Locomotion in Modular Robots based on Central Pattern Generators

A. Sproewitz, R. Moeckel, J. Maye, M. Asadpour, A. J. Ijspeert*
School of Computer and Communication Science
EPFL - Ecole Polytechnique Fédérale de Lausanne
Station 14, CH-1015 Lausanne, Switzerland
{*Alexander.Sproewitz, Jerome.Maye, Auke.Ijspeert*}*@epfl.ch, masoud.asadpour@gmail.com, moeckel@ini.phys.ethz.ch*

Abstract We address the problem of learning to locomote in modular robotic systems, i.e. systems made from multiple homogenous or heterogenous and highly autonomous modules being assembled into a bigger robotic structure, other than monolithic robots (such as humanoid robot systems or predefined quadruped robots). We are particulary interested in *online* learning of robust and adaptive locomotion for arbitrary robotic structures. We are using Central Pattern Generators (an approach inspired by nature coping with redundancies in animal bodies and the easy generation of locomotion patterns) in combination with a gradient-free optimization method (Powell's method). We applied our approach on three different robot configurations, interesting locomotion modes are obtained after running the optimization for less than 60 min. Our CPG model can be implemented easily in a distributed system among the modules with a low demand for computational power, it exhibits limit cycle behaviour (temporary perturbations are rapidly forgotten), the CPG produces smooth trajectories when control parameters are abruptly changed, and it is robust against imperfect communication between modules. As a testing and application platform we have developed the modular robotic system YaMoR. Among other key features it provides wireless communication among the several YaMoR units based on a specially developed Scatternet Protocol (SNP). As for the future use as an autonomous reconfigurable system (Self-Reconfigurable Modular Robots) we are exploring reconfiguration strategies based on a central approach using graph signatures.

Introduction As for all mobile robots, one of the key features for a modular robot is robust locomotion. Designing efficient locomotion controllers for such modular systems is however a difficult and unsolved problem. Their locomotion control suffers from all the traditional difficulties of locomotion control in robots with multiple degrees of freedom. Additional difficulties arise due to their modular structure. (Self-reconfiguring) modular robots are not necessarily optimized for a specific locomotion tasks (sub-optimal in terms of balance and mass distribution, sufficient actuation, alignment and amount of degree of freedom). Therefore the control of locomotion requires multi-dimensional coordinated rhythmic patterns that need to be correctly tuned such as to satisfy multiple constraints: the capacity to generate forward motion possibly with low energy consumption, without falling over, while adapting to possibly complex terrain (uneven ground, obstacles), and while allowing the modulation of speed and direction.

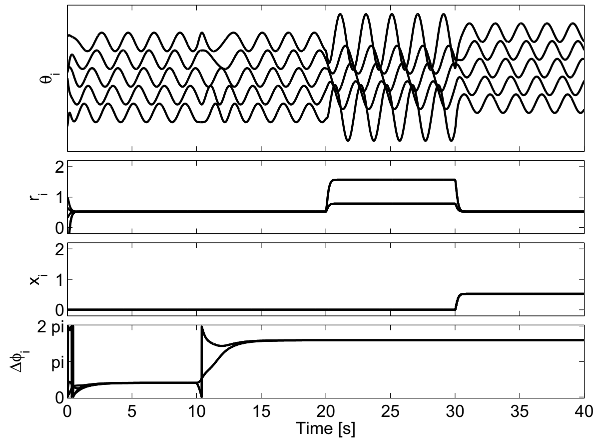

Figure 1: Example of abrupt parameter changes in the CPG model. Output signals θ_i of the 5 oscillators (top). Amplitude state variables r_i (second from top). Offset state variables x_i (third from top). Phase differences $\Delta\phi_i = \phi_{i+1} - \phi_i$ between neighbor oscillators (bottom).

Locomotion control We propose a framework for learning locomotion controllers based on two components: a central pattern generator and a gradient-free optimization algorithm: Powell's method (Press et al., 1994). Central pattern generators (CPGs) are neural networks capable of producing coordinated patterns of rhythmic activity without any rhythmic inputs from sensory feedback or from higher control centers (Delcomyn, 1980). We present a CPG based on a system of N coupled amplitude-controlled phase oscillators with N being the number of connected active modules per robot configuration. See Fig. 1 for an example of 5 oscillators coupled in series, while control parameters are being abruptly changed. Our CPG is implemented and tested on our YaMoR modular robotic system. A YaMoR module has one actuated degree of freedom (RC hobby servomotor, 1.1 Nm torque), can be mechanically attached up to 5 other modules (using a pin and groove system, see Fig. 2 for a 8-module assembly),

*To whom correspondence should be addressed.

