CONTROLO 2006
7th Portuguese Conference on Automatic Control
Instituto Superior Técnico, Lisboa, Portugal
September 11-13, 2006

TEMPORAL COORDINATION OF SIMULATED TIMED TRAJECTORIES FOR TWO VISION-GUIDED VEHICLES: A NONLINEAR DYNAMICAL SYSTEMS APPROACH

Cristina P Santos *

* *University of Minho, Industrial Electronics Department*

Abstract: We present an attractor based dynamics that autonomously generates temporally discrete movements and temporally coordinated movements for two vehicles, stably adapted to changing online sensory information. Movement termination is entirely sensor driven. We build on a previously proposed solution in which timed trajectories and sequences of movements were generated as attractor solutions of dynamic systems. We present a novel system composed of two coupled dynamical architectures that temporally coordinate the solutions of these dynamical systems. The coupled dynamics enable synchronization of the different components providing an independence relatively to the specification of their individual parameters.
We apply this architecture to generate temporally coordinated trajectories for two vision-guided mobile robots in a simulated environment, whose goal is to reach a target in an approximately constant time while navigating within a non-structured environment. The results illustrate the robustness of the proposed decision-making mechanism and show that the two vehicles are temporal coordinated: they terminate their movements approximately simultaneously.

Keywords: Non-linear dynamical systems, autonomy, robotics, timed trajectories, competitive dynamics, timing

1. INTRODUCTION

This article addresses the problem of generating timed trajectories and temporally coordinated movements for two wheeled vehicles, when relatively low-level, noisy sensorial information is used to steer action. The developed architectures are fully formulated in terms of nonlinear dynamical systems which lead to a flexible timed behavior stably adapted to changing online sensory information. The generated trajectories have controlled and stable timing (limit cycle type solutions). Incoupling of sensory information enables sensor driven termination of movement.

Specifically, we address the following questions: Can the temporal coordination among different degrees-of-freedom (dofs) be applied to the robotics domain such that a tendency to synchronize among two vehicles is achieved? Can the applied dynamical systems approach provide a theoretically based way of tuning the movement parameters such that it is possible to account for relationships among these?

These questions are positively answered and shown in an exemplary simulation in which two low-level vehicles temporally coordinated must reach a target within a certain time independently of the environment configuration or the distance to the target. The results illustrate the robustness of the proposed decision-making mechanism and show that the two vehicles are temporal coordinated: if a robot movement is affected by the environment configuration such that it will take longer to reach the target, the control level coordinates the two robots such that they terminate approximately simultaneously. The proposed solution provides a flexible and reactive framework for adaptive motor planning and control that reduces the dimensionality of the control problem.

We build on previous work (Santos, 2004; Schöner and Santos, 2001; Schöner, 1994), where we have shown that the proposed approach is sufficient versatile to generate, through limit cycle attractors, a whole variety of rich forms of behavior, including both rhythmic and discrete tasks. The online linkage to online noisy sensorial information, was achieved through the coupling of these dynamical systems to time-varying sensory information (Schöner, 1994; Santos, 2004). In (Santos, 2004), this architecture was implemented in a real vehicle and integrated with other dynamical architectures which do not explicitly parameterize timing requirements (Santos, 2004). In (Schöner and Santos, 2001), we have generated temporally coordinated movements among two PUMA arms by coupling two such dynamical systems.

This work is innovative in the formalization and utilization of movement primitives, both in the context of biological and robotics research. Further, it significantly facilitates generation of movement, sequences of movements and temporally coordinated movements. We apply autonomous differential equations to model how behaviors related to locomotion are programmed in the oscillatory feedback systems of "central pattern generators" in the nervous systems (Schöner, 1994). Coordination can be modeled through mutual coupling of such differential equations. This coordination through coupling resembles the generation of coordinated patterns of activation in locomotory behavior of nervous systems (Santos, 2004).

The idea of using dynamic systems for movement generation is not new. For instance, the *Dynamical Systems approach to autonomous robotics*, has been developed for the control of autonomous vehicles (Schöner and Dose, 1992; Bicho *et al.*, 2000). Other solutions (Raibert, 1986; Clark *et al.*, 2000; Williamson, 1998) have tried to address the timing problem, by generating time structure at the level of control. More generally, the nonlinear control approach to locomotion pioneered by (Raibert, 1986) amounts to using limit cycle attractors that emerge from the coupling of a nonlinear dynamical control system with the physical environment of the robot. A limitation of such approaches is that they essentially generate a single motor act in rhythmic fashion and remain limited with respect to the integration of multiple constraints (but see (Schaal *et al.*, 2000) where temporally discrete movement is also generated).

