations

$$\dot{x} = (\mu - r^2)x - \omega y + KF(t) \quad (5.96)$$

$$\dot{y} = (\mu - r^2)y + \omega x \quad (5.97)$$

$$\dot{\omega} = -KF \frac{y}{r} \quad (5.98)$$

which gives in polar coordinates the following equations

$$\dot{r} = (\mu - r^2)r + KF \cos \phi \quad (5.99)$$

$$\dot{\phi} = \omega - KF \frac{\sin \phi}{r} \quad (5.100)$$

$$\dot{\omega} = -KF \sin \phi \quad (5.101)$$

where  $\sqrt{\mu}$  is the amplitude of oscillations. When  $\omega$  has converged, the oscillator is phase-locked with the input signal. For a perturbation  $F = \sin(\omega_F t)$ , we approximately have  $\phi \simeq \omega_F t - \frac{\pi}{2}$  since the output of the oscillator is  $x = r \cos \phi$ . Thus we can calculate the behavior of  $\omega$ .

$$\dot{\omega} = -K \sin(\omega_F t) \sin \phi \quad (5.102)$$

$$\simeq \frac{K}{2} \sin(2\omega_F t) \quad (5.103)$$

Integrating this equation yields,

$$\omega \simeq \omega_0 - \frac{K}{4\omega_F} \cos(2\omega_F t) \quad (5.104)$$

Thus the frequency will oscillate around a mean value  $\omega_0$ , with frequency  $2\omega_F \text{ rad} \cdot \text{s}^{-1}$  and amplitude approximately  $\frac{K}{4\omega_F}$ .

The question now is to find the mean value  $\omega_0$ . We postulated that  $\phi \simeq \omega_F t - \frac{\pi}{2}$  thus we get

$$\omega_F \simeq \dot{\phi} \quad (5.105)$$

$$\simeq \omega - K \sin(\omega_F t) \frac{\sin \phi}{r} \quad (5.106)$$

$$\simeq \omega_0 - \frac{K}{4\omega_F} \cos(2\omega_F t) + \frac{K}{2r} \sin(2\omega_F t) \quad (5.107)$$

Thus we get

$$\Delta\omega = \omega_F - \omega_0 \quad (5.108)$$

$$\simeq -\frac{K}{4\omega_F} \cos(2\omega_F t) + \frac{K}{2r} \sin(2\omega_F t) \quad (5.109)$$

$\Delta\omega$  represents the difference between the input frequency and the frequency of the oscillator. Averaging  $\Delta\omega$  over one period will give us the mean deviation of  $\omega_0$ . Because of the  $r$  term in the second part of the equation and because  $r$  has also a perturbing function with period  $2\omega_F$ , integrating  $\Delta\omega$  over one period will not give a zero mean and thus the difference will not be zero and the adaptive frequency oscillator will not exactly converge to the correct frequency. To understand how this deviation occurs, we look at the frequency response of  $r$ . First we rewrite

$$\dot{r} = (\mu - r^2)r + K \sin(\omega_F t) \cos \phi \quad (5.110)$$

$$= \mu r - r^3 + \frac{K}{2} - \frac{K}{2} \cos(2\omega_F t) \quad (5.111)$$

and look at the system

$$\dot{r} = \mu r - r^3 + \frac{K}{2} - \frac{K}{2} u(t) \quad (5.112)$$

where  $u(t) = \cos(2\omega_F t)$ . Since the system has a cubic term, it is difficult to know its frequency response. However, we know that when  $u(t) = 0$ , the system has at maximum 3 fixed points and only one is  $> 0$ , say  $r_0$ . It is also  $> \sqrt{\mu}$  when  $K > 0$  and it is stable (by looking at the linearization around  $r_0$ ). The vector field is always pointing in the direction of  $r_0$ . Thus we postulate that the behavior of  $r$  will approximately be  $r = r_0 - A_r \cos(2\omega_F t - \gamma)$ . The amplitude  $A_r$  and the phase shift  $\gamma$  are defined by the frequency response of this linear system.

Analytically it is difficult to determine the frequency response for a nonlinear system but we can calculate it for the linearization of the system around  $r_0$ , we should have a good approximation for relatively small  $K$ . We thus have for new system

$$\dot{\tilde{r}} = (\mu - 3r_0^2)\tilde{r} + u(t) \quad (5.113)$$

whose transfer function is

$$H(s) = \frac{-K}{2(s - \mu - 3r_0^2)} \quad (5.114)$$

Numerically we show that because of the phase shift,  $\Delta\omega$  will be positive and thus  $\omega_0 < \omega_F$ . We also numerically evaluated the frequency response of the nonlinear system. Figure 5.15 shows the result of the prediction for  $\Delta\omega$  using the frequency response of the linearized system, the nonlinear one and the real values for the adaptive Hopf oscillator. It is clear from the graph that our postulate of  $\phi \simeq \omega_F t - \frac{\pi}{2}$  and the frequency response of  $r$  explain well the deviations from the expected frequency.

We justified here our preference for phase oscillators, since they have the same basic properties as Hopf oscillators but do not have the inconvenient of frequency deviation and more generally of a supplementary radius dynamics. They are then the most suitable model to understand the fundamental properties of frequency adaptation.

### 5.3. Pool of adaptive frequency oscillators

---

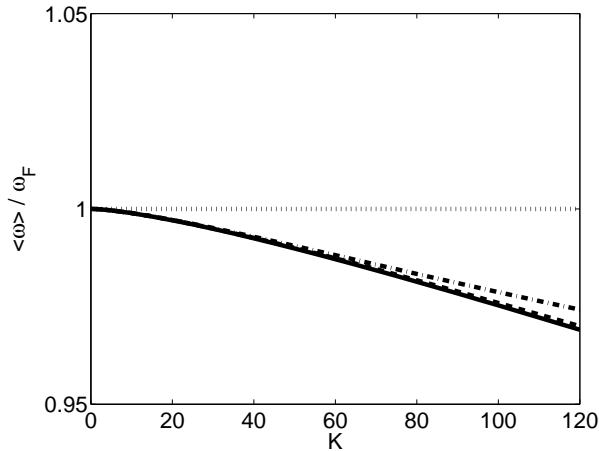


Figure 5.15: Relative mean error of convergence  $\frac{\langle \omega \rangle}{\omega_F}$  of the adaptive frequency Hopf oscillator (plain line). We also show the predictions made with the linearization of  $r$  (dash-dotted line) and the frequency response of  $r$  numerically evaluated (dashed line).

#### 5.2.4 Conclusion

In this section, we have shown analytically that the adaptive frequency phase oscillator had an exponential convergence for its frequency when entering the entrainment basin, the relaxation time being defined by  $\tau$ . We also showed numerically that the final oscillations of  $\omega$  after convergence were dependent on this relaxation time, similar to what is known as Heisenberg boxes in signal processing. However an analytical characterization of this relation is still missing. Our analysis was performed on a simple adaptive frequency oscillator (based on phase oscillators), but we know that more complex adaptive frequency oscillators can be built. We do not provide an analysis of such oscillators, but preliminary results with oscillators such as the van der Pol oscillator show that the fundamental concepts developed here (exponential convergence, the  $\tau$  parameter, the uncertainty relationship) should be qualitatively the same in other oscillators.

## 5.3 Pool of adaptive frequency oscillators

In this section we show how we can use the results of the previous section to build pools of oscillators able to do signal processing in an original way.

### 5.3.1 Motivation

When using a large number of adaptive frequency Hopf oscillators (a pool) coupled via a negative mean field, we showed in [19] that it was possible to very well approximate

the frequency spectrum of signals in real-time, ranging from signals with discrete spectra to ones with time-varying spectra and also continuous spectra. The resolution of the approximation can be made arbitrary good by increasing the number of oscillators present in the pool, although the total energy in the final spectrum is bounded by the mean field.

One interesting observation was that for time-varying spectra, the ability of the oscillators to follow changing frequencies was similar to the behavior of a low-pass filter with cutoff frequency at  $1 \text{ rad} \cdot \text{s}^{-1}$ . This means that oscillators can really well track changing frequencies as long as the rate of change of the frequencies is lower than  $1 \text{ rad} \cdot \text{s}^{-1}$ . It is an interesting observation because it suggests that behind the nonlinear nature of the adaptation mechanism there exists some linear behavior. Using the results of the previous sections, we will give in this section an explanation of this phenomena.

To overcome the limitation of the maximum energy density in the frequency spectrum and the need to use a large number of oscillators for a resolution in the frequency spectrum, we extend, in this section, the pool of oscillators by adding a weight to each oscillator in the mean field sum, and a dynamic equation for the weight, such that the system can also learn the energy content related to each frequency component of the input signal [109]. In other words, instead of needing  $N$  oscillators to fill a given “peak” in the spectrum, a single oscillator with weight is sufficient.

By also adding coupling between the oscillators it is possible to construct a system that exhibits a limit cycle that can produce as an output any periodic patterns [109] as we will show in Section 5.4. The resulting limit cycle has interesting properties such as stability and smooth modulation of the periodic pattern in frequency and amplitude by changing the frequency and weight vectors. Furthermore, the representation of the signal (and its state space) as differential equations allows us to use control theoretic tools to design a controller for biped robots [112] using the limit cycle together with feedback loops as we will show in Section 5.4.

In the following, we use the results of Section 5.2 to explain the linear behavior observed in tracking frequencies for the pool of oscillators. Then we extend our analysis to pools of oscillators with a dynamic weights that adapt to the energy content of the frequencies. We also provide numerical examples to support our analysis.

### 5.3.2 Frequency analysis with a pool of adaptive frequency oscillators

In this section, we first review the main results of our previous contribution [19] and then we discuss the performance of the system in light of the results of the previous section.

#### Frequency analysis with coupled nonlinear oscillators

The original idea of [19] is to use a pool of  $N$  Hopf oscillators coupled via a negative mean field, as it is presented in Figure 5.16. The oscillators receive as an input the difference

### 5.3. Pool of adaptive frequency oscillators

---

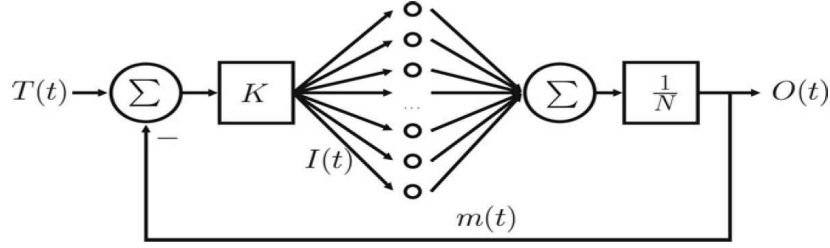


Figure 5.16: Structure of the pool of adaptive frequency oscillators that is able to reproduce a given signal  $T(t)$ . The mean field produced by the oscillators is fed back negatively on the oscillators (taken from [19]).

between the signal to analyze and the mean field produced by the pool.

In this section we use phase oscillators instead of Hopf oscillators, since their representation is simpler and they do not have the drawback of possessing a radius (see Section 5.2.3 for more details). However the main results for phase oscillators can be transposed to the case of Hopf oscillators. We also introduce the parameter  $\tau$  controlling the relaxation time, as it plays a role in the performance of the system. The evolution equations are then

$$\dot{\phi}_i = \tau^{-1}\omega_i - KI(t) \sin \phi_i \quad (5.115)$$

$$\dot{\omega}_i = -KI(t) \sin \phi_i \quad (5.116)$$

$$I(t) = T(t) - \frac{1}{N} \sum_{i=0}^N \cos \phi_i \quad (5.117)$$

The result of the frequency analysis is directly represented by the distribution of the  $\omega_i$ . Especially, we see that if this distribution is equal to the frequency spectrum of the signal to analyze,  $T(t)$ , then it is a solution of the differential equation.

The resolution of the final distribution depends on the number  $N$  of oscillators. Indeed, each oscillator contributes to the frequency spectrum in the order  $\frac{1}{N}$ . This introduces also a limitation in the type of spectra the system can analyze since if the power of the signal to analyze is higher than 1, the system will only be able to recover partially its spectrum. If the power is smaller than 1, the oscillators not participating in the recovery of the spectrum can cancel each other (for example by having out of phase oscillations).

Numerical results from [19], showed that the system can track discrete spectra as well as continuous and time-varying ones. For the performance of the frequency tracking, we numerically saw that the system was behaving like a linear low-pass filter with cutoff frequency at  $1 \text{ rad} \cdot \text{s}^{-1}$ .

### Linear behavior of the pool of oscillators

We can now explain why the system is behaving like a linear system when tracking changing frequencies and moreover we can predict that  $\tau^{-1}$  will be the cutoff frequency of the response (i.e. how fast the system can track changing frequencies). It means that the amplitude response of frequency tracking will be smaller than  $\frac{\sqrt{2}}{2}$  for spectra changing at a higher rate.

Lets consider Equations (5.115)-(5.117) with  $N = 1$  and a simple cosine input. We use  $N = 1$  and a simple cosine input for clarity of the argument, similar observations could be done in the more general case ( $N > 1$ ,  $T(t)$  arbitrary). We can then rewrite the equations as

$$\dot{\phi} = \tau^{-1}\omega - K(\cos(\omega_F t) - \cos \phi) \sin \phi \quad (5.118)$$

$$\dot{\omega} = -K(\cos(\omega_F t) - \cos \phi) \sin \phi \quad (5.119)$$

looking at the differences  $\phi_d = \phi - \omega_F t$  and  $\omega_d = \omega - \tau\omega_F$  we then get

$$\dot{\phi}_d = \tau^{-1}\omega_d - \frac{K}{2}(\sin \phi_d + \sin(\phi_d + 2\omega_F t) - \sin(2\phi_d + 2\omega_F t)) \quad (5.120)$$

$$\dot{\omega}_d = \frac{K}{2}(\sin \phi_d + \sin(\phi_d + 2\omega_F t) - \sin(2\phi_d + 2\omega_F t)) \quad (5.121)$$

Separating the time dependent fast oscillating terms from the time independent terms, we can apply the same analysis as we did in Section 5.2 for the oscillator without the feedback loop. The exponential convergence is not influenced by the negative feedback loop. This loop only influences the oscillations that adds to the exponential. These oscillations become 0 when  $\phi_d = 0$ . This shows that the negative feedback structure does not change the exponential behavior of the system, but it influences the amplitude of oscillations around the exponential. Another effect that might appear is an interaction between several oscillators through the feedback loop, indeed we often see that oscillators while converging to some frequency regroup into clusters. However the analysis of such interactions is beyond the scope of this thesis.

We can expect that the linear response of the system to changing frequencies will be the same as the adaptive phase oscillator without negative feedback loop. Figure 5.17 shows the experimental amplitude frequency response for one oscillator (note that the results would be the same for  $N > 1$ ). This response is calculated as follows, we send as input for the pool a sine wave with a time-varying frequency  $T(t) = \sin \phi$ , with  $\phi = \frac{1}{\omega_C} \sin(\omega_C t)$  so the instantaneous frequency of the signal is  $\dot{\phi} = \cos(\omega_C t)$ . During the steady-state behavior of the system, we take the complex Hilbert transform of the signal  $\frac{1}{N} \sum_N \omega_i$ , the frequency response  $H(\omega_C)$  is then this Hilbert transform divided by the Hilbert transform of  $\cos(\omega_C t)$ . We clearly see on the figure that the experimental results match very well what we predicted. The pool behaves like a low pass filter on the frequency space, its cutoff frequency being located at  $\tau^{-1}$ .



### 5.3. Pool of adaptive frequency oscillators

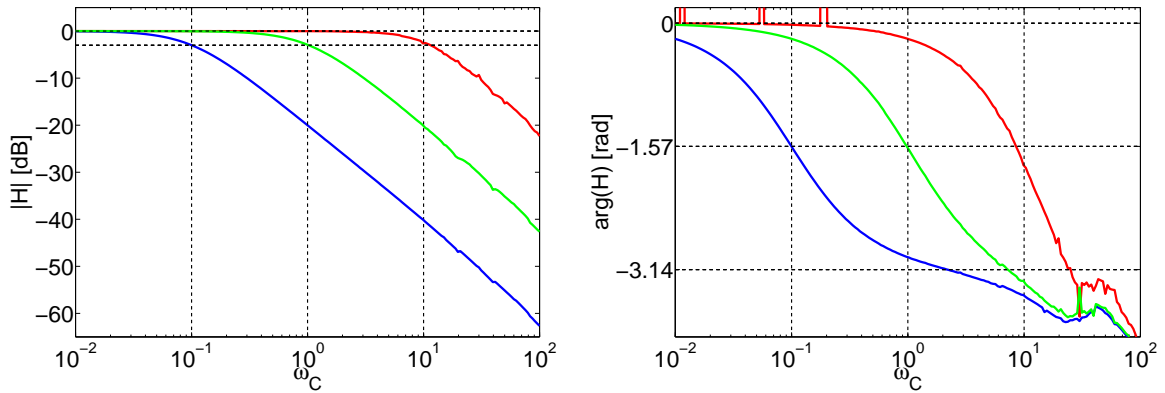


Figure 5.17: Frequency response of the pool of oscillators, the magnitude of the response is on the left figure, the phase delay on the right one (in this case  $N = 1$  but results are the same for higher values of  $N$ ).  $\tau = 1$  is represented by the green line,  $\tau = 0.1$  by the red line and  $\tau = 10$  by the blue line. The magnitudes 0dB and  $-3$ dB are represented by the two horizontal lines on the left figure. See the text for more details.

#### A word on the uncertainty relationship

We have seen in Section 5.2.2 that there was a relationship between the final amplitude of oscillations of  $\omega$  and the relaxation time  $\tau$  in the case of a single phase oscillator without the feedback loop. However, when introducing the negative feedback loop, we can in theory make the error go to 0 (if we have enough oscillators to fill the input frequency spectrum). And thus, we could think that we can use  $\tau$  as small as we want since the oscillators will converge to the correct frequencies without oscillations.

However, they cannot converge as fast as we want for free (and thus go beyond the fundamental limits of signal processing). First, because if the oscillators do not fill completely the spectrum of the signal in input (which is very likely if  $N$  is finite), all the residual frequencies will make the  $\omega_i$  oscillates with a very high amplitude (this amplitude will be related to the results of Section 5.2.2).

Second, assume that we can perfectly recover the input spectrum in theory. Then, the error will go to 0 eventually, whatever the value of  $\tau$ . However, during the transient there will still be oscillations and their amplitude will still be related to  $\tau$ . In Figure 5.18, we have made such experiments, for a sine wave as input signal with different frequencies and for one oscillator, with and without the feedback loop. From the results, we see that even though we are able to have convergence rate that depends on  $\tau$  with a 0 final amplitude of oscillations when using the feedback loop, the amplitude of the transient oscillations are still comparable to the one of the oscillator without feedback. For applications using the  $\omega$  signal in real-time, one will want to have a small relative amplitude of oscillations compared to range of working frequencies.

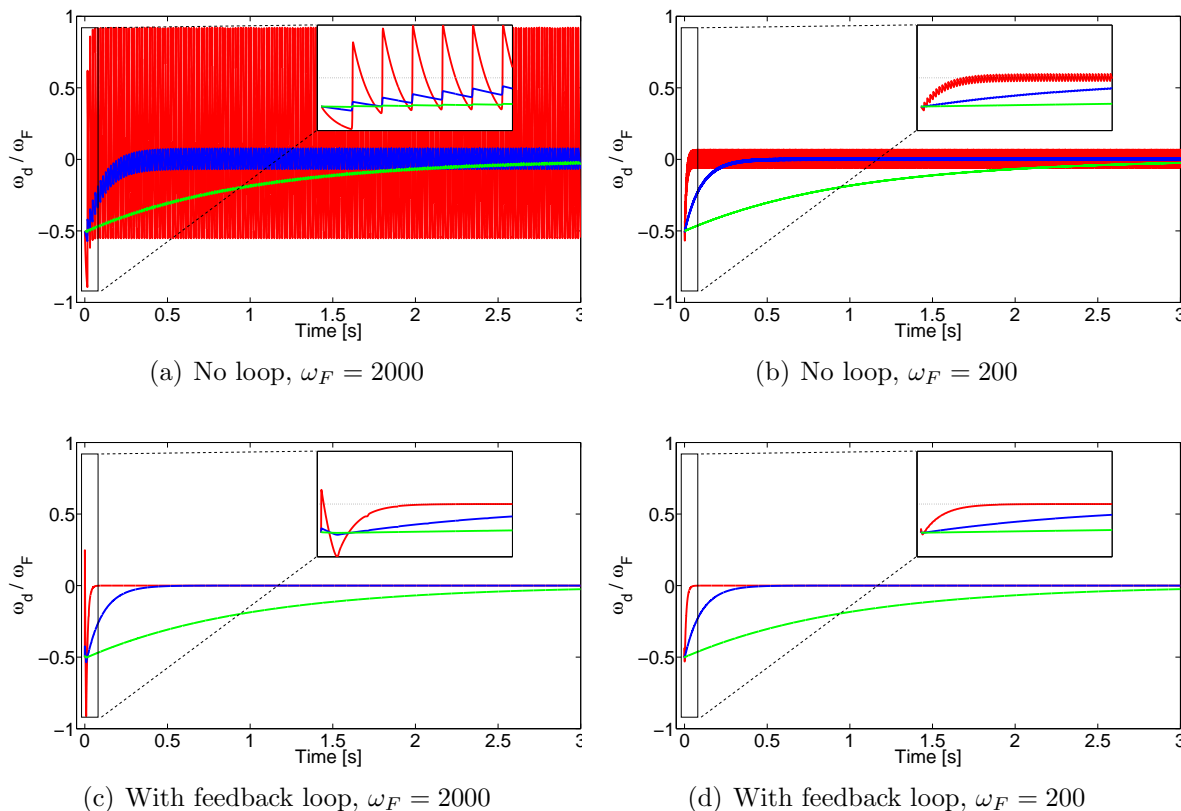


Figure 5.18: Comparative convergence behavior for a system with a single oscillator, without feedback loop ((a) and (b)) and with feedback loop ((c) and (d)). We plot the frequency differences  $\omega_d = \omega - \omega_F$  normalized by  $\omega_F$ . For each graph we show the behavior for different values of  $\tau$  (red for  $\tau = 0.01$ , blue for 0.1 and green for 1). For each experiments, we used  $\omega_D(t = 0) = 0.5$  and  $K = 10^5$ . See the text for the discussion on the results.