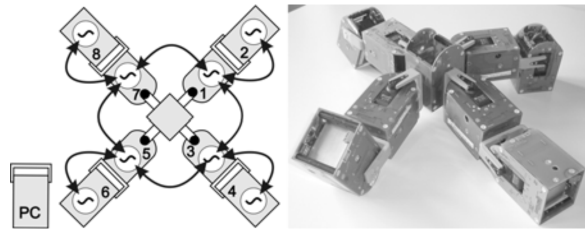

Figure 2: Quadruped robot assembly for 8 YaMoR modules (right) and its CPG structure (left). Solid lines show the bidirectional couplings between oscillators. The inner nodes (1, 3, 5 and 7) form a closed loop of interoscillator couplings.

has a self-sufficient power supply for approx. 1.5 h and onboard computation abilities, and communicates wirelessly with other modules via Bluetooth (Moeckel et al., 2006). Each module is programmed to run one nonlinear oscillator to control the oscillations of its servomotor. Nonlinear oscillators are coupled together across modules using Bluetooth communication to obtain specific gaits, i.e. synchronized patterns of oscillations among modules. Different stable gaits can be obtained by adjusting the parameters of the CPG that determine the frequency, amplitude and phase lag of oscillations.

Online optimization CPGs are also a good substrate for online optimization, a property that we focus on in this paragraph. Indeed an optimization algorithm, in our case Powell's method, can run in parallel to the CPG and regularly update its parameters via Bluetooth. Even so the updates representing abrupt changes, the CPG produces trajectories that smoothly converge towards the new limit cycle after a short transient period. Hence there is no need to stop or reset the robot between iterations. We focus on *online* optimization, i.e. learning while moving other than using a simulation or a model of our robot and the environment. As a first reason the transfer from a simulation/model to the real world is often problematic due to difficulties in modeling complex environments (parameters for friction and contact models; inhomogenious friction in real world; backlash and offsets in the motors; uncertainties in actuaction e.g. due to decreasing battery power). Second, it allows learning a locomotion gait for a new, previously unknown configuration such as after self-reconfiguration or self-assembly. Third, the locomotion gait may be continuously adapted to changes in the robotic structure (e.g. because of the addition, removal, or failure of modules). Fourth, locomotion can be adapted to changes in the environment. This work follows preliminary results in simulation with a slightly different type of CPG that had shown the potential of the approach for modular robotics (Marbach and Ijspeert, 2005). We applied our combined CPG/optimization method at three different robot structures: a snake configuration with two open parameters, a tripod robot with six and a quadruped assembly with 5 open optimization parameters (see Fig. 2 for the CPG implementation and the mechanical assembly of the quadruped robot).

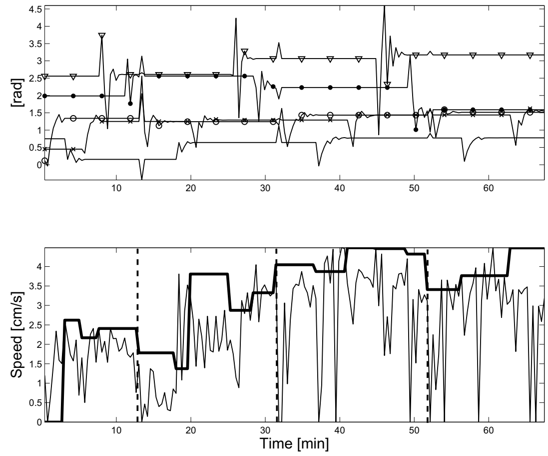

Figure 3: Powell's optimization for the quadruped robot, with five open parameters.

We use as a cost function the speed of the robot, based on the distance covered by locomotion for 8 s. Each evaluation of the optimization method takes in total 23 sec, we obtain interesting gaits (around 4.5 cm/s) for the quadruped robot within 60 min (Fig. 3). The two parameters for adjusting Powell's method are not specially fine-tuned but being rather selected to work for all our robot configurations.

The above combined approach to generate optimized locomotion patterns on arbitrary modular robot configurations, using CPGs and a gradient-free optimization algorithm shows good results for a very small amount of evaluations (compared to e.g., stochastic optimization methods). This provides a way for an online learning strategy, running CPGs and the optimization method in parallel on the real hardware and in real time. We would like to extend this adaptive locomotion learning to changing modular robot topologies (e.g., simulated growing, shrinking, possibly removing and adding limbs) and environmental situations.

References

Delcomyn, F. (1980). Neural basis for rhythmic behaviour in animals. *Science*, 210:492–498.

Marbach, D. and Ijspeert, A. J. (2005). Online optimization of modular robot locomotion. In *Proceedings of the IEEE Int. Conference on Mechatronics and Automation (ICMA 2005)*, pages 248–253.

Moeckel, R., Jaquier, C., Drapel, K., Dittrich, E., Upegui, A., and Ijspeert, A. J. (2006). Exploring adaptive locomotion with YaMoR, a novel autonomous modular robot with Bluetooth interface. *Industrial Robot*, 33(4):285–290.

Press, W., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1994). *Numerical recipes in C : the art of scientific computing, 2nd edition*. Cambridge University Press.