The work presented in this article extends the use of oscillators to generate timed trajectories and temporally coordinated tasks on a low-level vehicle. In robotics, the control of two dofs is generally achieved by considering the dofs are completely independent. However, in motor control of biological systems this independence is not verified. Movement coordination requires some form of planning and there exist an infinite number of possible movement plans for any given task. A rich area of research has been evolving to study the computational principles and implicit constraints in the coordination of multiple dofs, specifically the question whether or not there are specific principles in the organization of central nervous systems, that coordinate the movements of individual dofs. This research has been mainly directed towards coordination of rhythmic movement (Schaal *et al.*, 2000). This coordination has been addressed within the dynamic theoretical approach (Schöner, 1994) and more recently has been applied in the robotics domain (Buchli and Ijspeert, 2004). The applied dynamic concepts are herein generalized to understand the coordination of discrete movement.

2. TIMED TRAJECTORIES GENERATION

In this article, two low level vehicles must navigate in a simulated non-structured environment while being capable of reaching a target in an approximately constant time. For each robot, target position is internally acquired by a visual system mounted over the robot and robot velocity is controlled such that the vehicle has a fixed time to reach the target while continuously avoiding sensed obstacles in its path. The two robot movements are coupled in time such that if the two movements onsets are not perfectly simultaneous or if there time trajectories are evolving differently (one is going faster/slower than the other), leading to different movement times (time it takes to reach the target), this coupling coordinates the two movements such that they terminate approximately simultaneously.

The dynamical systems formulated in order to solve this robotic problem are divided onto two integrated architectures which act out at different levels. The dynamics of heading direction act out at the level of the turning rate. The dynamics of driving speed act out at the level of the driving speed and express time constraints. The ease with which these dynamical systems are integrated providing for system integration and behavioral organization is an advantage of our approach.

2.1 Attractor dynamics of heading direction

The robot action of turning is generated by letting the robot's heading direction, ϕ_{h}, measured relative to some allocentric reference frame, vary by making ϕ_{h} the behavioral variable of a dynamical system (for a full discussion see (Schöner and Dose, 1992)). This behavioral variable is governed by a nonlinear vector field in which task constraints contribute independently by modeling desired behaviors (*target acquisition*) as attractors and undesired behaviors (*obstacle avoidance*) as repellers of the overall behavioral dynamics. Integration of the *target acquisition*,$F_{\mathrm{tar}}(\phi_h)$ and *obstacle avoidance*,$F_{\mathrm{obs}}(\phi_h)$ contributions is achieved by adding each of them to the vector field that governs heading direction dynamics

$$\frac{d\phi_h}{dt} = F_{\mathrm{obs}}(\phi_h) + F_{\mathrm{tar}}(\phi_h) + F_{\mathrm{stoch}}(\phi_h). \quad (1)$$

We add a stochastic component force, F_{stoch}, to ensure escape from unstable states within a limited time. The complete behavioral dynamics for heading direction has been implemented and evaluated in detail on a physical mobile robot (Bicho *et al.*, 2000; Santos, 2004).

2.2 The dynamical systems of driving speed

The path velocity, v of the vehicle is controlled through a dynamical system architecture that generates timed trajectories for the vehicle as described in (Santos, 2004). Specifically, timed trajectories are modeled as time courses of behavioral variables (m, n) which are stable solutions of dynamical systems. Although only the variable, m, will be used to set the velocity of the robot, a second auxiliary variable, n, is needed to enable the system to undergo periodic motion.