### 5.3. Pool of adaptive frequency oscillators

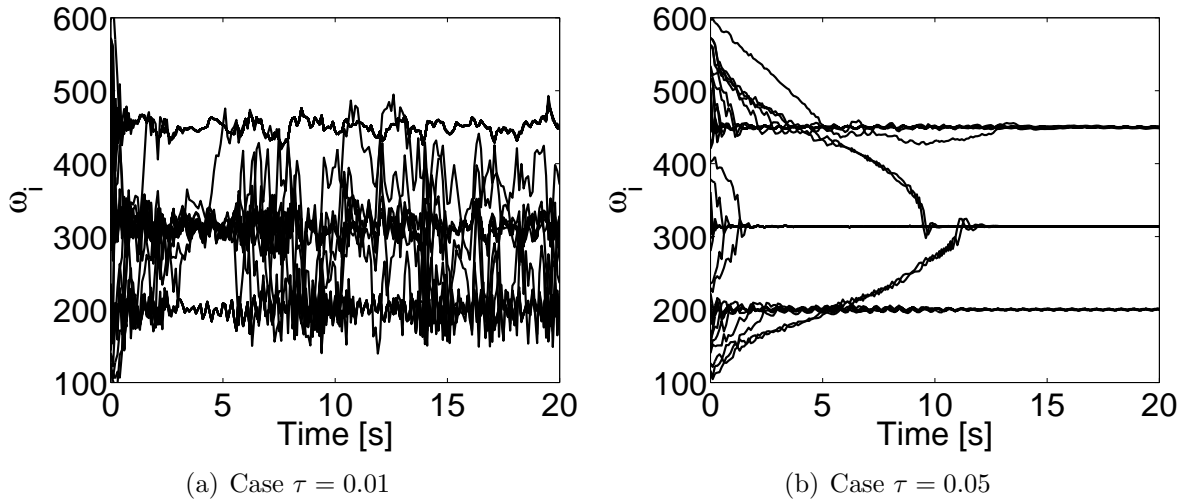


Figure 5.19: These graphs shows the convergence behavior of a pool of 50 oscillators to the frequencies of the input signal  $T(t) = 0.2 \sin(200t) + 0.4 \sin(100\pi t) + 0.4 \sin(450t)$  for two different values of  $\tau$ . In both cases we used the same initial conditions and  $K = 200$ .

Moreover, if the input signal has several frequency components (which is more realistic), then because of the large oscillations during convergence (if  $\tau$  is too small), the oscillators might never converge to the correct frequency components but keep oscillating. In Figure 5.19, we show experimental results that illustrate this fact. We used a pool of 50 oscillators to find the frequencies of an input signal for different values of  $\tau$ . The smallest frequency contained in the input has a period of  $\frac{\pi}{100}$ s, and we tested the system for a value of  $\tau = 0.01$  which is smaller than this frequency and a value of  $\tau = 0.05$  which is bigger. We see that in the first case most of the oscillators never converge to the correct frequencies, while in the second case, the oscillators fill the spectrum of the input. It shows that in general it is good to have  $\tau$  (that defines the implicit time-window) bigger than the characteristic period of the frequency to track and that we cannot use arbitrary small  $\tau$  for real applications.

#### 5.3.3 Dynamic adaptation to the energy content of the frequency spectrum

One problem with the pool of oscillators as we presented it is that the resolution of the analyzed frequency spectrum is highly dependent on the number of oscillators present in the pool and that to fill completely a frequency component, one has to wait that sufficiently many oscillators have converged to it. Indeed, one need many oscillators to fill the energy content of a frequency component. Moreover, the power of the signal to analyze must be less than 1.

In the following we present a way to associate to each oscillator a weight that will allow one oscillator to code for the whole energy content of a frequency component.

We add to each oscillator a new state variable  $\alpha_i$  that stands for its weight, then the output of the system is the weighted sum of the outputs of the oscillators, we also remove the averaging over the oscillators. The following equations describe the whole system

$$\dot{\phi}_i = \tau^{-1}\omega_i - KI(t) \sin \phi_i \quad (5.122)$$

$$\dot{\omega}_i = -KI(t) \sin \phi_i \quad (5.123)$$

$$\dot{\alpha}_i = \eta I(t) \cos \phi_i \quad (5.124)$$

$$I(t) = T(t) - \sum_{i=0}^N \alpha_i \cos \phi_i \quad (5.125)$$

where  $\eta$  is a positive constant. At the beginning the  $\alpha_i = 0$ . The dynamics of the new state variable can be seen as the correlation of the input  $I(t)$  and the output of the corresponding oscillator  $\cos \phi_i$ . When they have a frequency in common (i.e. when one oscillator is entrained by a frequency component of the input), then in average the correlation will be positive and  $\alpha_i$  will increase, but this frequency component will then disappear from  $I(t)$  because of the negative feedback, making  $\alpha_i$  converge exactly to the amplitude of the associated frequency. The other oscillators will only feel the remaining frequency components and converge to those. We see that for a discrete spectrum with a finite number of frequency components, we only need a finite number of oscillators to extract exactly the frequency spectrum with our method. We indeed see that the Fourier series decomposition of this signal is a solution of the equations such that  $I(t) = 0$  (we set the  $\omega_i$  to the frequencies of the series and the  $\alpha_i$  to the corresponding amplitudes).

### Dynamics of the new state variables

Assuming that the input signal has a discrete spectrum, we write it as  $T(t) = \sum_j A_j \cos(\omega_{F_j}t + \psi_j)$ . The dynamics of the new state variables  $\alpha_i$  can then be written as

$$\dot{\alpha}_i = \eta \left( \sum_j A_j \cos(\omega_{F_j}t + \psi_j) - \sum_k \alpha_k \cos \phi_k \right) \cos \phi_i \quad (5.126)$$

which gives

$$\begin{aligned} \dot{\alpha}_i = & \eta \left( \sum_j \frac{A_j}{2} [\cos(\omega_{F_j}t + \psi_j - \phi_i) + \cos(\omega_{F_j}t + \psi_j + \phi_i)] \right. \\ & \left. - \sum_{k \neq i} \frac{\alpha_k}{2} [\cos(\phi_k - \phi_i) + \cos(\phi_k + \phi_i)] - \frac{\alpha_i}{2} (1 + \cos 2\phi_i) \right) \end{aligned} \quad (5.127)$$

### 5.3. Pool of adaptive frequency oscillators

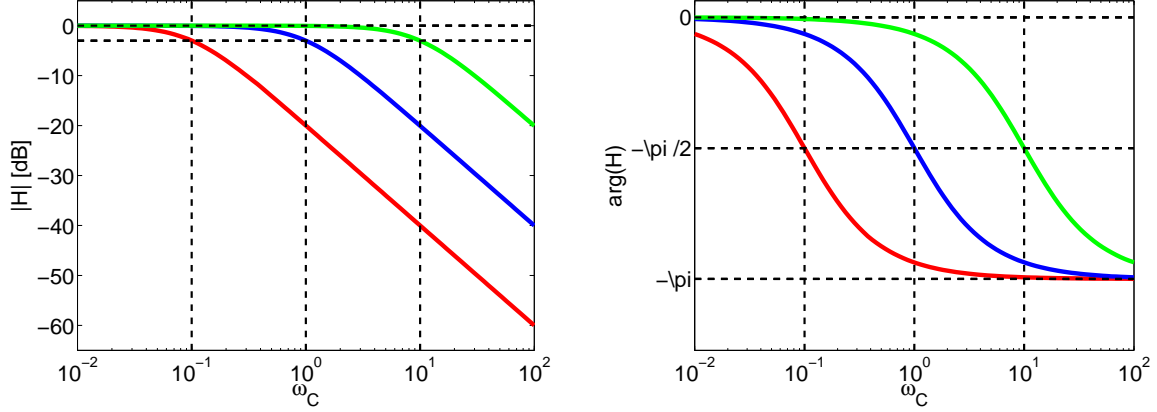


Figure 5.20: Frequency response of the amplitude adaptation  $\alpha_i$ . The left graph shows the magnitude and the right graph the delay of the response. Red line is for  $\eta = 0.2$ , the blue one for  $\eta = 2$  and the green one for  $\eta = 20$ . The vertical lines of the left graph show the magnitudes 0 and  $20 \log_{10} \frac{\sqrt{2}}{2}$ . We used  $K = 100$ ,  $\tau = 1$  and  $\omega_F = 1000$ .

When the oscillator  $i$  has not converged to any frequency component of the input, the right hand side of the equation is composed of oscillating terms (fast and slow) and of a non oscillating term that makes  $\alpha_i$  go to 0. So in this case the dynamics is in average exactly what we want, since there is no energy related to frequency  $\omega_i$ .

In the case where the oscillator  $i$  has converged to a frequency component of the input, or at least when it is in the corresponding entrainment basin (i.e.  $\phi_i \simeq \omega_{F_j} t + \psi_j$  for some  $j$ ), there is one oscillating term in the first sum that becomes constant and Equation 5.127 can be rewritten as

$$\dot{\alpha}_i = \frac{\eta}{2}(A_j - \alpha_i) + O.T. \quad (5.128)$$

where  $O.T.$  stands for oscillating terms. Thus  $\alpha_i$  converges exponentially fast to the correct amplitude  $A_j$ . The relaxation time is then  $\frac{2}{\eta}$ . In the case where several oscillators have already converged to the same frequency, Equation 5.127 reads

$$\dot{\alpha} = \frac{\eta}{2}(A_j - \sum_k \alpha_k - \alpha_i) + O.T. \quad (5.129)$$

where the sum is to be taken over the oscillators that have converged to the frequency  $\omega_{F_j}$ . There is still exponential convergence, but in this case  $\alpha_i$  converges to the remaining amplitude that was not taken by the other oscillators.

In order to confirm the linear behavior in average of the  $\alpha_i$  and its exponential convergence, we measured the frequency response of this variable when the amplitude of a sine wave is modulated at a certain frequency. We use only one oscillator in this experiment

and the input signal is  $(1 + \cos(\omega_C t)) \cos(\omega_F t)$ , where  $\omega_C$  is the frequency of variation of the amplitude. We choose  $\omega_F \gg \omega_C$ , since the representation of the sine wave with a time-varying amplitude is not unique and there might be an interaction between the frequency and amplitude adaptations, which we do not want. Figure 5.20 shows the result of the experiments. We clearly see that the system acts as a low pass filter and that the cutoff frequency is equal to  $\frac{\omega}{2}$  as we predicted previously. However, we do not take into account several oscillators and the possible interactions between these oscillators is still to be analyzed but is beyond the scope of this thesis. It seems that this interaction is nonlinear and is obviously not easy to understand. Nevertheless this first analysis gives an idea of the average behavior of the system and experimental tests showed us that the interaction between the oscillators becomes critical in limiting cases (e.g. when the relaxation time of the  $\alpha_i$  is smaller than the period of oscillations of the input to analyze).

### Examples

In this section we give examples of the behavior of the system when tracking the frequencies of different types of signals.

**Discrete spectra** The first example we show is to track spectrum of the signal  $T(t) = 1.3 \cos(30t) + \cos(30\sqrt{2}t) + 1.4 \cos(\frac{30\pi}{\sqrt{2}}t)$  that we already used in Section 5.2.2. We tested the system for two values of  $K$  to show the behavior of the oscillators where they were starting or not in the entrainment basins corresponding to the input. We use exactly 3 oscillators to show that they are sufficient to perfectly recover a spectrum consisting of 3 frequencies. The results are shown in Figure 5.21. We show the evolution of the state variables  $\omega_i$  and  $\alpha_i$  together with the absolute difference between the input signal  $T(t)$  and the output of the oscillators  $O(t)$ . We also show the spectral distance, which is the distance between the spectrum of the input and the spectrum defined by the  $\omega_i$  and  $\alpha_i$  variables. We assume that two frequencies are equal if they differ by less than 1%. Note that in general to calculate the amplitude associated to one frequency component, one has to take into account the phase differences between the oscillators into consideration.

For the case  $K = 10$ , the initial conditions of the frequencies are out of the entrainment basins, we see that the convergence is not exponential at the beginning. We also see that the amplitudes  $\alpha_i$  start to increase only when the corresponding oscillator's frequency matches the correct input frequency. We also see the exponential convergence of the  $\alpha_i$ . Interestingly we see that the red  $\omega_i$  crosses the frequencies already filled by the other oscillators. Note that it might not always be the case and sometimes several oscillators might code the same frequency component, thus it is generally better to have a higher number of oscillators than frequencies that one wants to recover.

We also show the case where  $K = 100$ , because it shows that the oscillators can go much faster in learning the frequency spectrum of an input (in less than 5s which corresponds to less than 50 periods of the smallest frequency) even when they start

### 5.3. Pool of adaptive frequency oscillators

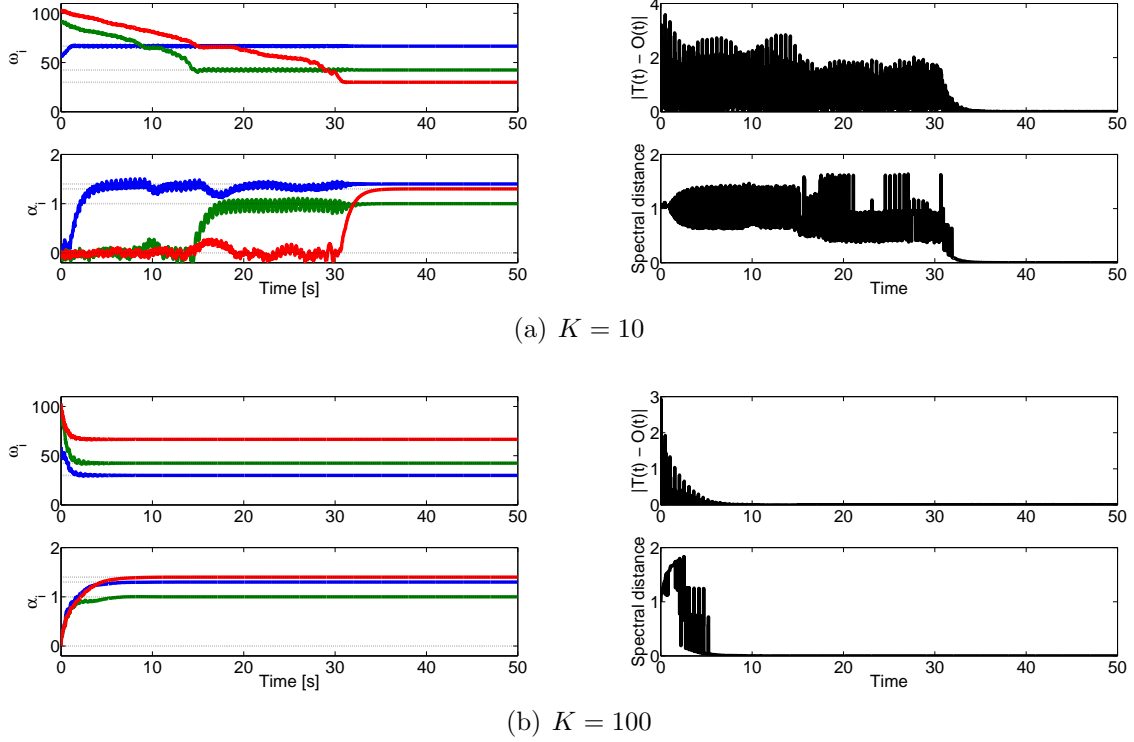


Figure 5.21: Examples of decomposition of the spectrum of an input signal  $T(t) = 1.3 \cos(30t) + \cos(30\sqrt{2}t) + 1.4 \cos(\frac{30\pi}{\sqrt{2}}t)$  with a pool of  $N = 3$  oscillators for two different coupling strengths. The parameters used in the simulations are  $\tau = 0.5$  and  $\eta = 2$ ,  $K = 10$  (top) and  $K = 100$  (bottom). Refer to the text for more details.

quite far from the desired frequencies (the red  $\omega_i$  starts at more than 100 to converge to  $30 \text{ rad} \cdot \text{s}^{-1}$ ). Second, this is the same coupling as in the open loop case where only one oscillator did not manage to get one of the frequency components of the input as we explained in Section 5.2.2 because the entrainment basins were not containing the corresponding frequencies anymore. Interestingly, in the system with the feedback loop, because of the interactions between the oscillators via the mean field and the fact that an increase of the  $\alpha_i$  induces a decrease in coupling strength, the system can learn correctly the frequency spectrum of the input.

**Time-varying spectra** So far we have discussed a simple example to show the basic properties of the system. Now we show an application where the system tracks a time varying spectrum, with appearing and disappearing frequency components, in order to give an idea on the capabilities of the pool.

Figure 5.22 shows such an example. It is composed of one ascending linear chirp, one

descending quadratic chirp and two frequency-modulated gaussians. It is an interesting example because it needs both the frequency and amplitude tracking capabilities of the system.

The upper graph shows the frequency distribution of the pool of oscillators as a function of time. This representation gives the same information as a spectrogram resulting from a windowed Fourier transform. We see that the system is able to track the chirps and to appropriately locate the gaussians. Thus all the important features of the signal are visible.

Second, we also notice that the error between the output of the pool and the input is almost always 0, except when a new component appears (the gaussian) or when the chirps cross, but still the match is quite good.

Third, the time evolution of the  $\omega_i$  and  $\alpha_i$  shows that oscillators that are not used to encode the chirps are recruited when an event appears (the gaussians). We can also see the clusters of frequencies and amplitudes that represent the different signals.

### Importance of the choice of parameters

There are 4 parameters to choose when using the pool of oscillators, the number of oscillators  $N$ , the coupling strength  $K$ , the relaxation time of the frequency variables  $\tau$  and the relaxation time of the weights  $\frac{2}{\eta}$ . The choice is very important since it can either degrade completely the performance of the system or be such that the oscillators never converge to the correct frequencies.

For the number of oscillators, it defines the maximum number of frequencies that the system can identify, so in general the higher the better. However the number should be small enough that real time computations are still possible.

The coupling strength will mainly define the width of the entrainment basins, thus a high value is desirable since it will allow an exponential convergence from many initial conditions. However a too high value of  $K$  will hide some frequency components and if all the oscillators converge to some frequencies, then the others will not be represented.

The  $\tau$  parameter, from our experience, should be chosen such that it is higher than the period of the frequencies to track. It seems also that  $\eta$  should be chosen such that  $\frac{2}{\eta} > \tau$ . However to rigorously set rules to choose these different parameters, we still need to make a deeper analysis of the system, particularly the influence of the interactions between the oscillators, but this analysis is out of the scope of this thesis.