We set two spatially fixed coordinates systems each centered on the initial robot position: one for the x and the other for the y spatial coordinates of robot movement. A dynamical system which generates both stable oscillations (limit cycle solutions) and two stationary states (Schöner and Santos, 2001; Santos, 2004), is defined for each of these fixed coordinate systems as follows:

$$\begin{pmatrix} \dot{m}_i \\ \dot{n}_i \end{pmatrix} = 5\ |u_{\mathrm{init,i}}| \begin{pmatrix} m_i \\ n_i \end{pmatrix} + |u_{\mathrm{hopf,i}}| f_{\mathrm{hopf,i}} + 5\ |u_{\mathrm{final,i}}| \begin{pmatrix} m_i - A_{\mathrm{ic}} \\ n_i \end{pmatrix} + \mathrm{gwn}, \quad (2)$$

where the index $i = x, y$ refers to dynamics of x and y spatial coordinates of robot movement. A neural dynamics controls the switching between the three regimes through three "neurons" $u_{\mathrm{j,i}}$ ($j = \mathrm{init, hopf, final}$). The "init" and "final" contributions generate stable stationary solutions at $m_i = 0$ for "init" and A_{ic} for "final" with $n_i = 0$ for both. These states are characterized by a time scale of $\tau = 1/5 = 0.2$.

Herein, an approach is defined to achieve temporal coordination among the two robots, by coupling these two architectures in a way that generates phase-locking in the oscillation regime. This was achieved by modifying the "Hopf" contribution that generates the limit cycle solution as follows:

$$f_{\mathrm{hopf,i}} = \begin{pmatrix} \alpha & -\omega \\ \omega & \alpha \end{pmatrix} \begin{pmatrix} m_{\mathrm{i}} - \frac{A_{\mathrm{ic}}}{2} \\ n_{\mathrm{i}} \end{pmatrix} - \gamma_{\mathrm{i}} \left(\left(m_{\mathrm{i}} - \frac{A_{\mathrm{ic}}}{2} \right)^2 + n_{\mathrm{i}}^2 \right) \begin{pmatrix} m_{\mathrm{i}} - \frac{A_{\mathrm{ic}}}{2} \\ n_{\mathrm{i}} \end{pmatrix} + c|u_{\mathrm{hopf,j}}| \begin{pmatrix} \cos\theta_{\mathrm{ij}} & -\sin\theta_{\mathrm{ij}} \\ \sin\theta_{\mathrm{ij}} & \cos\theta_{\mathrm{ij}} \end{pmatrix} \begin{pmatrix} m_{\mathrm{j}} \\ n_{\mathrm{j}} \end{pmatrix} \quad (3)$$

where index j refers to index i time courses of the coupled dynamical system (the other robot), $\gamma_{\mathrm{i}} = \frac{4\ \alpha}{A_{\mathrm{ic}}^2}$ defines amplitude of Hopf contribution and θ_{ij} is the desired relative phase among oscillators i and j ($-\theta_{\mathrm{ij}}$ among oscillators j and i). For instance, for x spatial coordinates $(m_{\mathrm{x}}, n_{\mathrm{x}})$ of robot 1 is coupled with $(m_{\mathrm{x}}, n_{\mathrm{x}})$ of robot 2. The coupling term is multiplied with the neuronal activation of the other system's Hopf state so that coupling is effective only when both components are in the oscillation regime. Because we want both coupled dynamical systems to be in-phase we set $\theta_{\mathrm{ij}} = 0$ degrees. This "Hopf" contribution provides a stable periodic solution (limit cycle attractor) with cycle time $T = \frac{2\pi}{\omega} = 20$s. We use it because it can be completely solved analytically, providing complete control over its stable states. This analytical specification is an innovative aspect of our work. Relaxation to the limit cycle solution occurs at a time scale of $1/(2\ \alpha) = 0.2$ time units.

The dynamics of (2) are augmented by a Gaussian white noise term, gwn, that guarantees escape from unstable states and assures robustness to the system.

2.2.1. Neural dynamics

The "neuronal" dynamics of $u_{\mathrm{j,i}} \in [-1, 1]$ ($j = \mathrm{init, final, hopf}$) switches the dynamics from the initial and final posture states into the oscillatory regime and back. The competitive dynamics are given by

$$\alpha_{\mathrm{u}} \dot{u}_{\mathrm{j,i}} = \mu_{\mathrm{j,i}} u_{\mathrm{j,i}} - |\mu_{\mathrm{j,i}}| u_{\mathrm{j,i}}^3 - 2.1 \sum_{a \neq j} u_{\mathrm{a,i}}^2 u_{\mathrm{j,i}} + \mathrm{gwn} \quad (4)$$

where "neurons" can go "on" (=1) or "off" (=0). This dynamics enforces competition among task constraints depending on the neural competitive parameters ("competitive advantages"), μ_i. The neuron, u_i, with the largest competitive advantage, $\mu_i > 0$, is likely to win the competition, although for sufficiently small differences between the different μ_i values multiple outcomes are possible (the system is multistable).