### 5.3.4 Conclusion

In this section we presented a system to perform a kind of dynamic Fourier series decomposition. To the best of our knowledge this is a completely novel way of implementing a Fourier series decomposition. The system is able to find the frequencies and associated amplitudes of an input signal in a dynamic manner. The performances of the system in



### 5.3. Pool of adaptive frequency oscillators

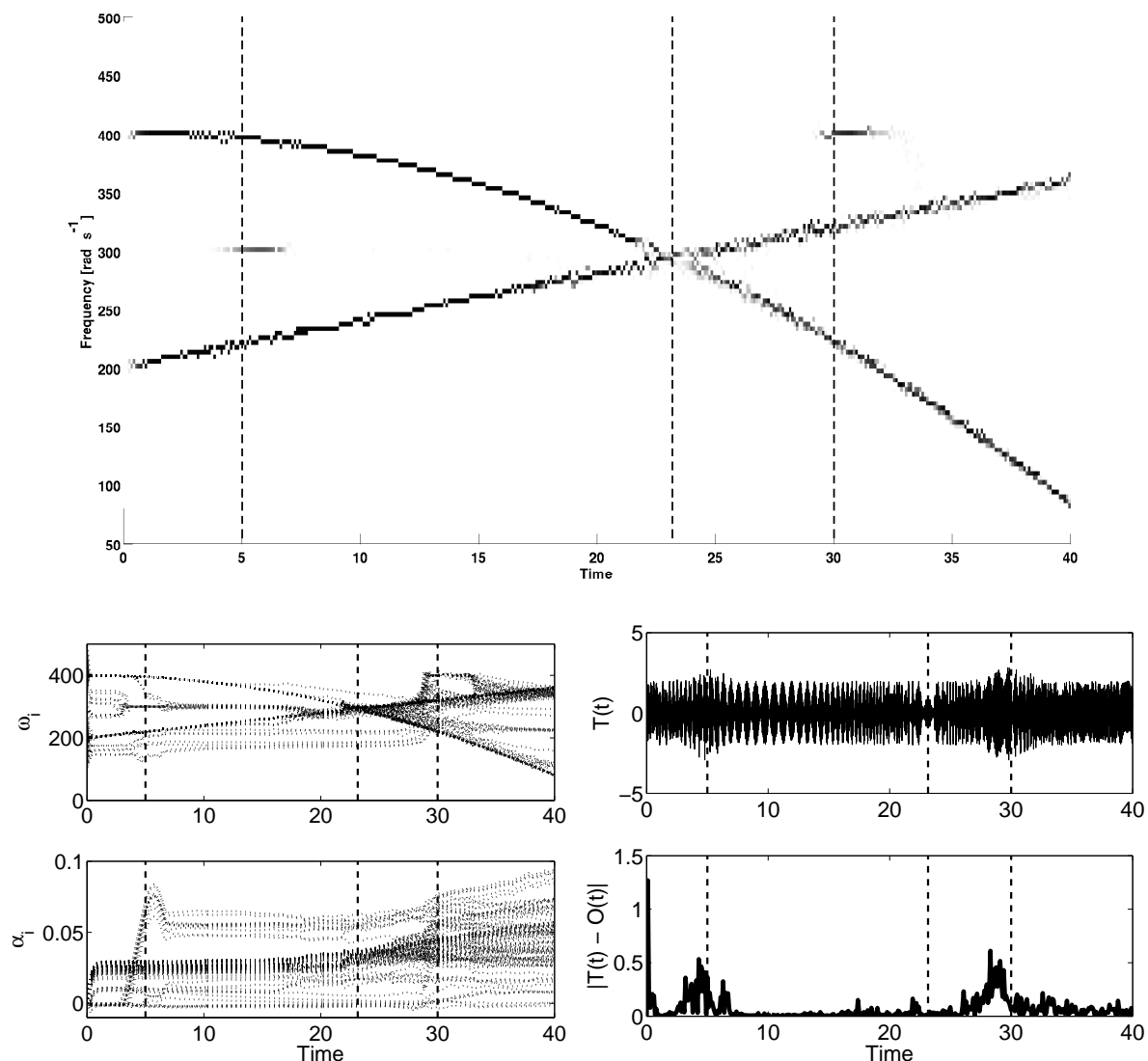


Figure 5.22: These graphs show the result of analysis of a signal with a time-varying spectrum using a pool of adaptive frequency oscillators, using amplitude adaptation. The input signal is composed of one ascending linear chirp  $\sin(200t + 2t^2)$ , one descending quadratic chirp  $\sin(400t - \frac{t^3}{15})$ , and two frequency modulated Gaussians located at  $t = 5$  and  $30$ :  $\sin(300t) \exp(-\frac{(t-5)^2}{2.5})$  and  $\sin(400t) \exp(-\frac{(t-30)^2}{5})$ . The pool is composed of  $N = 100$  oscillators,  $\tau = 0.05$  and  $\eta = 0.2$ . The upper figure shows the frequency distribution of the oscillators weighted by their respective amplitude as a function of time. The lower left graph shows the evolution of the  $\omega_i$  and  $\alpha_i$  variables and the lower right graph shows the input signal  $T(t)$  and the difference between the output of the pool and the input. The vertical dashed bars signal the important event in time: the maximum of the 2 Gaussians and the crossing of the chirps.

tracking changing frequencies and amplitudes are characterized by the parameters  $\tau$  and  $\eta$  and their behavior can be assimilated to lowpass filters. There are still open questions for these systems, as for example how to choose the different parameters for a given application. To answer these questions, an analysis of the interactions between the oscillators might be needed. Then a systematic analysis of these properties could be used to specify for which applications our approach could be competitive compared to traditional signal processing approaches. We must note that we do not intend to compete with state of the art signal processing methods at this point, but to show that it is possible to implement similar processing into a dynamical system. A strength of our system is that it is completely distributed and could be implemented on an analog electronic device using standard components such as phase-locked loops.

Investigating a pool of oscillators with different  $\tau$  and  $\eta$  for different oscillators would also be interesting. Then we could have different oscillators for different ranges of frequencies and time resolutions. In our approach, we decompose the signal using a basis made of sines (because we use a phase oscillator), it would be interesting to see how changing this basis would change the performance of the system (e.g. relaxation oscillators) and maybe to see if it is possible to find a basis that would be more similar to a wavelet basis.

Although the design of the pool of oscillators was mainly driven by scientific curiosity, these concepts could be used in real applications. For example the pool of oscillators, where coupling between the oscillators is added, can be used to construct limit cycles for robotics control [112] as we will show in the next section. The representation of a periodic trajectory with a surrounding state space as a set of differential equations can be very useful in control because then tools from control theory can be used.

## 5.4 Programmable central pattern generators

In this section we show how to construct limit cycles with arbitrary shape using the pools of adaptive frequency oscillators that we presented in the previous section with in addition coupling between the oscillators. This can serve to construct programmable CPGs and we present an application of these CPGs to biped locomotion. The goal of this section is to show a real application of the concept of programmable CPGs. Therefore the application to biped locomotion must be taken as an example and not as the central aspect of this work. This work was originally published in [112].

### 5.4.1 Motivation

This section presents a programmable Central Pattern Generator (CPG) for the online generation of periodic trajectories, and its application to the control of biped locomotion in a simulated Hoap-2 robot.

## 5.4. Programmable central pattern generators

---

As an alternative to methods using pre-recorded trajectories (e.g. ZMP-based [147]) and methods using heuristic control laws (e.g. Virtual Model control [106]), CPGs encode rhythmic trajectories as limit cycles of nonlinear dynamical systems, typically systems of coupled nonlinear oscillators. This offers multiple interesting features such as the stability properties of the limit cycle behavior (i.e. perturbations are quickly forgotten), the smooth online modulation of trajectories through changes in the parameters of the dynamical system, and entrainment phenomena when the CPG is coupled with a mechanical system. Interesting examples of CPGs applied to biped locomotion include [135, 41].

One drawback of the CPG approach is that most of the time these CPGs have to be tailor made for a specific application, and there are very few methodologies to construct a CPG for generating an arbitrary periodic signal. In [71], a method is presented which uses regression techniques to shape limit cycles of nonlinear dynamical systems, but that method requires preprocessing the teaching signal to extract its main period.

In this section, we present a novel system of coupled adaptive frequency oscillators that can learn arbitrary periodic signals in a supervised learning framework. An interesting aspect of our approach is that the learning is completely embedded into the dynamical system, and does not require any external regression or optimization algorithms, nor any preprocessing of the teaching signal. The system essentially implements a kind of dynamic Fourier series representation. We apply our system to the control of locomotion of a 23-DOF simulated humanoid robot. Results are presented demonstrating how pre-recorded walking trajectories can be learned with the system and then modulated online using the CPGs limit cycle properties. In particular, we show how sensory feedback can be integrated into the CPGs to increase the basin of stability of the gaits, and how the speed of walking can be modulated and even reversed by using a single control parameter.

### 5.4.2 Generic Central Pattern Generators

In this section we present our model of a generic CPG that we use to encode periodic trajectories. First, we present in details the architecture of the CPG which is made of adaptive oscillators and then we discuss the intrinsic properties of the system that makes it suitable for periodic movement control.

#### Architecture of the CPG

The basic building block of our generic CPG is the adaptive frequency Hopf oscillator that we developed in the previous sections

$$\dot{x} = \gamma(\mu - r^2)x - \omega y + \epsilon F(t) \quad (5.130)$$

$$\dot{y} = \gamma(\mu - r^2)y + \omega x \quad (5.131)$$

$$\dot{\omega} = -\epsilon F(t) \frac{y}{r} \quad (5.132)$$

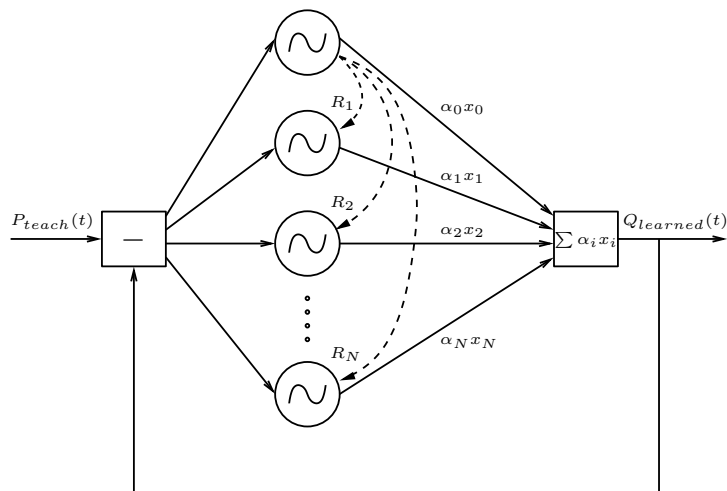


Figure 5.23: Structure of the network of adaptive Hopf oscillators. Each oscillator receives the same learning signal  $F(t) = P_{teach}(t) - \sum_i \alpha_i x_i$ , which is the difference between the signal to be learned,  $P_{teach}(t)$ , and the signal already learned,  $Q_{learned}(t) = \sum_i \alpha_i x_i$ . Then all the oscillators (except oscillator 0) receive the scaled phase input  $R_i$  from oscillator 0. Refer to Equations (5.133)-(5.137) and to the text for more details.

where  $r = \sqrt{x^2 + y^2}$ ,  $\mu$  controls the amplitude of the oscillations,  $\gamma$  controls the speed of recovery after perturbation,  $\omega$  controls the frequency of the oscillations,  $F(t)$  is a periodic input to which the oscillator will adapt its frequency and  $\epsilon > 0$  is a coupling constant. Its frequency will adapt to one of the frequency component of the input  $F(t)$ . The frequency component adapted will depend on the initial conditions for  $\omega$ .

**Generic CPG** The basic idea for constructing the generic CPG is to use coupled adaptive oscillators to reproduce a periodic signal [109] as we shown in the previous section. The output of a CPG is usually multidimensional but in this section we present a network of coupled oscillators to encode one dimension. However, we will show in Section 5.4.3 how we can use several coupled generic CPGs to encode multidimensional trajectories.

We use a network of oscillators with the additional variables representing the amplitudes of the learned frequencies, then the output of the CPG will be the weighted sum of the outputs of the oscillators with the associated amplitude variables. Moreover, we associate to each oscillator a variable encoding for the phase difference between the oscillator and the first oscillator of the network, thus enabling us to reproduce any phase relationship between the oscillators. Figure 5.23 shows the structure of the network. The

## 5.4. Programmable central pattern generators

---

equations describing this CPG are as follow

$$\dot{x}_i = \gamma(\mu - r_i^2)x_i - \omega_i y_i + \epsilon F(t) + \tau \sin\left(\frac{\omega_i}{\omega_0}\theta_0 - \theta_i - \phi_i\right) \quad (5.133)$$

$$\dot{y}_i = \gamma(\mu - r_i^2)y_i + \omega_i x_i \quad (5.134)$$

$$\dot{\omega}_i = -\epsilon F(t) \frac{y_i}{r_i} \quad (5.135)$$

$$\dot{\alpha}_i = \eta x_i F(t) \quad (5.136)$$

$$\dot{\phi}_i = \sin\left(\frac{\omega_i}{\omega_0}\theta_0 - \theta_i - \phi_i\right) \quad (5.137)$$

with

$$\theta_i = \text{sgn}(x_i) \cos^{-1}\left(-\frac{y_i}{r_i}\right) \quad (5.138)$$

$$F(t) = P_{\text{teach}}(t) - Q_{\text{learned}}(t) \quad (5.139)$$

$$Q_{\text{learned}}(t) = \sum_{i=0}^N \alpha_i x_i \quad (5.140)$$

where  $\tau$  and  $\epsilon$  are coupling constants and  $\eta$  is a learning constant. The output of the CPG,  $Q_{\text{learned}}$ , is the weighted sum of the outputs of each oscillator.  $F(t)$  represents the negative feedback, which in average is the remaining of the teaching signal  $P_{\text{teach}}(t)$  the CPG still has to learn.  $\alpha_i$  represents the amplitude associated to the frequency  $\omega_i$  of oscillator  $i$ . Its equation of evolution maximizes the correlation between  $x_i$  and  $F(t)$ , which means that  $\alpha_i$  will increase only if  $\omega_i$  has converged to a frequency component of  $F(t)$  (the correlation will be positive in average) and will stop increasing when the frequency component  $\omega_i$  will disappear from  $F(t)$  because of the negative feedback loop.  $\phi_i$  is the phase difference between oscillator  $i$  and 0. It converges to the phase difference between the instantaneous phase of oscillator 0,  $\theta_0$ , scaled at frequency  $\omega_i$  and the instantaneous phase of oscillator  $i$ ,  $\theta_i$ . Each adaptive oscillator is coupled with oscillator 0, with strength  $\tau$  to keep correct phase relationships between oscillators, using a diffusive coupling inspired by the Kuramoto coupling scheme [3] to achieve phase synchronization. We mention that with this coupling scheme, the system is more than just a dynamic Fourier series representation because the oscillators can have any phase relationship and not only 0,  $\frac{\pi}{2}$ ,  $\pi$  or  $\frac{3\pi}{2}$  phase differences.

With this generic architecture, we are able to learn any periodic input signal. We just have to provide  $P_{\text{teach}}$  the periodic trajectory we want to learn as input and integrate the system of equations. After convergence, we can set  $F(t) = 0$  (no more input nor feedback loop) and the periodic signal stays encoded into the network of oscillators. The learning process is embedded in the equations, there is no need of any external optimization or learning algorithm. In Section 5.4.3 we will see how this concept of generic CPG can be extended to learn multidimensional signals.

### Properties of the generic CPG

In this section, we present a numerical experiment where the generic CPG learns a simple signal  $P_{\text{teach}} = 0.8 \sin(15t) + \cos(30t) - 1.4 \sin(45t) - 0.5 \cos(60t)$ . The network we use is composed of 4 oscillators. Figure 5.24 shows the result of the experiment. An interesting aspect of this generic CPG is that the frequencies of the oscillators are first adapted, each oscillator converges to one of the frequency component 15, 30, 45 and 60. Only when an oscillator matches the frequency of the teaching signal is the corresponding amplitude adapted and then the corresponding frequency component disappears from the signal  $F(t)$ , as can be seen by the sudden decrease in the error. The phase variables stabilize when the involved oscillators have their frequencies correctly tuned. After learning, the periodic signal is encoded in the network of oscillators, as can be seen in Figure 5.24(a).

If there are not enough oscillators to code for all the frequency components of the teaching signal, the system will only learn the frequency components with the more power. Thus, the learned trajectory will only be an approximation of the teaching one. However, if there are more oscillators than frequency components to learn, either some oscillators will not converge to any frequency and their contribution to the learned signal will be null ( $\alpha = 0$ ) or some frequency components will be coded by several oscillators and the sum of the corresponding  $\alpha_i$  will match the amplitude of the frequency component.

This generic CPG possesses intrinsic properties of stability that are inherent to the Hopf oscillator, which has a structurally stable limit cycle. The CPG can thus produce trajectories that are stable to perturbations. This can be useful when integrating sensory feedback in the CPG to be sure that the sensory information will be forgotten as soon as it disappears from the environment.

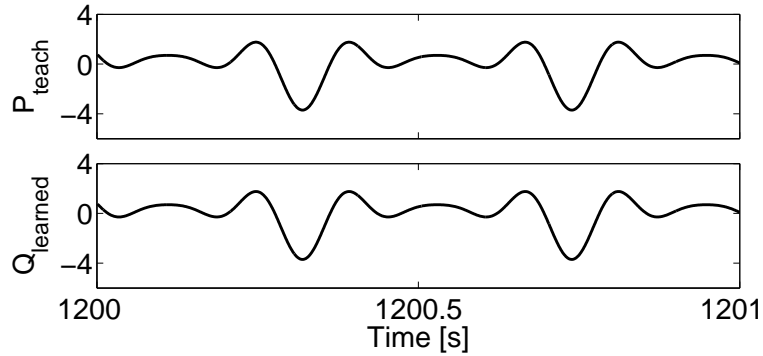
Another important aspect of the CPG is that it allows easy modulation of the amplitude and the frequency of the trajectory. Since the frequency and amplitude are linearly related to the vectors  $\vec{\omega}$  and  $\vec{\alpha}$ , simple modulation of these vectors can generate an infinite variation of stable trajectories from the learned input. Because of the properties of coupled oscillators, modulation of these parameters is always smooth and thus interesting for trajectory generation in a robot. Some of these properties are shown in Figure 5.25.

We have now introduced our generic CPG that can encode periodic inputs as stable limit cycles. In the next section, we show an application of this generic CPG as a controller for a humanoid robot. To prove the usefulness of the architecture, we apply it to the control of biped locomotion.