In order to control switching, the μ_i parameters are explicitly designed as functions of user commands, sensory events, or internal states and control the sequential activation of the different neurons (see (Steinhage and Schöner, 1998), for a general framework for sequence generation based on these ideas). We vary the μ-parameters between the values 1.5 and 3.5: $\mu_i = 1.5 + 2b_i$, where b_i are "quasi-boolean" factors taking on values between 0 and 1 (with a tendency to have values either close to 0 or close to 1). Hence, we assure that one neuron is always "on". Herein, the time, t, and target location, fully control the neural dynamics through the quasi-boolean parameters. A sequence of neural switches is generated by translating sensory conditions and logical constraints into values for these parameters(see (Santos, 2004) for a description).

The time scale of the neuronal dynamics is set to a relaxation time of $\tau_u = 0.02$, ten times faster than the relaxation time of the (m, n) dynamical variables. By using different time scales one can design the several dynamical systems separately (Santos, 2004; Steinhage and Schöner, 1998).

Temporally discrete movement is autonomously generated through a sequence of neural switches such that an oscillatory state exists during an appropriate time interval of about a half-cycle. This approximately half-cycle is movement time (MT), here $MT = 10$s.

2.3 Coupling to sensorial information

Ball position is acquired by simulating a camera mounted on the robot. The goal is to robustly detect a red ball standing in an unstructured, complex environment.

We apply a color based real-time tracker, Continuously Adaptive Mean Shift (CAMSHIFT) algorithm (Bradski, 1998), that handles several computer-vision application problems during its operation. This algorithm tracks the x', y' image coordinates and area of the color blob representing the red ball. A perspective projection model transforms the x', y' image coordinates onto the x, y world coordinates. To simulate sensor noise (which can be substantial if such optical measures are extracted from image sequences), we added either white or colored noise to the image coordinates. Here we show simulations that used colored noise, ζ, generated from

$$\dot{\zeta} = -\frac{1}{\tau_{\text{corr}}}\,\zeta + \sqrt{Q}\ \text{gwn} \tag{5}$$

where gwn is gaussian white noise with zero mean and unit variance, so that $Q = 5$ is the effective variance. The correlation time, τ_{corr}, was chosen as 0.2 sec. The simulated target location was thus

$$\begin{aligned} x_{\text{target}} &= \text{true } x_{\text{target}} + \zeta(t) \\ y_{\text{target}} &= \text{true } x_{\text{target}} + \zeta(t) \end{aligned} \tag{6}$$

2.4 Velocity

Robot velocity is controlled by a dynamics similar to that described in (Bicho *et al.*, 2000), such that the planning variable is in or near a resulting attractor of the dynamical system most of the time. This dynamics assures that velocity depends whether or not obstacles are detected for the current heading direction value. In case an obstacle has been detected, velocity is set as V_{obs}, which is computed as a function of the current distance to the obstacle (Bicho *et al.*, 2000). In case no obstacle has been detected, velocity is set as V_{timing}:

$$V_{\text{timing}} = \sqrt{\dot{m}_{\text{x}} + \dot{m}_{\text{y}}}\ , \tag{7}$$

where $m_{\text{x}}, m_{\text{y}}$ are the dynamical variables.

In the following, we briefly explain the dynamic architecture behavior of each robot. At $t = 0$ s the robot is resting at its initial fixed position, $x_{\text{R}_{\text{init}}}, y_{\text{R}_{\text{init}}}$. The robot rotates in the spot in order to orient towards or look for the target direction. At time t_{init}, timed forward movement is initiated. The periodic motion's amplitude, A_{mc}, is updated during periodic movement each time step as follows

$$A_{\text{mc}} = (x_{\text{target}} - x_{\text{R}_{\text{init}}}) - ((x_{\text{R}} - x_{\text{R}_{\text{init}}}) - m_{\text{x}}) \tag{8}$$

where x_{target} is x target position, x_{R} is x robot position and m_{x} is the dynamical variable.

The periodic solution is deactivated again when the x vehicle position comes into the vicinity of this periodic amplitude value, and the final postural state (which equals A_{mc}) is turned on instead. The same behavior applies for the dynamical systems defined for the y spatial coordinate.