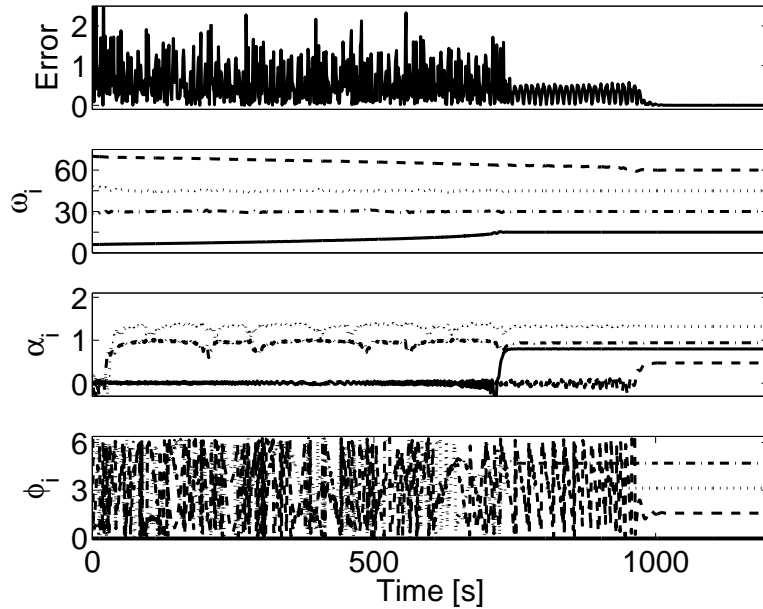
### 5.4.3 Application to biped locomotion

In this section we show how, given a sample trajectory, we can use our generic CPG architecture to control biped locomotion on a simulation of the Hoap-2 (a 25-DOF humanoid robot built by Fujitsu). First, we present the controller architecture made of several coupled generic CPGs, one for each DOF. Then we show how we can easily integrate sensory

## 5.4. Programmable central pattern generators

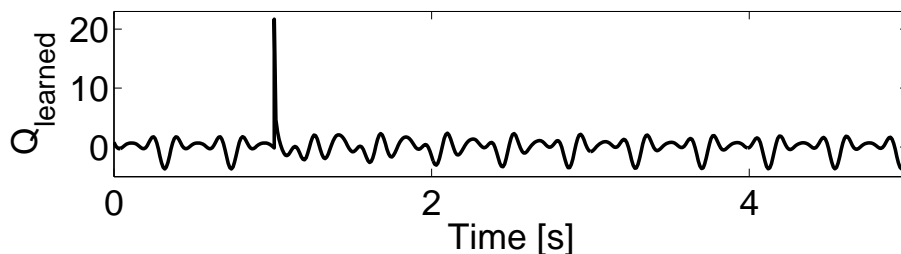


(a) Result of learning

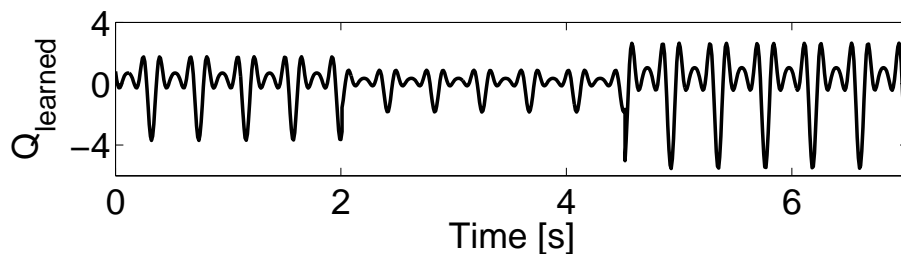


(b) Evolution of the state variables of the CPG

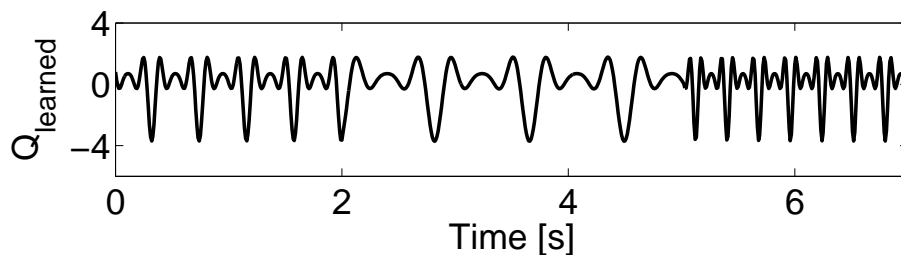
Figure 5.24: Figure 5.24(a) shows the input signal to learn,  $P_{teach}$ , in the upper graph and the result of learning  $Q_{learned}$  in the lower graph. It is obvious that the network correctly learned the input pattern. Figure 5.24(b) shows the evolution of the state variables of the generic CPG during learning of an input signal ( $P_{teach} = 0.8 \sin(15t) + \cos(30t) - 1.4 \sin(45t) - 0.5 \cos(60t)$ ) and the evolution of the error of learning. The upper graph is a plot of the error, defined by  $error = \|P_{teach} - Q_{learned}\|$ . The 3 other graphs show the evolution of the frequencies,  $\omega_i$ , the amplitudes,  $\alpha_i$  and the phases,  $\phi_i$ . The variables for each oscillator are plotted, variables of oscillator 0 are the plain lines, variables for oscillator 1 are the dotted-dashed lines, variables for oscillator 2 are the dotted lines and the dashed lines represent oscillator 3. The initial conditions are  $\alpha_i(0) = \phi_i(0) = 0$ ,  $x_i(0) = 1$ ,  $y_i(0) = 0 \forall i$ ,  $\mu = 1$ ,  $\gamma = 8$ ,  $\epsilon = 0.9$ ,  $\eta = 0.5$  and  $\tau = 2$ . The frequencies  $\omega_i(0)$  are uniformly distributed from 6 to 70.



(a) Stability against perturbation



(b) Modulation of amplitude



(c) Modulation of frequency

Figure 5.25: Figure 5.25(a) presents the evolution of the output of the generic CPG when perturbed. At time  $t_p = 1$  a perturbation occurs on all the oscillators of the CPG. We clearly see that the CPG quickly recovers its original behavior, thus proving the stability properties of the system. Figure 5.25(b) shows the behavior of the system when the amplitude  $\vec{\alpha}$  is changed. At time  $t = 2$ , the amplitude is divided by 2 and at time  $t = 4.5$  the amplitude is multiplied by 3. Figure 5.25(c) shows the behavior of the network when the frequency  $\vec{\omega}$  is changed. At time  $t = 2$  the frequency is divided by 2 and at time  $t = 5$  frequency is multiplied by 3. In both graphs, we can notice the smoothness of the trajectory when the parameters are changed.



## 5.4. Programmable central pattern generators

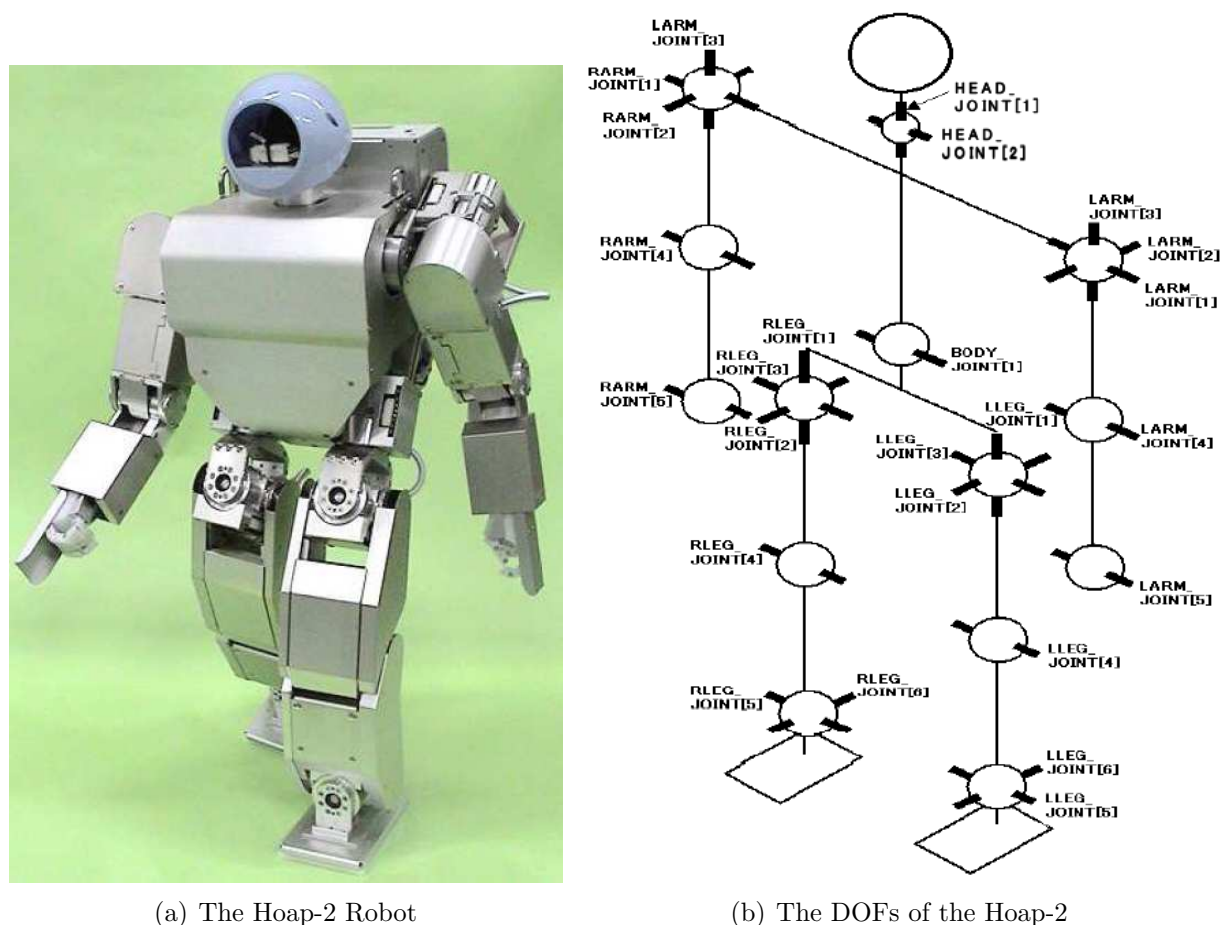


Figure 5.26: Real Hoap-2 robot (Fig. (a)) and schematic of its DOFs (Fig. (b)), this pictures were taken from [1]. We can directly see which DOF the CPG of Figure 5.27 controls on the schematic.

feedback in the controller that generates the trajectories. The lower level control is done by a PID controller.

### The controller architecture

In our controller architecture, we control 10 of the 25 DOFs of the robot. For the moment, the arms have fixed position. We control 2 of the 3 DOFs of each hip, the 3rd one which controls vertical rotation is not used. We also control the DOFs of the knees and the ankles. Figure 5.26 shows a schematic view of the Hoap-2 robot and its DOFs.

We use one generic CPG for each controlled DOF, each CPG is made of 3 oscillators as can be seen in Figure 5.27. For coordinating these several DOFs, for each leg we use a chain coupling from the hip to the ankle of the first oscillator of each CPG. And we

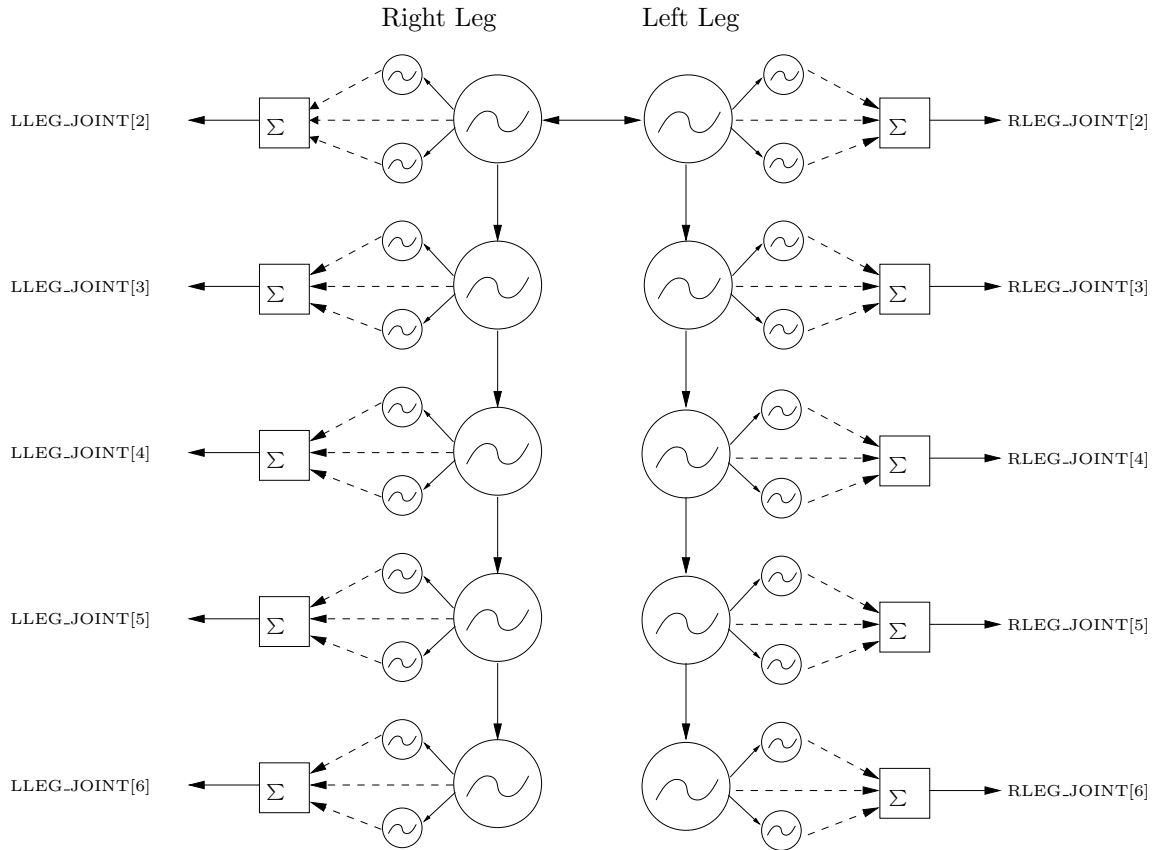


Figure 5.27: Structure of the CPG for the humanoid. We use a generic CPG as presented in Section 5.4.2 for each DOF of the legs. We also add state variables that will learn the phase differences between the generic CPGs of the legs (the descending arrows). Antisymmetric coupling is also done between the 2 legs through the main oscillator of the first DOF of each leg (horizontal arrow). The trajectories generated for each DOF is the weighted sum of the corresponding 3 oscillators.

## 5.4. Programmable central pattern generators

---

add a symmetric coupling between the first oscillators of the Hip2 joints of each leg, to conserve a  $\pi$  phase difference between the legs.

The coupling scheme to keep correct phase differences between the DOFs of one leg is similar to the one we presented in Section 5.4.2, for the oscillators of one generic CPG. The phase difference between 2 DOFs is also learned using the same evolution equation as in Equation 5.137. The equation of coupling between the oscillators and the learning rule for the phase difference are as follow

$$\dot{x}_{0,k} = (\mu - r^2)x_{0,k} - \omega_{0,k}y_{0,k} + \tau \sin(\theta_{0,k} - \phi_{0,k}) \quad (5.141)$$

$$\dot{\phi}_{0,k} = \sin(\theta_{0,k-1} - \theta_{0,k} - \phi_{0,k}) \quad (5.142)$$

where  $(0, k)$  denotes the first oscillators of the  $k$ th CPG. The other terms are the same as defined in Equations (5.133)-(5.137). Thus, in addition to the 10 generic CPGs made of 3 oscillators, we add 8 new state variables to the system that will learn the correct phase difference between the CPGs of each DOF. Figure 5.27 shows the architecture of the controller.

We trained the generic CPGs with sample trajectories of walk motion of the Hoap-2 robot provided by Fujitsu. Each trajectory was a teacher signal to the corresponding CPG controlling the associated DOF. All the control parameters of the CPGs converged correctly and, after learning, the sample trajectories are encoded in the controller as can be shown in Figure 5.28. We clearly see that the learning was successful since the learned trajectories match well the sample trajectories. The system is able to generate the learned trajectories and moreover we benefit of properties of the CPGs, such as limit cycle behavior, amplitude and frequency modulation and the possibility to add feedback pathways. Now, online trajectory generation rather than following fixed trajectory is possible.

### Feedback pathways

In this section we introduce three kinds of feedback pathways. We discuss each of them in the following.

**Lateral stability** The first feedback pathway we introduce is for maintaining lateral stability during locomotion. This feedback pathway is inspired by the vestibular system in humans that measures the tilt of the body and activates contralateral muscle to keep balance. In this sense, we use the Gyros located in the chest of the robot to calculate the lateral tilt of the body. When this tilt is increasing we want the robot to tilt in the opposite lateral direction. There are 2 DOFs controlling the lateral direction in the robot, one DOF in the hip called LEG\_JOINT[2] (Hip1) and one in the ankle called LEG\_JOINT[6] (Ankle2) (Figure 5.26). Consequently we will introduce the feedback pathways in the CPGs controlling these joints.

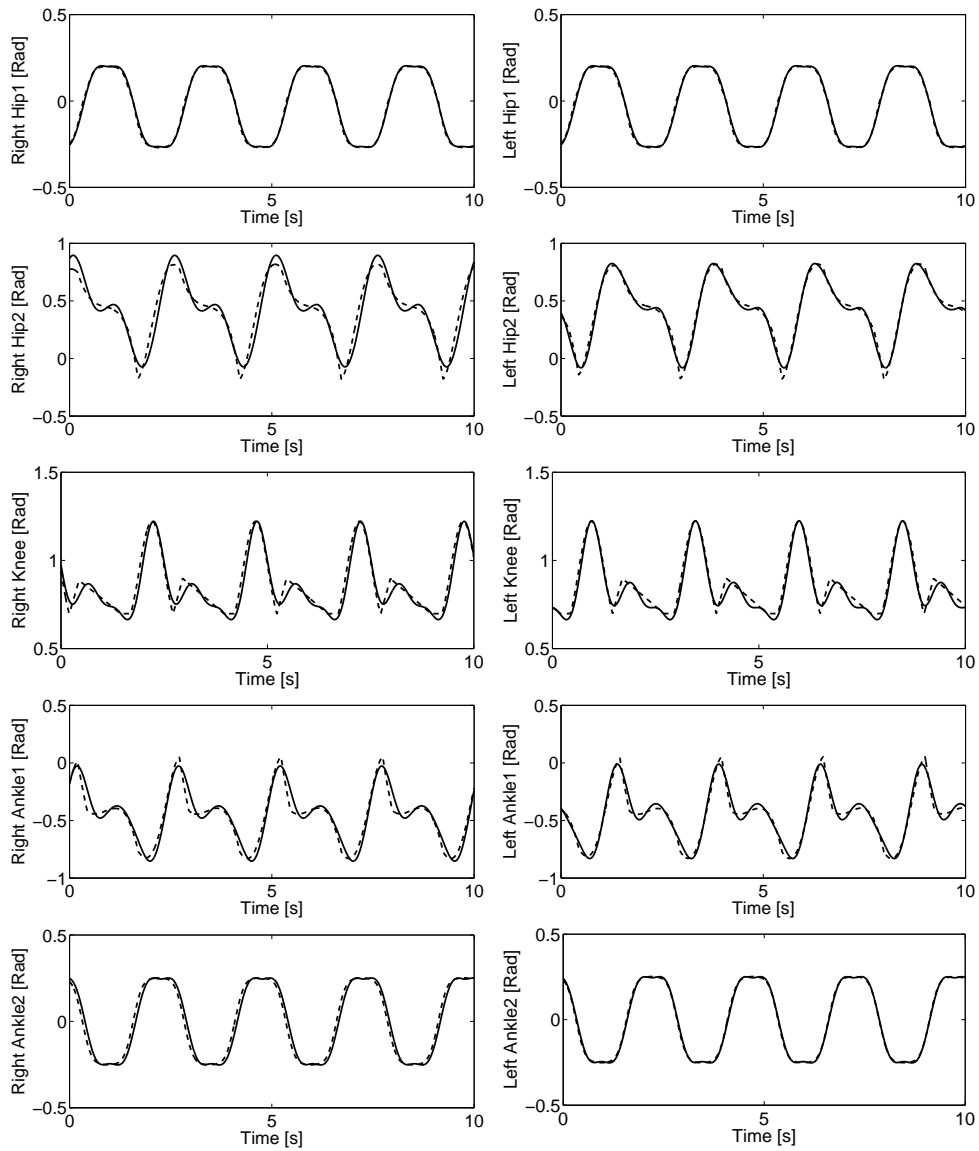


Figure 5.28: Result of training of the generic CPG. We plotted the 10 controlled DOFs, the plain line corresponds to the output of the CPG for each DOF, the dashed line corresponds to the sample trajectory.

## 5.4. Programmable central pattern generators

---

We notice from Figure 5.28 that the Hip1 joints have the same trajectories on both legs. We notice the same for the Ankle2 joints. The effect of feedback we want should be of opposite effect on the Ankles and on the Hip to keep the ankle parallel to the ground. The feedback pathways should also influence in the same way both legs.

Let  $\psi_{\text{lateral}}$  be the lateral tilt of the body, then we set the feedback for the ankles and the hips as

$$g_{\text{Ankle2}} = K_{\text{lateral}}|\psi_{\text{lateral}}| \quad (5.143)$$

$$g_{\text{Hip1}} = -K_{\text{lateral}}|\psi_{\text{lateral}}| \quad (5.144)$$

the gain  $K_{\text{lateral}}$  is the same for both feedback pathways because we want to assure that we have a symmetric change of trajectory, so when the ankle touches the ground, correct orientation is preserved.

We project these feedbacks on the radius of the limit cycle of all the oscillators associated to the Hip1 and Ankle2 joints. We make this projection to be sure that the phase is preserved, because we are only interested in amplitude of trajectories. The following Equation shows the principle

$$\dot{x}_i = (\mu - r^2)x_i - \omega_i y_i + \tau \sin(\theta_i - \phi_i) + g_k \frac{x_i}{r_i} \quad (5.145)$$

$$\dot{y}_i = (\mu - r^2)y_i + \omega_i x_i + g_k \frac{y_i}{r_i} \quad (5.146)$$

where  $x_i, y_i$  are the state variables of the  $i$ th oscillator,  $g_k$  is the feedback term ( $g_{\text{Ankle2}}$  or  $g_{\text{Hip1}}$ ).

**Pendulum effects compensation** When walking the body of the robot has the dynamics of an inverted pendulum. When modulating the speed of walking, we will change these effects and the controller has to compensate for these effects. Therefore we introduce feedback to compensate tilt of the body in the sagittal plane in the same way we did above. Let  $\psi_{\text{Pendulum}}$  be the angle of tilt of the body in the direction of walking, then we set the following feedback term

$$g_{\text{Pendulum}} = K_{\text{Pendulum}}\psi_{\text{Pendulum}} \quad (5.147)$$

we project this feedback term on the radius of all the oscillators of the CPGs associated to the Ankle1 joints (LEG\_JOINT[5]) and the Knee joints (LEG\_JOINT[4]).

**Phase resetting** The effect of pendulum will also influence the frequency at which the legs touch the ground, which will be slightly different than the frequency of the trajectory generation in the controller.

To compensate this effect, we introduce phase resetting of the oscillators each time the right leg touches the ground. Importance of phase resetting for biped locomotion was

already discussed in [95] where they showed that it creates entrainment of the controller with the body dynamics of the robot. This induce tight coupling between the body and the controller.

#### 5.4.4 Experimental results

In this section we present experiments we did with the CPG we presented. We did these experiment with a simulation of the Hoap-2 robot in Webots [90]. This simulator is based on ODE [2], an open source physics engine for simulating 3D rigid body dynamics. The model of the robot is as close to the real robot as the simulation enables us to do. It means we simulate the exact number of DOFs, the same mass distribution and inertia matrix for each limb, the same sensors (gyroscope and accelerometer in the chest, load sensors on the bottom of the feet).

The architecture of CPG we presented generates online trajectories for each joint. We use these trajectories as desired angles for the PID controllers controlling each joint.

When increasing the stepping frequency and therefore the speed of locomotion, the CPG has to react faster to sensory feedback. By changing the gains of the feedback pathways, we can change this speed of response, thus we define gains for the feedback that depends on the speed of locomotion

$$K_{\text{Lateral}} = 2000.0 + 200.0 * (\zeta - 1.0) \quad (5.148)$$

$$K_{\text{Pendulum}} = 1000.0(\zeta - 1.0) \quad (5.149)$$

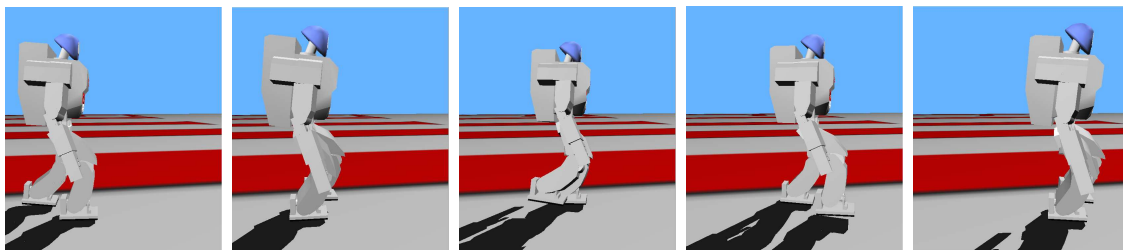
where  $\zeta$  is the ratio between the frequency to which we modulate the controller and the original frequency of the learned trajectory. In these equations we see that the gains increase as the speed of walking increases. We tested the CPG model with the simulated Hoap-2 robot. First of all we modulated the speed of walking by changing  $\zeta$ . We managed to increase the speed of the robot up to 50% of the original speed by simply setting  $\zeta = 1.5$ . This correspond to a speed of approximately  $0.12 \text{ m.s}^{-1}$ . We also managed to generate backward locomotion by simply inverting the sign of  $\vec{\omega}$ . Pictures of the robot walking at  $0.12 \text{ m.s}^{-1}$  and walking backward can be seen on Figure 5.29.

Moreover, by linearly changing  $\vec{\alpha}$  we managed to control the step length. It was possible to control the robot so that it made smaller steps and eventually stops if  $\vec{\alpha} = 0$ , it was possible to walk again by increasing  $\vec{\alpha}$ .

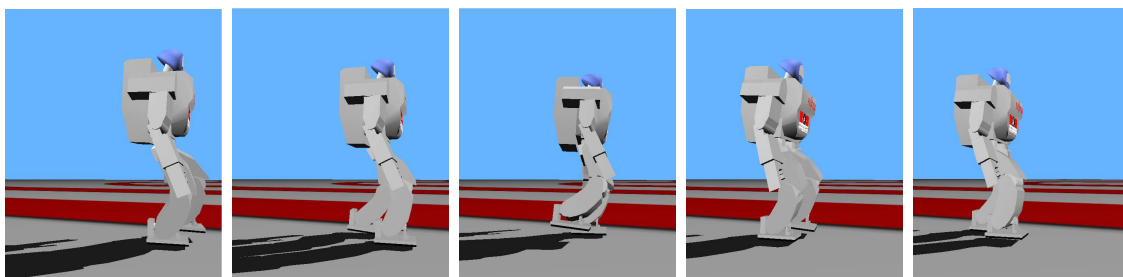
The feedback pathways enabled us to increase the speed of locomotion. Indeed if we do not activate the pathways, the robot falls when we increase speed of locomotion more than 5%. The contribution of the feedback, when increasing speed of locomotion can be seen in Figure 5.30. We plotted the trajectories generated without and with feedback when the robot walks 20% faster ( $\zeta = 1.2$ ). It is obvious on these graphs that the lateral feedback modifies quite a lot the trajectories of the Hip1 and Ankle2 joints. The importance of feedback on the other joints is less obvious but the experiments showed that without this pathway, the robot falls.

## 5.4. Programmable central pattern generators

---



(a) The robot is walking at  $0.12 \text{ m.s}^{-1}$



(b) The robot is walking backward

Figure 5.29: Snapshots of the robot while walking at higher speed (Figure (a)) and while walking backward (Figure (b)). The pictures have to be seen from left to right.

### 5.4.5 Conclusion

In this section we presented a new architecture for building programmable Central Pattern Generators used for online trajectory generation in autonomous robots. The interest of the method we presented is that we can encode arbitrary periodic trajectories as limit cycles in a network of coupled oscillators. Then we get all the properties of such systems, we can modulate the frequency and the amplitude in a smooth way, we have stability to perturbations and we can integrate feedback pathways. Moreover we showed that it was easy to couple several of such networks to generate coordinated multidimensional periodic trajectories. Furthermore, this new architecture is general enough to be applied to various fields where the control of periodic signals is important as, for example, in signal processing.

Afterwards, we showed an application of this programmable Central Pattern Generator to control a simulated Hoap-2 humanoid robot for biped locomotion<sup>3</sup>. We introduced simple feedback pathways and showed how from a sample trajectory we could build a controller able to modulate the speed of locomotion and the step length of the robot. These modulation are simple to do since we only have to vary the value of two parameters ( $\omega$  and  $\alpha$ ).

---

<sup>3</sup>This controller was also successfully applied on the real robot.

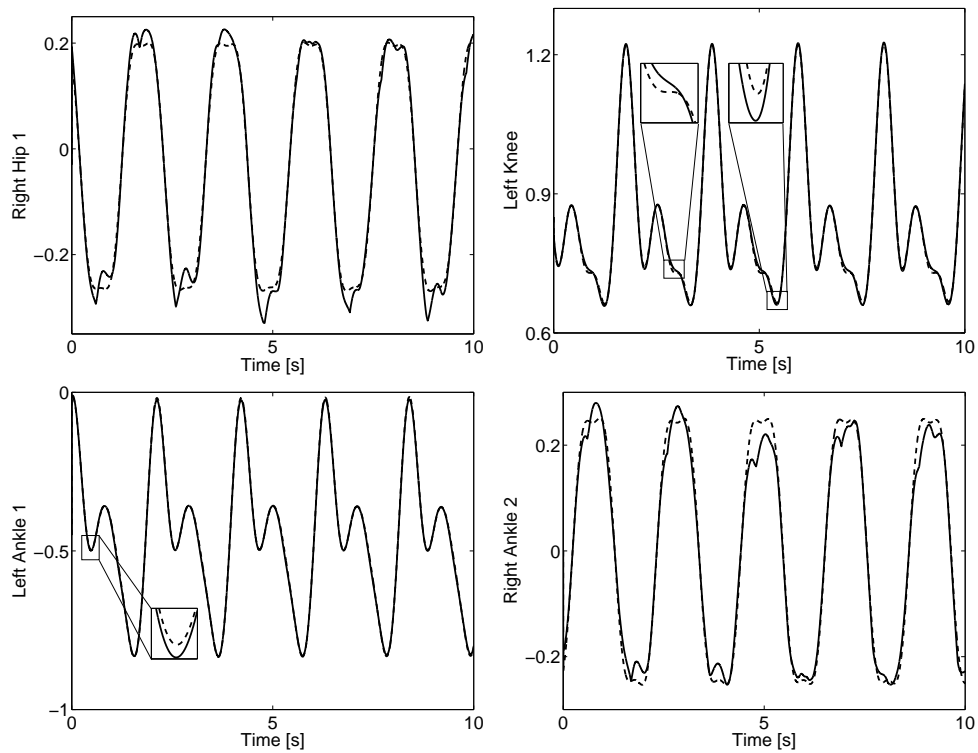


Figure 5.30: Effect of feedback on the generation of trajectory. The dotted line shows the trajectory initially encoded in the CPG, the plain line shows the trajectory generated by the CPG with the feedback pathways. The graphs are taken from trajectories at speed  $\zeta = 1.20$



### 5.5 Comments on programmable CPGs

What are the benefits of using programmable CPGs to embed periodic trajectories in limit cycles? As we have seen in the previous section, the main advantages of this system is that learning is embedded in the dynamics of the system and thus the CPG can easily learn patterns that change with time. Moreover, the pattern is encoded in a limit cycle and thus the system is stable against perturbations which is important if one wants to include sensory feedback in the CPG. It is also easy to smoothly modulate the pattern in frequency and amplitude.

However using many oscillators to encode each frequency component of the pattern to encode can lead to solutions with many oscillators (i.e. a high dimensional limit cycle) and moreover the inclusion of feedback on so many oscillators is not trivial. Should all these oscillators receive the same type of feedback? Then we can wonder what is the utility to use several oscillators instead of a simple oscillatory system (i.e. a low dimensional system) with a nonlinear filter that shapes the output of the oscillator to the desired pattern (see for example the work of [71, 72]). Otherwise should we use different feedbacks according to the type of harmonics that are encoded? And then it is far from clear how to design such a feedback controller that has a meaningful utility because the representation of the periodic pattern in frequency space does not give an intuitive framework for feedback design. Moreover it is not clear what would be the gain of such a system.

Among other methods to embed periodic trajectories in limit cycles one is of particular interest [71, 72]. The method is very elegant since it uses a simple harmonic oscillator combined with a nonlinear filter to learn the shape of the periodic signal to encode. The learning of the nonlinear filter is done using an incremental locally weighted regression technique from [123]. However one drawback from this method is that one needs to know the frequency of the periodic pattern to encode, which makes this method difficult to implement for online imitation learning where the pattern to encode might vary with time (which is a situation where programmable CPGs are very well suited).

Recently [48, 47] proposed an hybrid method combining the elegant aspects of both methods. It uses a network of adaptive frequency oscillators to learn the basic frequencies of the input signal and then uses the nonlinear filter to learn the exact shape of the periodic pattern. The results presented seem very promising since it suppresses the major drawbacks of the other approaches used separately. I think that it is through hybrid methods like this one that programmable CPGs can be very useful to encode movement primitives into limit cycle systems.



---

# CHAPTER 6

## CONCLUSION

---

Before closing this thesis I would like to discuss the different aspects addressed in this work. I will not go into much details since it was already done for each chapter but I will restate the original contributions achieved in this work together with a general discussion on the work accomplished.

### 6.1 Original contributions

**Crawling of human infants** (Chapter 3) Our study of crawling human infants showed that despite infants crawl for a very short period in their life and that they have a very different morphologies compared to other quadruped mammals, they share many gait similarities with them. Indeed, they perform a lateral sequence walk, which is common to all mammals but primates that do a diagonal sequence walk. Their basic limb kinematics is also similar to other mammals (one period flexion/extension of the most proximal joints, undulation of the spine in the horizontal plane). Finally there is a great variability in their stance duration which actually control locomotion speed, while swing duration is almost constant. This observation actually motivated the design of a locomotion specific oscillator for CPG based controllers.

**Locomotion specific oscillator** (Chapter 4) A simple oscillator in which we can independently control the duration of the ascending and descending phases of the oscillation was constructed. It was used to control the stance and swing durations of the limbs of the quadruped robots we controlled. This oscillator is well adapted to locomotion control and is sufficiently simple such that it is easy to understand and to control its dynamics. Experiments on simulated robots showed that separating the two locomotion phases was useful for locomotion control (stance duration can be used as a control parameter for locomotion speed and swing duration is fixed at a value that stabilizes the locomotion). Such an oscillator might be better suited for designs of CPG than other oscillators that are often used, such as the Matsuoka oscillator [76]. Indeed it has fewer state variables

and fewer parameters. Moreover its parameters directly set important quantities such as amplitude and frequencies (swing/stance).

**Methodology for CPG design** (Chapter 4) We tried to provide a methodology to design CPGs. Such a method was clearly missing in the field and it is hoped that readers will try it for their own robots. We showed that using theories from dynamical systems we could separate the design problem in three distinct steps. First the choice of an oscillator, then the design of the coupling architecture of the network and finally the inclusion of sensory feedback. We designed sensory feedback at the oscillators level to modulate the transitions between swing and stance phases. Interestingly this simple local feedback pathway was sufficient to improve greatly the locomotion of the three simulated robots we used. The most impressive results were with the GD robot which has some passive dynamics due to compliant joints and we hope that we will have more robots like this in the future.

**Adaptive frequency oscillators** (Chapter 5) We presented a mechanism for frequency adaptation in oscillators. It is an useful tool for CPG design and for dynamical systems in general since it creates infinite basins of attraction for synchronization (as opposed to synchronization with conventional oscillators). Moreover it has a memory of past interactions. This mechanism is generic enough to be applied to a large class of oscillators, from harmonic ones to relaxation types and even strange attractors. The mathematical analysis we performed helped to understand many fundamental aspects of this mechanism: its exponential convergence, the control of the relaxation time, the fundamental limits of resolution in time-frequency space that relate to Heisenberg boxes and the basins of attraction for multi frequency inputs. This mechanism was applied to adaptive control of locomotion of a legged robot with passive dynamics (work of J. Buchli [18]) and to the development of pools of oscillators as discussed next.

**Pool of oscillators** (Chapter 5) We built pools of oscillators with a negative feedback loop to recover the frequency spectrum of any periodic input. This can be viewed as a dynamical systems approach to Fourier series decomposition. To the best of our knowledge this is a very novel concept. In these pools there is not only frequency adaptation but also adaptation of the amplitude associated to each frequency component of the signal to decompose. They work particularly well for time-varying spectra. An application of this idea was the development of programmable CPGs in which we can encode dynamically any periodic pattern as a stable limit cycle. We also presented an application of these CPGs for biped locomotion control.

## 6.2 Final conclusion

After four years of research, the control of locomotion is still an open and difficult problem. In this thesis we took a bio-inspired approach for the control of legged locomotion and we used the dynamical systems framework to formulate our design.

We have proposed a systematic way to design central pattern generators and showed that it could successfully be applied to different robots. The controller we use is very simple and does not address all the aspect of locomotion control yet (e.g. explicit control of stability) but the results we obtained are promising and the chosen framework makes it compatible with other approaches discussed in Chapter 2 and therefore opens an interesting path of future research.

On the other hand we showed that by using the dynamical systems framework we could design learning systems where there is no separation between the learning algorithm and the learning substrate but where the dynamics of both systems are interconnected. The frequency adaptation mechanism we developed seems quite fundamental since it works for a large class of oscillators. It provides also an interesting approach to frequency analysis. This mechanism opens a wide range of future researches for applications in signal processing, adaptive control of robots and biological modeling.



---

## LIST OF FIGURES

---

2.1	Pictures of the iCub at various stages of its development. . . . .	8
3.1	Configuration of the experimental setup, with the 5 cameras and the crawling scene (top view). . . . .	21
3.2	This figure shows an example of preprocessing of the data, the left graph is raw data before processing and the right one shows the data after processing. The data corresponds to the position of the left hand in 3D (each line is a dimension). . . . .	21
3.3	This figure shows an example of the velocity profile during the swing and stance phases of the left arm of an infant (lower graph) and the corresponding time series of 3D positions (upper graph). The vertical lines show the separation between the swing (in gray) and stance (in white) phases. . .	23
3.4	Hildebrand diagram for the seven infants. . . . .	24
3.5	Typical footfall sequence of the infant standard crawling gait. The gray boxes show the stance phases and the white ones the swing phases. . . .	24
3.6	Median duration of the swing (top figure) and stance phases (bottom figure) with the interquartile range as error bars. Each group of data represents an infant, the four bars represent the left arm, the right arm, the left and right legs respectively. . . . .	26
3.7	Cycle frequency, 1/stance duration and 1/swing duration as a function of the speed of locomotion for 7 infants. The line shows the linear relation obtained by linear regression on the data. The legend is as follow: ● for A. ○ for Al. △ for E. ▽ for J. ■ for M. □ for O. and × for V. . . . .	26
3.8	Schematic of the joint angles we measured. We look at the movement of the limbs in the sagittal plane and the spine in the horizontal plane . . .	27
3.9	Stick figures from real data of a crawling sequence. The left graph shows a lateral view of the infant, the dashed line representing the left limbs. The right graph shows the same sequence from a top view, so the movement of the spine in the horizontal plane is visible. The black dots represent the position of the markers on the infant. There is a 100 ms duration between each figure. . . . .	28

3.10	Median fore limbs kinematics for the seven different infants. We show the shoulders and elbow angles. For each infant, we rescaled the trajectory into normalized swing and stance phases. The vertical dashed line indicates touch down of the hand. . . . .	29
3.11	Median hind limbs kinematics for the seven different infants. We show the hips and knees angles. For each infant, we rescaled in time the trajectory into normalized swing and stance phases. The vertical dashed line indicates touch down of the knee. . . . .	31
3.12	Median movement of the spine for each of the seven infants. We show 2 figures, for comparison with the movement of the limbs of the infants. For the left figure, the first vertical bar indicates the end of the swing of the left arm, the second one the start of swing of the right arm and the last bar shows the end of the swing of the right arm. For the right figure, the first vertical bar indicates the end of the swing of the right leg, the second one the start of swing of the left leg and the last bar shows the end of the swing of the left leg. On both figures the arrows denote the maximum curvature of the spine. . . . .	32
3.13	This figure shows the changes in swing duration (top figure) and stance duration (bottom figure) for the three infants that we measured twice. For each infant, the left set of bars corresponds to the first experiment, the right one corresponds to the second one. The data is represented as in Figure 3.6. . . . .	33
3.14	This figure shows the relation between the median speed of locomotion and the number of days since the infant started crawling. □ and ■ represent infant A. for experiment 1 and 2 respectively. △ and ▽ are for Al., ○ and ● for E., × for J., ☆ for M., ◁ for O. and + for V. . . . .	35
4.1	Schematic of the joint angle we measured. We look at the movement of the limbs in the sagittal plane. . . . .	42
4.2	Evolution of the joint angles of an infant during crawling. This is a reconstruction of a crawling sequence from the recordings of a crawling infant. We plot the joint angles (in radian) of the 4 limbs. For each limb, we plot the joint angles as defined in Figure 4.1. Hip and shoulder joints are plotted in plain line, the knee and elbow joints are in dashed line. The vertical lines delimit the swing and stance phases, the swing phase being the shortest one. . . . .	43



## List of Figures

---

4.3	This figure shows how we can independently control the ascending and descending durations of the oscillator, in 4.3(a) we plot the oscillations when $k_{stance} = k_{swing} = 4.(2\pi)^2$ , in 4.3(b) we plot $k_{stance} = \frac{1}{3}k_{swing} = 4.(2\pi)^2$ . In each plot we show the oscillations $x$ and the velocity $y$ . At time $t = 1.5$ , we perturb the system by setting $x$ and $y$ to a random value, we clearly see that the oscillations are stable. . . . .	47
4.4	The architecture of the CPG . . . . .	48
4.5	From the symmetry of the network, we derived the possible pattern of synchronization according to the possible subgroups of spatial symmetry. For each subgroup, we indicate the possible periodic solutions and their stability. The stability of the solutions was evaluated numerically, as shown in Figure 4.6. . . . .	48
4.6	We show the 4 possible patterns of synchrony we predicted from the symmetries of the network. We also show their stability properties by perturbing the oscillators. For patterns of Figs. 4.6(a) and 4.6(b) at time $t = 10s$ we add a perturbation of 0.01 to $x_1$ , we see that such a small perturbation completely destroys the patterns and the crawling pattern appears. For the pattern of Figure 4.6(c), which is the crawling pattern, at time $t = 10$ we add 1.0 to $x_1$ and at time $t = 20$ we set the state variables of each oscillator at a random value between $[-2, 2]$ , it is clear that this pattern is stable. The pattern in Figure 4.6(d) is a pace gait, at time $t = 10s$ we add a random noise between $[-0.2, 0.2]$ on each $x_i$ . For all the experiment, we set $k_{swing} = k_{stance} = \pi^2$ , $c1 = c2 = 1.0$ , $\beta = 100$ , $\sigma = 10$ , $\nu = 0.45$ , $b = k = 100$ and $\mu = 1$ . . . . .	49
4.7	In Figure 4.7(a) we modulate the frequency of the CPG. Initially we have $k_{swing} = k_{stance} = \pi^2$ , at $t = 10$ we set $k_{swing} = 4k_{stance} = 4\pi^2$ , which corresponds to a doubling of the speed of the swing and at $t = 20$ we set $k_{stance} = 4k_{swing} = 4\pi^2$ . In Figure 4.7(b) we modulate the amplitude of the pattern, we set $\mu = 1$ at $t = 0$ , then $\mu = 0.5$ at $t = 10$ and $\mu = 1.5$ at $t = 20$ . Note that an abrupt change in the control parameters ( $k_i, \mu$ ) leads to a smooth transition in the generated pattern. . . . .	52
4.8	Comparison of the real trajectories of the hip and shoulder joints and the trajectories generated by the CPG. The trajectories of the CPG are only shifted to oscillate around the same mean values as the real trajectories. We see that the trajectories generated by our model fit quite well the real ones, especially for the right limbs. . . . .	53
4.9	Characterization of the speed of the simulated iCub according to the duration of the stance and swing phases. The left color bar shows the correspondence between the colors and the speed (in $m \cdot s^{-1}$ ). . . . .	54
4.10	We show a sequence of crawling of both a real infant and the simulated robot. There is an interval of 120ms between each picture. . . . .	55