3. EXPERIMENTAL RESULTS

The dynamic architecture was simulated in Matlab/simulink (product of the MATHWORKS company). The dynamics of heading direction, timing, competitive neural, path velocity and dead-reckoning equations are numerically integrated using the Euler method. The cycle time is 70 ms and MT is 10s. Forward timed movement only starts for $t_{\text{init}} = 3$s.

In order to verify if temporal coordination among the two robot movements is achieved we have performed several experiments. Herein, due to space constraints, we illustrate only one exemplary experiment. During its path towards the target, robot 2 is faced with an obstacle which it must circumnavigate. This obstacle does not interfere with the robot 1 movement towards the target. Fig. 1 illustrates the robot motions and time stamps of these trajectories. The ball is depicted by a light circle. Small crosses around ball position indicate ball position as acquired by the vision systems. The robots path are indicated by lines formed by crosses. The interval between two consecutive crosses indicates the robot's path velocity since the time acquisition interval is constant: the smaller the velocity the closer the points. When the obstacle is no longer detected for the current heading direction, at $t = 9.1$s, robot 2 is strongly accelerated in order to compensate for the object circumnavigation.

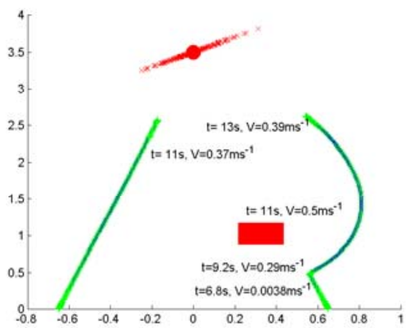

Fig. 1. A simulation run illustrating the robots' timed trajectories to meet the red ball.

Robot velocities are depicted in Fig. 2. v represents forward velocity of the robot. v_{timing} and v_{obs} represent velocity imposed by the discussed dynamical architecture and velocity imposed in case an obstacle is detected, respectively.

The proposal dynamic architecture without coupling ($c = 0$) is similar to work presented in (Santos, 2004), where results have shown that robot velocity is controlled such that the target is reached in an approximately constant time ($MT = 10$s) independently of the environment configuration and of the distance to the target.

The introduction of a coupling of this form tends to synchronize movement in the two robots. Thus, when x and/or y movement of robot 2 is affected

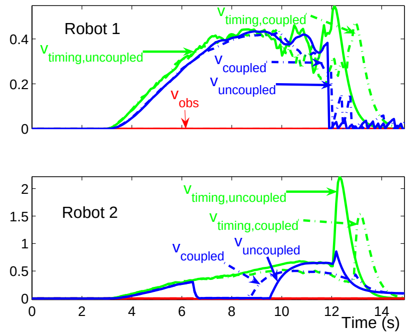

Fig. 2. Velocity variables for robot 1 and 2.

by the environment configuration such that its periodic motion amplitude is increased, robot 1 movement is coordinated through coupling such that movements of both robots terminate simultaneously. This results in delayed simultaneous switch, around $t = 12.8$s, among Hopf and final contributions for x and y dynamical systems of both robots (see Fig. 3).

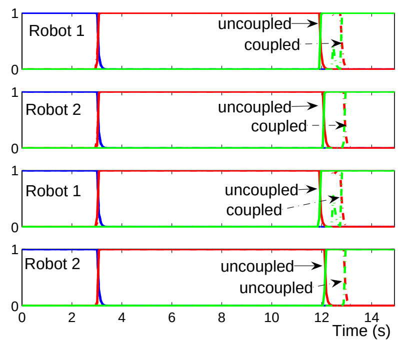

Fig. 3. Top and bottom panels illustrate u neural variables of x and y coordinate dynamical systems of both robots.

This discrete analogue of frequency locking is illustrated in Fig. 4. Note that synchronization only exists when both dynamical systems exhibit periodic motion.

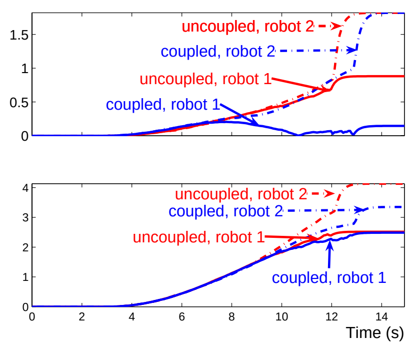

Fig. 4. Dynamical variables for robot 1 and 2 in x and y coordinate dynamical systems.