4.11	Example of behavior of the oscillator of Equations (4.15)-(4.17). The $x$ variable is shown. At $t = 0$ we have $\omega_{swing} = 0.5\pi$ , $\omega_{stance} = 8\pi$ and $\mu = 1$ . At $t = 10$ , we change $\omega_{swing} = \omega_{stance} = 0.5\pi$ . At $t = 20$ , we change $\mu = 0.5$ and at $t = 30$ , we set $\omega_{swing} = 2\pi$ , $\omega_{stance} = 0.4\pi$ and $\mu = 1$ . . . .	60
4.12	Network generating the trot, pace and bound gaits that was used in [114]. The same type of arrows denotes the same coupling function. The new networks, where new symmetries are added are shown in Figure 4.13. . .	61
4.13	Generic networks corresponding to four different gaits. The same type of arrows denote the same type of coupling functions. . . . .	63
4.14	Example of a gait transition. At time $t = 2$ we switch the coupling of the network from the walk matrix to the pace matrix. We also change $\omega_{stance}$ from $2\pi$ to $4\pi$ . $\omega_{swing} = 8\pi$ for all $t$ . . . . .	65
4.15	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. . . . .	66
4.16	These two figures show the existence of two generic solutions of the trot network and their instability, since at $t = 3$ we perturb the oscillators with a small perturbation (1% of the amplitude of the oscillator). The upper graph shows the behavior of the $x$ variable of each cell in the network. The lower graph shows the phase differences between the first cell and the second ( $\times$ ), the third ( $\circ$ ) and the fourth ( $\Delta$ ) cell. The initial conditions for the left graph are $(x(t), x(t), x(t), x(t))$ and for the right graph $(x(t), x(t + \frac{1}{4}), x(t + \frac{3}{4}), x(t + \frac{1}{2}))$ . . . . .	67
4.17	Phase space of an oscillator (left figure) 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 is shown on the right.	69
4.18	Schematic oscillator phase plot for the different types of feedback. . . . .	70
4.19	Schematic of a quadruped robot where the abstract limbs are represented by a segment defined by a length $h_i$ and an angle $\alpha_i$ relative to the body. The dashed segments represent a possible representation of the real limbs corresponding to this abstraction, with the detailed kinematic chains. . .	70
4.20	We show an example of the evolution of $z$ , defined by Equation (4.27), together with the evolution of the $x$ variable of the corresponding oscillator. At $t = 4.3$ , we simulate a fast transition from swing to stance (vertical dashed line). Here the parameters used are $K_1 = 30$ , $K_2 = 100$ , $\omega_{swing} = 2\pi$ and $\omega_{stance} = \pi$ . . . . .	71
4.21	The three different robots used in the experiments. . . . .	72
4.22	We show the locomotion speed of the Aibo for different values of $\omega_{stance}$ and $\omega_{swing}$ . The values of the parameters are shown in Hz (in Equation (4.17) they are expressed in $\text{rad} \cdot \text{s}^{-1}$ ). The locomotion speed is expressed in $\text{body length} \cdot \text{s}^{-1}$ . . . . .	74

## List of Figures

---

4.23	We show the locomotion speed of the iCub for different values of $\omega_{stance}$ and $\omega_{swing}$ . The values of the parameters are shown in Hz (in Equation (4.17) they are expressed in $\text{rad} \cdot \text{s}^{-1}$ ). The locomotion speed is expressed in $\text{body length} \cdot \text{s}^{-1}$ . . . . .	75
4.24	We show the locomotion speed of the Ghostdog for different values of $\omega_{stance}$ and $\omega_{swing}$ . The values of the parameters are shown in Hz (in Equation (4.17) they are expressed in $\text{rad} \cdot \text{s}^{-1}$ ). The locomotion speed is expressed in $\text{body length} \cdot \text{s}^{-1}$ . White squares show the experiments where the robot falls. . . . .	76
4.25	In the upper figure we show a typical swing phase of the iCub robot on a slope when there is not feedback, we notice that after touch down, the limb continues to move forward. The lower figures show the relation between the CPG outputs and the force sensors in the case of a CPG with feedback (right graph) and one without (left graph). . . . .	79
4.26	Trajectories of the GD robot on different slopes with feedback (lower graphs) and without feedback (upper graphs). We made tests for configurations corresponding to parameters in the range $2.8\pi < \omega_{swing} < 3.6\pi$ and $2\pi < \omega_{stance} < 5.2\pi$ which corresponds to the best parameter sets for both controllers with and without feedback. The black circles show the initial position of the robot, the square shows the slope on the terrain and the arrow shows the climbing direction. . . . .	80
4.27	Example of an automatic gait transition for the Ghostdog robot with a CPG with feedback. The CPG is coupled to generate a walk. At $t = 25$ we change $\omega_{stance}$ from $0.8\pi$ to $2.4\pi$ . During the whole experiment $\omega_{swing} = 2.4\pi$ . . . . .	81
5.1	We illustrate the coordinate system in which synchronization is most naturally discussed. The figure shows an arbitrary limit cycle. The system is strongly damped in direction perpendicular to the limit cycle $\vec{e}_r$ and marginally stable in direction tangential to the limit cycle $\vec{e}_\phi$ . This is the reason for the structurally stable limit cycle in the first place and allows for a resetting of the phase on the other hand. Note that the 2-dimensional representation is always valid for discussing a limit cycle since there exists always a 2 dimensional manifold which contains the limit cycle. Refer to text for a discussion of the perturbation $\vec{P}$ . . . . .	90
5.2	Plot of the evolution of $\omega$ for four different values of $\epsilon$ . Here we set $\mu = 1$ , $x(0) = 1$ and $y(0) = 0$ , the perturbing force is $F = \cos(30t)$ . For every value of $\epsilon$ , we see that $\omega$ converges to 30, which is the frequency of the input signal. Therefore, the system is able to learn the frequency of the input signal. We also notice that $\epsilon$ controls the convergence rate, the higher it is, the faster the system learns. . . . .	96

5.3 Results of the simulation of the first and second order approximations. For a simple input, here  $F = \sin(40t)$ ,  $\epsilon = 0.9$ , initial conditions are  $t_0 = 0$ ,  $w_0 = 30$ . The upper figure shows the evolution of the  $\omega$  variable for the initial dynamical system (Eq. (5.38)), the first order approximation  $\omega_\epsilon(t)$  and the 2nd order approximation  $\omega_{\epsilon^2}(t)$ . The lower figure shows quadratic errors between the initial system and the 2 approximations, for the evolution of  $\omega$ . . . . . 97

5.4 In this figure, we plotted  $\omega(t)$  for several initial conditions,  $\omega_0$ . The periodic input is Equation (5.43),  $\epsilon = 0.9$ . The dotted lines indicates the boundary between the different basins of attraction, corresponding to the different frequency components of the input, that were predicted analytically. . . . 100

5.5 The left plot of this figure represents the evolution of  $\omega(t)$  when the adaptive Hopf oscillator is coupled to the z variable of the Lorenz attractor. The right plot represents the z variable of the Lorenz attractor. We clearly see that the adaptive Hopf oscillators can correctly learn the pseudo-frequency of the Lorenz attractor. See the text for more details. . . . . 101

5.6 Frequency spectrum of the Van der Pol oscillator, both plot with  $\omega = 10$ . The left figure is an oscillator with  $\alpha = 10$  and on the right the nonlinearity is higher  $\alpha = 50$ . On the y-axis we plotted the square root of the power intensity, in order to be able to see smaller frequency components. . . . . 103

5.7 Plot of the frequency of the oscillations of the Van der Pol oscillator according to  $\omega$ . Here  $\alpha = 50$ . There are 2 plots, in dotted line the oscillator is not coupled and in plain line the oscillator is coupled to  $F = \sin 30t$ . The strength of coupling is  $\epsilon = 2$ . We clearly see basins of phase-locking, the main one for frequency of oscillations 30. The other major basins appears each  $\frac{30}{n}$  (dotted horizontal lines). We also notice small entrainment basins for some frequencies of the form  $\frac{30p}{q}$ . For a more detailed discussion of these results refer to the text. . . . . 104

5.8 This figure shows the convergence of  $\omega$  for several initial frequencies. The Van der Pol oscillator is perturbed by  $F = \sin(30t)$ , with coupling  $\epsilon = 0.7$ ,  $\alpha = 50$ . We clearly see that the convergence directly depends on the initial conditions and as expected the different kinds of convergence correspond to the several entrainment basins of Figure 5.7. . . . . 105

## List of Figures

---

- 5.9 We show the adaptation of the Van der Pol oscillator to the frequency of various input signals: (a) a simple sinusoidal input ( $F = \sin(40t)$ ), (b) a sinusoidal input with uniformly distributed noise ( $F = \sin(40t) + \text{uniform noise in } [-0.5, 0.5]$ ), (c) a square input ( $F = \text{square}(40t)$ ) and (d) a sawtooth input ( $F = \text{sawtooth}(40t)$ ). For each experiment, we set  $\epsilon = 0.7$  and  $\alpha = 100$  and we show three plots. The right one shows the evolution of  $\omega(t)$ . The upper left graph is a plot of the oscillations,  $x$ , of the system, at the beginning of the learning. The lower graph shows the oscillations at the end of learning. In both graphs, we also plotted the input signal (dashed). In each experiment,  $\omega$  converges to  $\omega \simeq 49.4$ , which corresponds to oscillations with a frequency of  $40 \text{ rad} \cdot \text{s}^{-1}$  like the input and thus the oscillator correctly adapts its frequency to the frequency of the input. . . . . 107
- 5.10 We show results for several adaptive oscillators. For each oscillator, we give its equation in the right column,  $\omega$  corresponding to the adaptive parameter. We also specify the values of the different parameters used in the experiments. In the left column we plotted results of the experiment. Each figure is composed of 3 plots. The right one is a plot of the evolution of  $\omega$ . The left ones are plots of the oscillations (the  $x$  variable) and of the input signal  $F$  (dashed line), before (upper figure) and after (lower figure) adaptation. . . . . 109
- 5.11 (a) We plot  $\omega_d$  for the adaptive frequency phase oscillator (in blue), the approximated system up to the second order term of the Taylor series (in dashed red) and the exponential convergence (in black). We used  $\omega_F = 30$ ,  $\omega(0) = 50$  and  $K = 1000$ . We see the good match between the approximations and the original system. (b) This figure shows the final relative amplitude of oscillations of  $\omega$  after convergence of the original system as a function of  $K$ . In this experiment we used  $\omega_F = 30$ . . . . . 116
- 5.12 Left: relative standard deviation  $\sigma_{\Delta\omega}$  of the  $\omega$  variable around the converged frequency as a function of  $\tau\omega_F$ . The diagonal dashed line shows the linear approximation for these values for  $\tau\omega_F > 1$ . Right: amplitude of the oscillations of  $\omega$  after convergence. Note the log scale on the two graphs. In this experiment, we used  $K = 10^7$  and  $\tau = 0.1, 1$  and  $10$  for the red, blue and green lines respectively. . . . . 118
- 5.13 Example of convergence of  $\omega$  for small coupling ( $K = 20$ ). The input signal is  $F(t) = \cos(60t)$ ,  $\omega(0) = 90$ . The vertical dashed line shows the limit of the entrainment basin, we notice that convergence becomes exponential after the frequency of the oscillator enters in it. . . . . 120

5.14 These figures shows the entrainment basins of a phase oscillator (in dashed line) for two different inputs, the vertical dotted lines represent the frequency components of the forcing signal. The light gray area represents the region where there is exponential convergence of the frequency adaptation. The thick black lines separate the region of convergence (i.e. towards which frequency component the oscillator goes). The left graph shows result for a periodic signal  $F(t) = 1.3 \cos(30t + 0.4) + \cos(60t) + 1.4 \cos(90t + 1.3)$ , the right graph shows results for a non periodic signal  $F(t) = 1.3 \cos(30t) + \cos(30\sqrt{2}) + 1.4 \cos(\frac{30\pi}{\sqrt{2}})$ . See the text for discussion of the results and an explanation of the dark gray zone of the right graph. 122

5.15 Relative mean error of convergence  $\frac{\langle \omega \rangle}{\omega_F}$  of the adaptive frequency Hopf oscillator (plain line). We also show the predictions made with the linearization of  $r$  (dash-dotted line) and the frequency response of  $r$  numerically evaluated (dashed line). . . . . 125

5.16 Structure of the pool of adaptive frequency oscillators that is able to reproduce a given signal  $T(t)$ . The mean field produced by the oscillators is fed back negatively on the oscillators (taken from [19]). . . . . 127

5.17 Frequency response of the pool of oscillators, the magnitude of the response is on the left figure, the phase delay on the right one (in this case  $N = 1$  but results are the same for higher values of  $N$ ).  $\tau = 1$  is represented by the green line,  $\tau = 0.1$  by the red line and  $\tau = 10$  by the blue line. The magnitudes 0dB and  $-3$ dB are represented by the two horizontal lines on the left figure. See the text for more details. . . . . 129

5.18 Comparative convergence behavior for a system with a single oscillator, without feedback loop ((a) and (b)) and with feedback loop ((c) and (d)). We plot the frequency differences  $\omega_d = \omega - \omega_F$  normalized by  $\omega_F$ . For each graph we show the behavior for different values of  $\tau$  (red for  $\tau = 0.01$ , blue for 0.1 and green for 1). For each experiments, we used  $\omega_D(t = 0) = 0.5$  and  $K = 10^5$ . See the text for the discussion on the results. . . . . 130

5.19 These graphs shows the convergence behavior of a pool of 50 oscillators to the frequencies of the input signal  $T(t) = 0.2 \sin(200t) + 0.4 \sin(100\pi t) + 0.4 \sin(450t)$  for two different values of  $\tau$ . In both cases we used the same initial conditions and  $K = 200$ . . . . . 131

5.20 Frequency response of the amplitude adaptation  $\alpha_i$ . The left graph shows the magnitude and the right graph the delay of the response. Red line is for  $\eta = 0.2$ , the blue one for  $\eta = 2$  and the green one for  $\eta = 20$ . The vertical lines of the left graph show the magnitudes 0 and  $20 \log_{10} \frac{\sqrt{2}}{2}$ . We used  $K = 100$ ,  $\tau = 1$  and  $\omega_F = 1000$ . . . . . 133

## List of Figures

---

- 5.21 Examples of decomposition of the spectrum of an input signal  $T(t) = 1.3 \cos(30t) + \cos(30\sqrt{2}t) + 1.4 \cos(\frac{30\pi}{\sqrt{2}}t)$  with a pool of  $N = 3$  oscillators for two different coupling strengths. The parameters used in the simulations are  $\tau = 0.5$  and  $\eta = 2$ ,  $K = 10$  (top) and  $K = 100$  (bottom). Refer to the text for more details. . . . . 135
- 5.22 These graphs shows the result of analysis of a signal with a time varying spectra with a pool of adaptive frequency oscillators, using the amplitude adaptation. The input signal is composed of one ascending linear chirp  $\sin(200t + 2t^2)$ , one descending quadratic chirp  $\sin(400t - \frac{t^3}{15})$ , and two frequency modulated gaussians located at  $t = 5$  and  $30$ :  $\sin(300t) \exp^{-\frac{(t-5)^2}{2.5}}$  and  $\sin(400t) \exp^{-\frac{(t-30)^2}{5}}$ . The pool is composed of  $N = 100$  oscillators,  $\tau = 0.05$  and  $\eta = 0.2$ . The upper figure shows the frequency distribution of the oscillators weighted by their respective amplitude as a function of time. The lower left graph shows the evolution of the  $\omega_i$  and  $\alpha_i$  variables and the lower right graph shows the input signal  $T(t)$  and the difference between the output of the pool and the input. The vertical dashed bars signals the important event in time: the maximum of the 2 gaussians and the crossing of the chirps. . . . . 137
- 5.23 Structure of the network of adaptive Hopf oscillators. Each oscillator receives the same learning signal  $F(t) = P_{teach}(t) - \sum_i \alpha_i x_i$ , which is the difference between the signal to be learned,  $P_{teach}(t)$ , and the signal already learned,  $Q_{learned}(t) = \sum_i \alpha_i x_i$ . Then all the oscillators (except oscillator 0) receive the scaled phase input  $R_i$  from oscillator 0. Refer to Equations (5.133)-(5.137) and to the text for more details. . . . . 140
- 5.24 Figure 5.24(a) shows the input signal to learn,  $P_{teach}$ , in the upper graph and the result of learning  $Q_{learned}$  in the lower graph. It is obvious that the network correctly learned the input pattern. Figure 5.24(b) shows the evolution of the state variables of the generic CPG during learning of an input signal ( $P_{teach} = 0.8 \sin(15t) + \cos(30t) - 1.4 \sin(45t) - 0.5 \cos(60t)$ ) and the evolution of the error of learning. The upper graph is a plot of the error, defined by  $error = \|P_{teach} - Q_{learned}\|$ . The 3 other graphs show the evolution of the frequencies,  $\omega_i$ , the amplitudes,  $\alpha_i$  and the phases,  $\phi_i$ . The variables for each oscillator are plotted, variables of oscillator 0 are the plain lines, variables for oscillator 1 are the dotted-dashed lines, variables for oscillator 2 are the dotted lines and the dashed lines represent oscillator 3. The initial conditions are  $\alpha_i(0) = \phi_i(0) = 0$ ,  $x_i(0) = 1$ ,  $y_i(0) = 0 \forall i$ ,  $\mu = 1$ ,  $\gamma = 8$ ,  $\epsilon = 0.9$ ,  $\eta = 0.5$  and  $\tau = 2$ . The frequencies  $\omega_i(0)$  are uniformly distributed from 6 to 70. . . . . 143

5.25	Figure 5.25(a) presents the evolution of the output of the generic CPG when perturbed. At time $t_p = 1$ a perturbation occurs on all the oscillators of the CPG. We clearly see that the CPG quickly recovers its original behavior, thus proving the stability properties of the system. Figure 5.25(b) shows the behavior of the system when the amplitude $\vec{\alpha}$ is changed. At time $t = 2$ , the amplitude is divided by 2 and at time $t = 4.5$ the amplitude is multiplied by 3. Figure 5.25(c) shows the behavior of the network when the frequency $\vec{\omega}$ is changed. At time $t = 2$ the frequency is divided by 2 and at time $t = 5$ frequency is multiplied by 3. In both graphs, we can notice the smoothness of the trajectory when the parameters are changed.	144
5.26	Real Hoap-2 robot (Fig. (a)) and schematic of its DOFs (Fig. (b)), this pictures were taken from [1]. We can directly see which DOF the CPG of Figure 5.27 controls on the schematic. . . . .	145
5.27	Structure of the CPG for the humanoid. We use a generic CPG as presented in Section 5.4.2 for each DOF of the legs. We also add state variables that will learn the phase differences between the generic CPGs of the legs (the descending arrows). Antisymmetric coupling is also done between the 2 legs through the main oscillator of the first DOF of each leg (horizontal arrow). The trajectories generated for each DOF is the weighted sum of the corresponding 3 oscillators. . . . .	146
5.28	Result of training of the generic CPG. We plotted the 10 controlled DOFs, the plain line corresponds to the output of the CPG for each DOF, the dashed line corresponds to the sample trajectory. . . . .	148
5.29	Snapshots of the robot while walking at higher speed (Figure (a)) and while walking backward (Figure (b)). The pictures have to be seen from left to right. . . . .	151
5.30	Effect of feedback on the generation of trajectory. The dotted line shows the trajectory initially encoded in the CPG, the plain line shows the trajectory generated by the CPG with the feedback pathways. The graphs are taken from trajectories at speed $\zeta = 1.20$ . . . . .	152



---

# LIST OF TABLES

---

3.1	Date of birth, experience of crawling (i.e. the number of days since estimated start of crawling), number of complete steady crawling cycles that were extracted from the experiments for each limb, and body mass the day of the experiment for the seven infants (note that A., Al. and E. were seen two times) . . . . .	20
3.2	Differences in median values for the duration of the swing and stance phases of each limb for the 3 infants after the 2nd experiment (p-value and percentage of change in the median value, a negative percentage means that the value has decreased of that percentage) and differences in speed of locomotion. The bold numbers represent p-values $< 5\%$ and their associated variation in the median value. . . . .	34
4.1	These two tables show the different possible solutions corresponding to the trot and walk networks. For both networks we show the possible pattern of solution for the 4 cells together with the associated group of spatial (K) and spatiotemporal (H) symmetries. For example $(x(t), y(t), y(t + \frac{1}{2}), z(t))$ means that the solutions for cells 2 and 3 are the same up to a time shift of half a period, while the solutions of cells 1 and 4 are different. For the walk network $\mathbb{Z}_4$ is the group generated by $((1423))$ , while $\mathbb{Z}_2$ is generated by $((12)(34))$ . For the trot network $\mathbb{D}_4$ is the full group of symmetries of the gait, for the other groups we show the generators in parentheses, where $\tau = ((14)(2)(3))$ , $\sigma = ((1)(23)(4))$ , $\kappa = ((12)(34))$ , $\nu = ((13)(24))$ and $\varrho = ((1243))$ . . . . .	64
4.2	Correlation between speed of locomotion and $\omega_{swing}$ and $\omega_{stance}$ . We used a Spearman correlation test. The numbers in bold are correlations with a corresponding p-value $< 0.01$ . . . . .	74
5.1	This table summarizes the maximum errors of the simulation for the first and second order approximations discussed from Figure 5.3 . . . . .	98



---

## BIBLIOGRAPHY

---

- [1] Hoap-2 instruction manual. [Online]  
<http://www.automation.fujitsu.com/en/products/products09.html>.
- [2] Open dynamic engine documentation. [Online] <http://www.ode.org>.
- [3] J.A. Acebrón, L.L. Bonilla, C.J. Pérez Vicente, F. Ritort, and R. Spigler. The kuramoto model: A simple paradigm for synchronization phenomena. *Reviews of Modern Physics*, 77(1):137–185, 2005.
- [4] J.A. Acebron and R. Spigler. Adaptive frequency model for phase-frequency synchronization in large populations of globally coupled nonlinear oscillators. *Physical Review Letters*, 81:2229–2232, 1998.
- [5] K. E. Adolph, B. Vereijken, and M. A. Denny. Learning to crawl. *Child Development*, 69(5):1299–1312, October 1998.
- [6] R.M. Alexander and A.S. Jayes. a dynamic similarity hypothesis for the gaits of quadruped mammals. *Journal of Zoology*, 201:135–152, September 1983.
- [7] S. Aoi and K. Tsuchiya. Locomotion control of a biped robot using nonlinear oscillators. *Autonomous Robots*, 19:219–232, 2005.
- [8] S. Aoi and K. Tsuchiya. Adaptive behavior in turning of an oscillator-driven biped robot. *Autonomous Robots*, 23(1):37–57, 2007.
- [9] Y.I. Arshavsky, T.G. Deliagina, and G.N. Orlovsky. Pattern generation. *Current Opinion in Neurobiology*, 7(6):781–789, 1997.
- [10] M. A. Ashley-Ross. Hindlimb kinematics during terrestrial locomotion in a salamander (*dicamptodon tenebrosus*). *Journal of experimental biology*, 193:255–283, 1994.
- [11] R. Beira, M. Lopes, M. Praa, J. Santos-Victor, A. Bernardino, G. Metta, F. Becchi, and R. Saltarn. Design of the robot-cub (icub) head. In *IEEE International Conference on Robotics and Automation, ICRA2006*, Orlando, May 2006.

- 
- [12] Igor Belykh, Vladimir Belykh, and Martin Hasler. Generalized connection graph method for synchronization in asymmetrical networks. *Physica D*, 224(1-2):42–51, 2006.
- [13] R. Borisyuk, M. Denham, F. Hoppensteadt, Y. Kazanovich, and O. Vinogradova. Oscillatory model of novelty detection. *Network: Computation in neural systems*, 12:1–20, 2001.
- [14] G. Brambilla, J. Buchli, and A.J. Ijspeert. Adaptive four legged locomotion control based on nonlinear dynamical systems. In *From Animals to Animats 9. Proceedings of the Ninth International Conference on the Simulation of Adaptive Behavior (SAB'06)*, volume 4095 of *Lecture Notes in Computer Science*. Springer Verlag, 2006.
- [15] J. Buchli, F. Iida, and A.J. Ijspeert. Finding resonance: Adaptive frequency oscillators for dynamic legged locomotion. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3903–3909, 2006.
- [16] J. Buchli and A.J. Ijspeert. A simple, adaptive locomotion toy-system. In S. Schaal, A.J. Ijspeert, A. Billard, S. Vijayakumar, J. Hallam, and J.A. Meyer, editors, *From Animals to Animats 8. Proceedings of the Eighth International Conference on the Simulation of Adaptive Behavior (SAB'04)*, pages 153–162. MIT Press, 2004.
- [17] J. Buchli, L. Righetti, and A.J. Ijspeert. A dynamical systems approach to learning: a frequency-adaptive hopper robot. In *Proceedings of the VIIIth European Conference on Artificial Life ECAL 2005*, *Lecture Notes in Artificial Intelligence*, pages 210–220. Springer Verlag, 2005.
- [18] J. Buchli, L. Righetti, and A.J. Ijspeert. Engineering entrainment and adaptation in limit cycle systems – from biological inspiration to applications in robotics. *Biological Cybernetics*, 95(6):645–664, 2006.
- [19] J. Buchli, L. Righetti, and A.J. Ijspeert. Frequency analysis with coupled nonlinear oscillators. *Physica D*, 237(13):1705–1718, 2008.
- [20] Francesco Bullo and Andrew D. Lewis. *Geometric Control of Mechanical Systems*, volume 49 of *Texts in Applied Mathematics*. Springer Verlag, New York-Heidelberg-Berlin, 2004.
- [21] P.L. Buono and M. Golubitsky. Models of central pattern generators for quadruped locomotion. i. primary gaits. *Journal of Mathematical Biology*, 42(4):291–326, 2001.
- [22] D. Carrier. Activity of the hypaxial muscles during walking in the lizard iguana iguana. *Journal of experimental biology*, 152:453–470, 1990.

## Bibliography

---

- [23] J.G. Cham, J.K. Karpick, and M.R. Cutkosky. Stride period adaptation of a biomimetic running hexapod. *Int. Journal of Robotics Research*, 23(2):141–153, 2004.
- [24] A.H. Cohen, P.J. Holmes, and R. Rand. The nature of coupling between segmented oscillations and the lamprey spinal generator for locomotion: a mathematical model. *J. Math. Biol.*, 13:345–369, 1982.
- [25] A.H. Cohen and P. Wallen. The neural correlate of locomotion in fish. "fictive swimming" induced in a in vitro preparation of the lamprey spinal cord. *Exp. Brain Res.*, 41:11–18, 1980.
- [26] J.J. Collins and S.A. Richmond. Hard-wired central pattern generators for quadrupedal locomotion. *Biological Cybernetics*, 71(5):375–385, 1994.
- [27] J.J. Collins and I. Stewart. Hexapodal gaits and coupled nonlinear oscillator models. *Biological cybernetics*, 68:287–298, 1993.
- [28] S.H. Collins, M. Wisse, and A. Ruina. A 3-D passive-dynamic walking robot with two legs and knees. *The International Journal of Robotics Research*, 20(7):607–615, 2001.
- [29] Steve Collins, Andy Ruina, Russ Tedrake, and Martijn Wisse. Efficient Bipedal Robots Based on Passive-Dynamic Walkers. *Science*, 307(5712):1082–1085, 2005.
- [30] S. Degallier, L. Righetti, and A. J. Ijspeert. Hand placement during quadruped locomotion in a humanoidrobot: A dynamical system approach. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2047–2052, November 2007.
- [31] S. Degallier, L. Righetti, L. Natale, F. Nori, G. Metta, and A. J. Ijspeert. A modular bio-inspired architecture for movement generation for the infant-like robot icub. In *2008 IEEE International Conference on Biomedical Robotics and Biomechanics*, 2008. In press.
- [32] F. Delcomyn. Neural basis for rhythmic behaviour in animals. *Science*, 210:492–498, 1980.
- [33] I. Delvolvé, P. Branchereau, R. Dubuc, and J.-M. Cabelguen. Fictive rhythmic motor patterns induced by NMDA in an in vitro brain stem-spinal cord preparation from an adult urodele. *Journal of Neurophysiology*, 82:1074–1077, 1999.
- [34] M.R. Dimitrijevic, Y. Gerasimenko, and Pinter M.M. Evidence for a spinal central pattern generator in humans. *Ann N Y Acad Sci*, 860:360–376, 1998.

- 
- [35] D. Dimitrov, H.J. Ferreau, P.B. Wieber, and M. Diehl. On the implementation of model predictive control for on-line walking pattern generation. In *IEEE International Conference on Robotics and Automation*, pages 2685–2690, 2006.
- [36] J.M. Dixon, J.A. Tuszyński, and D. Sept. Orthogonal trajectories and analytical solutions of the Van der Pol equation without forcing. *Physics Letters A*, 239:65–71, February 1998.
- [37] D. Eck. Finding downbeats with a relaxation oscillator. *Psychological Research*, 66:18–25, 2002.
- [38] D. Eilam. Postnatal development of body architecture and gait in several rodent species. *Journal of experimental biology*, 200:1339–1350, 1997.
- [39] Ö. Ekeberg. A combined neuronal and mechanical model of fish swimming. *Biological Cybernetics*, 69:363–374, 1993.
- [40] G. Endo, Morimoto J., T. Matsubara, J. Nakanishi, and G. Cheng. Learning cpg-based biped locomotion with a policy gradient method: Application to a humanoid robot. *International Journal of Robotics Research*, 27(2):213–228, February 2008.
- [41] G. Endo, J. Nakanishi, J. Morimoto, and G. Cheng. Experimental studies of a neural oscillator for biped locomotion with qrio. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 598–604, Barcelona, Spain, 2005.
- [42] B. Ermentrout. An adaptive model for synchrony in the firefly *pterptyx malaccae*. *J. Math. Biol.*, 29:571–585, 1991.
- [43] M. S. Fischer, N. Schilling, M. Schmidt, D. Haarhaus, and H. Witte. Basic limb kinematics of small therian mammals. *Journal of experimental biology*, 205:1315–1338, 2002.
- [44] S. Frigon and S. Rossignol. Experiments and models of sensorimotor interactions during locomotion. *Biological Cybernetics*, 95(6):607–627, 2006.
- [45] L. M. Frolich and A. A. Biewener. Kinematic and electromyographic analysis of the functional role of the body axis during terrestrial and aquatic locomotion in the salamander *ambystoma tigrinum*. *Journal of experimental biology*, 162:107–130, 1992.
- [46] Y. Fukuoka, H. Kimura, and A.H. Cohen. Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts. *The International Journal of Robotics Research*, 3–4:187–202, 2003.

## Bibliography

---

- [47] A. Gams, S. Degallier, A.J. Ijspeert, and J. Lenarčič. Dynamical system for learning the waveform and frequency of periodic signals & application to drumming. In *Proceedings of the 17th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD2008)*, 2008.
- [48] A. Gams, L. Righetti, A. J. Ijspeert, and J. Lenarčič. A dynamical system for online learning of periodic movements of unknown waveform and frequency. In *2008 IEEE International Conference on Biomedical Robotics and Biomechanics*, 2008. In Press.
- [49] E. Goldfield. Transition from rocking to crawling: postural constraints on infant movement. *Developmental Psychology*, 25(6):913–919, 1989.
- [50] M. Golubitsky and I. Stewart. *The symmetry perspective: from equilibrium to chaos in phase space and physical space*. Basel, Boston, Berlin: Birkhäuser, 2002.
- [51] M. Golubitsky and I. Stewart. Nonlinear dynamics of network: the groupoid formalism. *Bulletin of the American Mathematical Society*, 43:305–364, 2006.
- [52] M. Golubitsky, I. Stewart, P.-L. Buono, and J.J. Collins. A modular network for legged locomotion. *Physica D*, 115:56–72, 1998.
- [53] M. Golubitsky, I. Stewart, P.L. Buono, and J.J. Collins. Symmetry in locomotor central pattern generators and animal gaits. *Nature*, 401(6754):693–695, October 1999.
- [54] M. Golubitsky, I. Stewart, and A. Torok. Patterns of synchrony in coupled cell networks with multiple arrows. *SIAM Journal of applied dynamical systems*, 4(1):78–100, 2005.
- [55] J. Gray. Studies in the mechanics of the tetrapod skeleton. *Journal of experimental biology*, 20:88–116, 1944.
- [56] S. Grillner. Locomotion in vertebrates: Central mechanisms and reflex interaction. *Physiological reviews*, 55(2):247–304, April 1975.
- [57] S. Grillner. Neurobiological bases of rhythmic motor acts in vertebrates. *Science*, 228(4696):143–149, 1985.
- [58] S. Grillner and P. Walln. Central pattern generators for locomotion, with special reference to vertebrates. *Annual reviews in Neurosciences*, 8:233–261, 1985.
- [59] M. Hildebrand. Symmetrical gaits of horses. *Science*, 150(3697):701–708, November 1965.

- [60] M. Hildebrand. Symmetrical gaits of primates. *American Journal of Physical Anthropology*, 26:119–130, 1967.
- [61] M. Hildebrand. Analysis of asymmetrical gaits. *journal of mammalogy*, 58(2):131–156, 1977.
- [62] S. Hyon, J. G. Hale, and G. Cheng. Full-body compliant human-humanoid interaction: Balancing in the presence of unknown external forces. *IEEE Transactions on Robotics*, 23(5):884–898, 2007.
- [63] S.H. Hyon, J. Morimoto, and G. Cheng. Hierarchical motor learning and synthesis with passivity-based controller and phase oscillator. In *IEEE International Conference on Robotics and Automation (ICRA08)*, pages 2705–2710, 2008.
- [64] F. Iida, G. Gomez, and R. Pfeifer. Exploiting body dynamics for controlling a running quadruped robot. In *12th International Conference on Advanced Robotics (ICAR05)*, page 229235, 2005.
- [65] 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*, 315(5817):1416–1420, 2007.
- [66] 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.
- [67] A.J. Ijspeert. Central pattern generators for locomotion control in animals and robots: a review. *Neural Networks*, 21(4):642–653, 2008.
- [68] A.J. Ijspeert and A. Crespi. Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA 2007)*, pages 262–268, 2007.
- [69] A.J. Ijspeert, A. Crespi, and J.M. Cabelguen. Simulation and robotics studies of salamander locomotion. Applying neurobiological principles to the control of locomotion in robots. *Neuroinformatics*, 3(3):171–196, 2005.
- [70] A.J. Ijspeert, J. Hallam, and D. Willshaw. Evolving swimming controllers for a simulated lamprey with inspiration from neurobiology. *Adaptive Behavior*, 7(2):151–172, 1999.
- [71] A.J. Ijspeert, J. Nakanishi, and S. Schaal. Learning rhythmic movements by demonstration using nonlinear oscillators. In *Proceedings of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS2002)*, pages 958–963, 2002.



## Bibliography

---

- [72] A.J. 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)*, pages 1398–1403, 2002.
- [73] R. Kempter, W. Gerstner, and J.L. van Hemmen. Hebbian learning and spiking neurons. *Physical Review E*, 59:4498–4514, 1999.
- [74] H.K. Khalil. *Nonlinear Systems*. Prentice Hall, 1996.
- [75] H. Kimura, S. Akiyama, and K. Sakurama. Realization of dynamic walking and running of the quadruped using neural oscillators. *Autonomous Robots*, 7(3):247–258, 1999.
- [76] H. Kimura, Y. Fukuoka, and A.H. Cohen. Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts. *International Journal of Robotics Research*, 26(5):475–490, 2007.
- [77] H. Kimura, Y. Fukuoka, and K. Konaga. Adaptive dynamic walking of a quadruped robot using a neural system model. *Adv. Robot.*, 15:859–878, 2001.
- [78] M. Kobilarov, M. Desbrun, J. E. Marsden, and G. S. Sukhatme. A discrete geometric optimal control framework for systems with symmetries. In *Proceedings of the 2007 Conference on Robotics: Science and Systems*, pages 1–8, 2007.
- [79] J.Z. Kolter and A.Y. Ng. Learning omnidirectional path following using dimensionality reduction. In *Proceedings of Robotics: Science and Systems*, 2007.
- [80] N. Kopell and G.B. Ermentrout. Coupled oscillators and the design of central pattern generators. *Mathematical biosciences*, 90:87–109, 1988.
- [81] Y. Kuramoto. *Chemical oscillations, Waves, and Turbulence*. Springer Verlag Berlin Heidelberg, 1984.
- [82] E.W. Large and J.F. Kolen. Resonance and the perception of musical meter. *Connection science*, 6:177–208, 1994.
- [83] S.G. Larson, D. Schmitt, P. Lemelin, and M.W. Hamrick. Uniqueness of primate forelimb posture during quadrupedal locomotion. *American Journal of Physical Anthropology*, 112:87–101, 2000.
- [84] S.G. Larson, D. Schmitt, P. Lemelin, and M.W. Hamrick. Limb excursion during quadrupedal walking: how do primates compare to other mammals? *Journal of Zoology*, 255:353–365, 2001.

- 
- [85] K. Matsuoka. Sustained oscillations generated by mutually inhibiting neurons with adaptation. *Biol. Cybern.*, 52:367–376, 1985.
- [86] T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 9(2):62–82, 1990.
- [87] T. A. McMahon. *Muscles, Reflexes, and Locomotion*. Princeton University Press, 1984.
- [88] T.A. McMahon. The role of compliance in mammalian running gaits. *Journal of experimental biology*, 115:263–282, 1985.
- [89] G. Metta, D. Vernon, and G. Sandini. The robotcub approach to the development of cognition: Implications of emergent systems for a common research agenda in epigenetic robotics. In Kaplan F. Kozima H. Yano H. Konczak J. Metta G. Nadel J. Sandini G. Stojanov G. Berthouze, L. and C. Balkenius, editors, *Fifth International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems (EpiRob2005)*, pages 111–115, 2005.
- [90] O. Michel. Webots: Professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):39–42, 2004.
- [91] K. Minassian, B. Jilge, F. Rattay, M.M. Pinter, H. Binder, F. Gerstenbrand, and M.R. Dimitrijevic. Stepping-like movements in humans with complete spinal cord injury induced by epidural stimulation of the lumbar cord: electromyographic study of compound muscle action potentials. *Spinal Cord*, 42:401–416, 2004.
- [92] S. Morgenthaler. *Introduction à la statistique*. Presses polytechniques et universitaires romandes, 2001.
- [93] J. Morimoto, G. Endo, J. Nakanishi, D. Bentevegna, and C.G. 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.
- [94] V.H. Mucino, M.M. Rodgers, M.B. Harris, and K.H. Yang. Three dimensional characterization of infant crawling kinematics. In J.L. Stein, editor, *Biomechanics of normal and prosthetic gait: presented at the Winter Annual Meeting of the American Society of Mechanical Engineers*, pages 101–107, December 1987.
- [95] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato. Learning from demonstration and adaptation of locomotion with dynamical movement primitives. *Robotics and Autonomous Systems*, 47:79–91, 2003.