Coupling two such dynamical systems removes the need to compute exactly identical movement times for two robot movements that must be temporally coordinated. Even if there is a discrepancy in the movement time programmed by the parameter, ω, of the Hopf dynamics (which corresponds to larger MTs due to complex environment configurations), coupling generates identical effective movement times.

One interesting aspect is that since the velocities applied to the robots are different depending if there is coupling or not, this results in slightly different qualitative paths followed by the robot.

4. CONCLUSION/OUTLOOK

In this article, an attractor based dynamics autonomously generated temporally discrete and coordinated movements. The task was to temporally coordinate the timed movements of two low-level vehicles, which must navigate in a simulated non-structured environment while being capable of reaching a target within a certain time independently of the environment configuration. Movement termination was entirely sensor driven and autonomous sequence generation was stably adapted to changing unreliable simulated visual sensory information. We applied autonomous differential equations to formulate two integrated dynamical architectures which act out at the heading direction and driving speed levels of each robot. Each robot velocity is controlled by a dynamical systems architecture based on previous work (Santos, 2004), which generates timed trajectories. Temporal coordination of the two robots is enabled through the coupling among these architectures.

Results enable to positively answer to the two questions addressed in the introduction. The former asked if synchronization among two vehicles can be achieved when we apply temporal coordination among dofs. Results illustrate the dynamic architecture robustness and show that such a coupling tends to synchronize movement in the two robots, a tendency captured in terms of relative timing of robots movements. The later question asked if the applied approach provides a theoretically based way of tuning the movement parameters such that it is possible to account for relationships among these. Results show that the coupled dynamics enable synchronization of the robots providing an independence relatively to the specification of their individual movement parameters, such as movement time, movement extent, etc. This synchronization reduces computational requirements for determining identical movement parameters across robots. From the view point of engineering applications, the inherent advantages are huge, since the control system is released from the task of recalculating the movement parameters of the different components.

Future work will mainly address how to extend the described model to achieve more complex behavior for systems with several dofs. We will address the approach extension to robust locomotion generation and movement controllers for robots as this framework finds a great number of applications in service tasks and seems ideal to achieve intelligent and more human like prostheses.

REFERENCES

Bicho, E., P. Mallet and G. Schöner (2000). Target representation on an autonomous vehicle with low-level sensors. *The Int. Journal of Robotics Research* **19**(5), 424–447.

Bradski, G. (1998). Computer vision face tracking as a component of a perceptual user interface. Applications of Computer Vision. Princeton. pp. 214–219.

Buchli, J. and A. Ijspeert (2004). Distributed central pattern generator model for robotics application based on phase sensitivity analysis. In: *BioADIT*. Vol. 3141. pp. 333–349.

Clark, M, G Anderson and R Skinner (2000). Coupled oscillator control of autonomous mobile robots. *Autonomous Robots* **9**, 189–198.

Raibert, M H (1986). *Legged robots that balance*. MIT Press. Cambridge, Massachusetts.

Santos, C (2004). Generating timed trajectories for an autonomous vehicle: A non-linear dynamical systems approach. In: *IEEE Int. Conf. On Robotics and Automation*. IEEE. New Orleans. pp. 3741–3746.

Schaal, S, S Kotosaka and D Sternad (2000). Nonlinear dynamical systems as movement primitives. In: *IEEE Int. Conf. On Humanoid Robotics*. IEEE, Cambridge, MA.

Schöner, G (1994). Dynamic theory of action- perception patterns: The time-before-contact-paradigm. *Human Mov. Science* **3**, 415–439.

Schöner, G. and C. Santos (2001). Control of movement time and sequential action through attractor dynamics: A simulation study demonstrating object interception and coordination. In: *9th Intelligent Symp. On Intelligent Robotic Systems*. Toulouse, France.

Schöner, G. and M. Dose (1992). A dynamical systems approach to task-level system integration used to plan and control autonomous vehicle motion. *Robotics and Autonomous Systems* **10**, 253–267.

Steinhage, A. and G. Schöner (1998). Dynamical systems for the behavioral organization of autonomous robot navigation. In: *Spie-Intelligent Sys. Manufactors*. pp. 169–180.

Williamson, M. (1998). Rhythmic robot arm control using oscillators. In: *IEEE Int. Conf. On Int. Robots and Systems*.