## Bibliography

---

- [96] C. Niemitz. Kinematics and ontogeny of locomotion in monkeys and human babies. *Zeitschrift fur Morphologie und Anthropologie*, 83:383–400, March 2002.
- [97] J. Nishii. A learning model for oscillatory networks. *Neural networks*, 11:249–257, 1998.
- [98] J. Nishii. Learning model for coupled neural oscillators. *Network: Computation in neural systems*, 10:213–226, 1999.
- [99] T. Nishikawa, F.C. Hoppensteadt, and Y.C. Lai. Oscillatory associative memory network with perfect retrieval. *Physica D*, 197:134–148, 2004.
- [100] G. N. Orlovsky, T. G. Deliagina, and S. Grillner. *Neuronal control of locomotion: from mollusc to man*. Oxford University Press, 1999.
- [101] K. Pearson. Neural adaptation in the generation of rhythmic behavior. *Annual review of physiology*, 62:723–753, 2000.
- [102] J. Peters and S. Schaal. Reinforcement learning of motor skills with policy gradients. *Neural Networks*, 21:682–697, 2008.
- [103] A. Pikovsky, R. Rosenblum, and J. Kurths. *Synchronization, A universal concept in nonlinear sciences*, volume 12 of *Cambridge Nonlinear Science Series*. Cambridge University Press, Cambridge, UK, 2001.
- [104] R. Playter, M. Buehler, and Raibert M. Bigdog. In *Proceedings of SPIE*, volume 6320, 2006.
- [105] J. Pratt, C.M. Chew, A. Torres, P. Dilworth, and G. Pratt. Virtual model control: An intuitive approach for bipedal locomotion. *The International Journal of Robotics Research*, 20(2):129–143, 2001.
- [106] J. Pratt, P. Dilworth, and G. Pratt. Virtual model control of bipedal walking robot. In *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA1997)*, pages 193–198, 1997.
- [107] M.H. Raibert. Trotting, pacing and bounding by a quadruped robot. *Journal of Biomechanics*, 23:79–98, 1990.
- [108] L. Righetti. Control and synchronization with nonlinear dynamical systems for an application to humanoid robotics. Master’s thesis, Ecole Polytechnique Fédérale de Lausanne, March 2004.

- [109] L. Righetti, J. Buchli, and A.J. Ijspeert. From dynamic hebbian learning for oscillators to adaptive central pattern generators. In *Proceedings of 3rd International Symposium on Adaptive Motion in Animals and Machines – AMAM 2005*. Verlag ISLE, Ilmenau, 2005. Full paper on CD.
- [110] L. Righetti, J. Buchli, and A.J. Ijspeert. Dynamic hebbian learning in adaptive frequency oscillators. *Physica D*, 216(2):269–281, 2006.
- [111] L. Righetti and A.J. Ijspeert. Design methodologies for central pattern generators: an application to crawling humanoids. In *Proceedings of Robotics: Science and Systems*, pages 191–198, 2006.
- [112] L. Righetti and A.J. Ijspeert. Programmable central pattern generators: an application to biped locomotion control. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pages 1585–1590, May 2006.
- [113] L. Righetti and A.J. Ijspeert. Design methodologies for central pattern generators: Control of quadruped robots. In preparation, 2008.
- [114] L. Righetti and A.J. Ijspeert. Pattern generators with sensory feedback for the control of quadruped locomotion. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA 2008)*, pages 819–824, May 2008.
- [115] L. Righetti, A. Nylen, K. Rosander, and A.J. Ijspeert. Is the locomotion of crawling human infants different from other quadruped mammals? Under review, 2008.
- [116] D. Ritter. Epaxial muscle function during locomotion in a lizard (*varanus salvator*) and the proposal of a key innovation in the vertebrate axial musculoskeletal system. *Journal of experimental biology*, 198:2477–2490, 1995.
- [117] D. Ritter. Axial muscle function during lizard locomotion. *Journal of experimental biology*, 199:2499–2510, 1996.
- [118] S. Rossignol. Locomotion and its recovery after spinal injury. *Current opinion in neurobiology*, 10:708–716, 2000.
- [119] S. Rossignol, R. Dubuc, and J.P. Gossard. Dynamic sensorimotor interactions in locomotion. *Physiological reviews*, 86:89–154, 2006.
- [120] S. Rutishauser, A. Sproewitz, L. Righetti, and A. J. Ijspeert. Passive compliant quadruped robot using central pattern generators for locomotion control. In *2008 IEEE International Conference on Biomedical Robotics and Biomechatronics*, 2008. In Press.

## Bibliography

---

- [121] Leyendecker S., S. Ober-Bilbaum, J.E. Marsden, and M. Ortiz. Discrete mechanics and optimal control for constrained multibody dynamics. In *Proceedings of the 6th International Conference on Multibody Systems, Nonlinear Dynamics, and Control*, pages 1–10, 2007.
- [122] G. Sandini, G. Metta, and D. Vernon. Robotcub: an open framework for research in embodied cognition. In *IEEE-RAS/RSJ International Conference on Humanoid Robots (Humanoids 2004)*, pages 13–32, 2004.
- [123] S. Schaal and C.G. Atkeson. Constructive incremental learning from only local information. *Neural Computation*, 10(8):2047–2084, 1998.
- [124] M. Schmidt. Hind limb proportions and kinematics: are small primates different from other small mammals? *Journal of experimental biology*, 208:3367–3383, 2005.
- [125] D. Schmitt. Compliant walking primates. *Journal of zoology*, 248:149–160, 1999.
- [126] D. Schmitt and P. Lemelin. Origins of primate locomotion: gait mechanics of the woolly opossum. *American Journal of Physical Anthropology*, 118:231–238, 2002.
- [127] G. Schöner, W.Y. Jiang, and J.A.S. Kelso. A synergetic theory of quadrupedal gaits and gait transitions. *Journal of theoretical Biology*, 142:359–391, 1990.
- [128] L. Sentis and O. Khatib. Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics*, 2(4):505–518, December 2005.
- [129] L.J. Shapiro, B. Demes, and J. Cooper. Lateral bending of the lumbar spine during quadrupedalism in strepsirhines. *Journal of human evolution*, 40:231–259, 2001.
- [130] L.J. Shapiro and D.A. Raichlen. Limb proportions and the ontogeny of quadrupedal walking in infant baboons (*papio cynocephalus*). *Journal of zoology*, 269:191–203, 2006.
- [131] M.L. Shik, F.V. Severin, and G.N. Orlovsky. Control of walking and running by means of electrical stimulation of the mid-brain. *Biophysics*, 11(4):756–765, 1966.
- [132] J.J.E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, New Jersey, 1991.
- [133] S. Strogatz. *Nonlinear Dynamics and Chaos. With applications to Physics, Biology, Chemistry, and Engineering*. Addison Wesley Publishing Company, 1994.
- [134] S.H. Strogatz and R.E. Mirollo. Splay states in globally coupled josephson arrays: analytical prediction of floquet multipliers. *Physical Review E*, 47(1):220–227, january 1993.

- 
- [135] G. Taga. Emergence of bipedal locomotion through entrainment among the neuro-musculo-skeletal system and the environment. *Physica D: Nonlinear Phenomena*, 75(1–3):190–208, 1994.
- [136] G. Taga. A model of the neuro-musculo-skeletal system for human locomotion. I. Emergence of basic gait. *Biological Cybernetics*, 73(2):97–111, 1995.
- [137] G. Taga. A model of the neuro-musculo-skeletal system for human locomotion. II. real-time adaptability under various constraints. *Biological Cybernetics*, 73(2):113–121, 1995.
- [138] G. Taga, Y. Yamagushi, and H. Shimizu. Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment. *Biological cybernetics*, 65:147–159, 1991.
- [139] K.L. Teo, C. J. Goh, and K. H. Wong. *A Unified Computational Approach to Optimal Control Problems*. John Wiley & Sons, Pitman Monographs and Surveys in Pure and Applied Mathematics, New York, 1991.
- [140] H. Traven, L. Brodin, A. Lansner, Ö. Ekeberg, P. Wallén, and S. Grillner. Computer simulations of nmda and non-nmda receptors mediated synaptic drive: sensory and supraspinal modulation of neurons and small networks. *J. of Neurophysiology*, 70(2):695–709, August 1993.
- [141] N.G. Tsagarakis, G. Metta, G. Sandini, D. Vernon, R. Beira, F. Becchi, L. Righetti, J. Santos-Victor, A.J. Ijspeert, M.C. Carrozza, and D.G. Caldwell. iCub - The Design and Realization of an Open Humanoid Platform for Cognitive and Neuroscience Research. *Journal of Advanced Robotics, Special Issue on Robotic platforms for Research in Neuroscience*, 21(10):1151–1175, 2007.
- [142] D. Vernon, G. Metta, and G. Sandini. A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. *IEEE Transactions on Evolutionary Computation, special issue on Autonomous Mental Development*, 11(1):1–31, 2007.
- [143] J.A. Vilensky. Locomotor behavior and control in human and non-human primates: comparisons with cats and dogs. *Neuroscience and behavioral reviews*, 11:263–274, 1987.
- [144] J.A. Vilensky and S.G. Larson. Primate locomotion: utilization and control of symmetrical gaits. *Annual review of Anthropology*, 18:17–35, 1989.
- [145] C. von Hofsten. An action perspective on motor development. *Trends in Cognitive Science*, 8:266–272, 2004.

## Bibliography

---

- [146] M. Vukobratovic and B. Borovac. Zero-moment point – thirty five years of its life. *International Journal of Humanoid Robots*, 1(1):157–173, 2004.
- [147] M. Vukobratovic, B. Borovac, D. Surla, and D. Stokic. *Biped locomotion. Dynamics, stability, control and application*. Springer, 1990.
- [148] W. Wang and J.J.E. Slotine. On partial contraction analysis for coupled nonlinear oscillators. *Biological Cybernetics*, 92(1):38–53, 2005.
- [149] P.B. Wieber and C. Chevallereau. Online adaptation of reference trajectories for the control of walking systems. *Robotics and Autonomous Systems*, 54(7):559–566, 2006.
- [150] M. Wieler, R.B. Stein, M. Ladouceur, M. Whittaker, A.W. Smith, S. Naaman, H. Barbeau, J. Bugaresti, and E. Aimone. Multicenter evaluation of electrical stimulation systems for walking. *archives of physical medicine and rehabilitation*, 80(5):495–500, May 1999.
- [151] T.L Williams. Phase coupling by synaptic spread in chains of coupled neuronal oscillators. *Science*, 258:662–665, 1992.





---

# CURRICULUM VITAE

---

born October 11th 1981 in Saint-Julien en Genevois, France  
Nationality French

## Education

2004-2008	PhD candidate at the Biologically Inspired Robotics Group, EPFL
2004	Master in Computer Sciences, EPFL Master thesis in the Biologically Inspired Robotics Group "Control and synchronization with nonlinear dynamical systems for an application to humanoid robotics"
1999	Scientific Baccalaureate specialty mathematics (grade Good) Lycée International de Ferney-Voltaire (France)

## Research interests

Control theory  
Nonlinear dynamical systems  
Control of locomotion  
Computational neurosciences, theoretical biology

## Academic experience

### Talks at international conferences

- 1st International Workshop on Nonlinear Dynamics and Synchronization 2008
- 4th Symposium on Adaptive Motion in Animals and Machines 2008
- Biorobotics: Research Advances, Standards, and Education: Workshop at ICRA08
- IEEE International Conference on Robotics and Automation 2008
- Robotics Science and Systems 2006

- 
- IEEE International Conference on Robotics and Automation 2006
  - 3rd Symposium on Adaptive Motion in Animals and Machines 2005
  - European Conference on Artificial Life 2003

### **Talks at universities**

- Computational learning and motor control lab - University of Southern California - May 2008
- Cognitive Sciences Brown Bag Talk @ Artificial Intelligence Laboratory - Zurich University - April 1st, 2008
- Nonlinear Systems Laboratory - MIT - May 11th, 2006
- Robot Locomotion Group - Computer Science and Artificial Intelligence Laboratory - MIT - May 12th, 2006
- Adaptive frequency oscillators for the design of Central Pattern Generators - IIT Allahabad, March 2006

### **Teaching**

- Supervision of thirteen student projects (semester and master projects)
- Teaching in nonlinear dynamical systems (2 lectures during 3 years, 2005-2007)
- Teaching in locomotion control (1 lecture in 2007)
- Teaching assistant at the Logical Systems Laboratory, EPFL (logical systems, 2005)
- Teaching assistant at the Processor Architecture Laboratory, EPFL (processors design, 2004)

### **Program committee member**

European Conference on Artificial Life (ECAL2005)

### **Others**

Reviewer for international journals and conferences

---

## Awards

- Finalist for the Best Student Paper Award at ICRA06  
The paper “Programmable Central Pattern Generators: an application to biped locomotion control” was one of the 3 finalists for the Best Student Paper Award at the International Conference on Robotics and Automation 2006.
- Jean Landry Award (April 2004)  
Awards an original and personal scientific work, obtained for the master thesis “Control and Synchronization with Nonlinear Dynamical Systems for an application to Humanoid Robotics”

## Publications

### Journal publications

- [1] J. Buchli, L. Righetti, and A.J. Ijspeert. Frequency analysis with coupled nonlinear oscillators. *Physica D*, 237(13):1705–1718, 2008.
- [2] L. Righetti, J. Buchli, and A.J. Ijspeert. Dynamic fourier series decomposition with adaptive frequency oscillators. In preparation, 2008.
- [3] L. Righetti and A.J. Ijspeert. Design methodologies for central pattern generators: Control of quadruped robots. In preparation, 2008.
- [4] L. Righetti, A. Nylen, K. Rosander, and A.J. Ijspeert. Is the locomotion of crawling human infants different from other quadruped mammals? Under review, 2008.
- [5] N.G. Tsagarakis, G. Metta, G. Sandini, D. Vernon, R. Beira, F. Becchi, L. Righetti, J. Santos-Victor, A.J. Ijspeert, M.C. Carrozza, and D.G. Caldwell. iCub - The Design and Realization of an Open Humanoid Platform for Cognitive and Neuroscience Research. *Journal of Advanced Robotics, Special Issue on Robotic platforms for Research in Neuroscience*, 21(10):1151–1175, 2007.
- [6] J. Buchli, L. Righetti, and A.J. Ijspeert. Engineering entrainment and adaptation in limit cycle systems – from biological inspiration to applications in robotics. *Biological Cybernetics*, 95(6):645–664, 2006.
- [7] L. Righetti, J. Buchli, and A.J. Ijspeert. Dynamic hebbian learning in adaptive frequency oscillators. *Physica D*, 216(2):269–281, 2006.

---

## Conference publications

- [1] S. Degallier, L. Righetti, L. Natale, F. Nori, G. Metta, and A. J. Ijspeert. A modular bio-inspired architecture for movement generation for the infant-like robot icub. In *2008 IEEE International Conference on Biomedical Robotics and Biomechanics*, 2008. In press.
- [2] A. Gams, L. Righetti, A. J. Ijspeert, and J. Lenarčič. A dynamical system for online learning of periodic movements of unknown waveform and frequency. In *2008 IEEE International Conference on Biomedical Robotics and Biomechanics*, 2008. In Press.
- [3] L. Matthey, L. Righetti, and A. J. Ijspeert. Experimental study of limit cycle and chaotic controllers for the locomotion of centipede robots. In *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1860–1865, September 2008.
- [4] L. Righetti and A.J. Ijspeert. Pattern generators with sensory feedback for the control of quadruped locomotion. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA 2008)*, pages 819–824, May 2008.
- [5] S. Rutishauser, A. Sproewitz, L. Righetti, and A. J. Ijspeert. Passive compliant quadruped robot using central pattern generators for locomotion control. In *2008 IEEE International Conference on Biomedical Robotics and Biomechanics*, 2008. In Press.
- [6] S. Degallier, L. Righetti, and A. J. Ijspeert. Hand placement during quadruped locomotion in a humanoidrobot: A dynamical system approach. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2047–2052, November 2007.
- [7] N. Tsagarakis, F. Becchi, L. Righetti, A. J. Ijspeert, and D. G. Caldwell. Lower body realization of the baby humanoid - icub. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3616–3622, November 2007.
- [8] S. Degallier, C. P. Santos, L. Righetti, and A. Ijspeert. Movement generation using dynamical systems: a humanoid robot performing a drumming task. In *IEEE-RAS International Conference on Humanoid Robots (HUMANOIDS06)*, 2006.
- [9] G. Metta, G. Sandini, D. Vernon, D. Caldwell, N. Tsagarakis, R. Beira, J. Santos-Victor, A. Ijspeert, L. Righetti, G. Cappiello, G. Stellin, and F. Becchi. The robotcub project – an open framework for research in embodied cognition. In

---

*Humanoids Workshop, Proceedings of the IEEE–RAS International Conference on Humanoid Robots*, December 2006.

- [10] L. Righetti and A.J. Ijspeert. Design methodologies for central pattern generators: an application to crawling humanoids. In *Proceedings of Robotics: Science and Systems*, pages 191–198, 2006.
- [11] L. Righetti and A.J. Ijspeert. Programmable central pattern generators: an application to biped locomotion control. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pages 1585–1590, May 2006.
- [12] J. Buchli, L. Righetti, and A.J. Ijspeert. A dynamical systems approach to learning: a frequency-adaptive hopper robot. In *Proceedings of the VIIIth European Conference on Artificial Life ECAL 2005*, Lecture Notes in Artificial Intelligence, pages 210–220. Springer Verlag, 2005.
- [13] L. Righetti, J. Buchli, and A.J. Ijspeert. From dynamic hebbian learning for oscillators to adaptive central pattern generators. In *Proceedings of 3rd International Symposium on Adaptive Motion in Animals and Machines – AMAM 2005*. Verlag ISLE, Ilmenau, 2005. Full paper on CD.
- [14] M. Vuletic, L. Righetti, L. Pozzi, and P. Ienne. Operating system support for interface virtualisation of reconfigurable coprocessors. In *Workshop on Application Specific Processors (WASP-2)*, 2004.
- [15] M. Vuletic, L. Righetti, L. Pozzi, and P. Ienne. Operating system support for interface virtualisation of reconfigurable coprocessors. In *In Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, 2004.
- [16] L. Righetti, S. Shokur, and M. Capcarre. Evolution of fault-tolerant self-replicating structures. In Banzhaf et al., editor, *The Seventh European Conference on Artificial Life: ECAL'2003*, 2003.

## Abstracts

- [1] L. Righetti, J. Buchli, and A.J. Ijspeert. Adaptive frequency oscillators and applications. In *Proceedings of the 1st International Workshop on Nonlinear Dynamics and Synchronization*, 2008. Abstract.
- [2] L. Righetti and A.J. Ijspeert. Quadruped locomotion: from infants crawling to the design of pattern generators for quadruped robots. In *Proceedings of the 4th International Symposium on Adaptive Motion in Animals and Machines – AMAM 2008*, 2008. Abstract.

- 
- [3] J. Buchli, L. Righetti, and A.J. Ijspeert. Adaptive dynamical systems: A promising tool for embodied artificial intelligence. In *Proceedings of 50th anniversary summit of artificial intelligence*, July 2006. Abstract.
- [4] J. Buchli, L. Righetti, and A.J. Ijspeert. Adaptive frequency oscillators applied to dynamic walking II. adapting to resonant body dynamics. In *Proceedings of Dynamic Walking*, 2006. Abstract.
- [5] L. Righetti, J. Buchli, and A.J. Ijspeert. Adaptive frequency oscillators applied to dynamic walking I. programmable pattern generators. In *Proceedings of Dynamic Walking*, 2006. Abstract.
- [6] L. Righetti and A.J. Ijspeert. Design methodologies for central pattern generators: towards 'intelligent' locomotion in robots. In *Proceedings of 50th anniversary summit of artificial intelligence*, July 2006. Abstract.
- [7] J. Buchli, L. Righetti, and A.J. Ijspeert. Adaptive dynamical systems for movement control. In *Proceedings of 3rd International Symposium on Adaptive Motion in Animals and Machines – AMAM 2005*, page 7. Verlag ISLE, Ilmenau, 2005. Abstract.
- [8] L. Righetti, J. Buchli, and A.J. Ijspeert. From dynamic hebbian learning for oscillators to adaptive central pattern generators. In *Proceedings of 3rd International Symposium on Adaptive Motion in Animals and Machines – AMAM 2005*, page 45. Verlag ISLE, Ilmenau, 2005. Abstract.

## Non peer-reviewed publications

- [1] L. Righetti. *Control of locomotion using dynamical systems: design methods and adaptive frequency oscillators*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, November 2008.
- [2] L. Righetti, J. Buchli, and A.J. Ijspeert. Programmable central pattern generators. In Auke Jan Ijspeert, Jonas Buchli, Allen Selverston, Mikhail Rabinovich, Martin Hasler, Wulfram Gerstner, Aude Billard, Henry Markram, and Dario Floreano (Editors), editors, *Dynamical principles for neuroscience and intelligent biomimetic devices*, pages 129–130. EPFL, 2006. ISBN 978-2-8399-0134-5.
- [3] A.J. Ijspeert, J. Buchli, A. Crespi, L. Righetti, and Y. Bourquin. Institute presentation: Biologically inspired robotics group at EPFL. *International Journal of Advanced Robotics Systems*, 2(2):175–199, 2005.

- 
- [4] L. Righetti. Control and synchronization with nonlinear dynamical systems for an application to humanoid robotics. Master's thesis, Ecole Polytechnique Fédérale de Lausanne, March 2004.