Neural Control of a Three-Legged Reconfigurable Robot With Omnidirectional Wheels

P. Manoonpong* and F. Wörgötter

Bernstein Center for Computational Neuroscience,
The Third Institute of Physics, University of Göttingen,
Göttingen 37077, Germany
**E-mail: poramate@physik3.gwdg.de*

P. Laksanacharoen

Mechanical and Aerospace Engineering Department,
King Mongkut's University of Technology North Bangkok,
Bangkok 10800, Thailand
E-mail: stl@kmutnb.ac.th

In this article, we present neural control of a three-legged reconfigurable robot with omnidirectional wheels. It is systematically synthesized based on a modular structure such that the neuromodules are small and their structure-function relationship can be understood. The resulting network consists of four main modules. A so-called minimal recurrent control (MRC) module is for sensory signal processing and state memorization. It directly drives the motion of two front wheels while a rear wheel is indirectly controlled through a velocity regulating network (VRN) module. In parallel, a simple neural oscillator network module serves as a central pattern generator (CPG) producing basic rhythmic signals for sidestepping where stepping directions are controlled by a phase switching network (PSN) module. The combination of these neuromodules generates various locomotion patterns. Applying sensor inputs to the neural controller enables the robot to avoid obstacles as well as a corner. The presented neuromodules are developed and firstly tested using a physical simulation environment, and then finally transferred to the real robot.

Keywords: Neural networks, Mobile robot control, Autonomous robots, Obstacle avoidance.

1. Introduction

During the last years, we have developed a physical three-legged reconfigurable robot with omnidirectional wheels.[1] It combines the concept of using legs, wheels, and rolling sphere for multi-locomotion modes. Due to its closed-spherical shape, it can roll passively where this rolling motion can minimize the friction and lead to energy efficiency.[2] For autonomous exploration, it will transform into two inter-connected hemispheres with extending its three legs for locomotion using wheels. If the wheels are broken, it could use the legs for further locomotion. To the best of our knowledge, this type of robot, which combines legs, wheels, and a rolling sphere for multi-modal locomotion, so far has not been developed by others. In general, there are several leg-wheel hybrid robots but without rolling sphere[3,4] while there are spherical rolling robots but without legs and wheels.[2,5]

Continuing the development of our robot system, this article presents neural control of the robot for the generation of active locomotion using wheels and legs as well as controlling a reactive obstacle avoidance behavior. Neural control exhibits dynamical features (e.g., periodic and hysteresis behavior) which are here exploited for the locomotion generation and robot behavior control. This neural network control also has a modular structure consisting of four modules. Due to its modularity, the controller is robust to changes of structures; i.e., modules can be completely removed leading to graceful degradation of the agent's functionality while as a whole the system can still function partially. We believe that our neural modules can be important components for locomotion generation in other complex robotic systems or they can serve as useful modules for other module-based neural control applications.

2. Three-Legged Reconfigurable Robot With Omnidirectional Wheels

The robot consists of three legs, omnidirectional wheels, and a body with two (hemi)spherical shells (Fig. 1). It has a total of eight DC servo motors. There are two motors at each leg where each of them moves the leg up and down and drives the wheel. There are two motors at the hinge (middle) joint for a transformation process. The process will allow one side hemisphere to open up another side of hemisphere at a time.

The robot is generally designed based on the concept of a spherical form where its three identical legs with the wheels are kept inside its shells (body) in order to perform passive rolling motion. This form (called dormant mode, Fig. 1(a)) provides the compact shape of the robot. For active locomotion, it will transform into two hemispherical shells where the wheeled legs are projected out of the shells (called transformed mode, Fig. 1(b)). To assure the proper positioning that the robot is allowed to split into two hemispheres, we use the data read from an accelerometer at the onset of an expansion

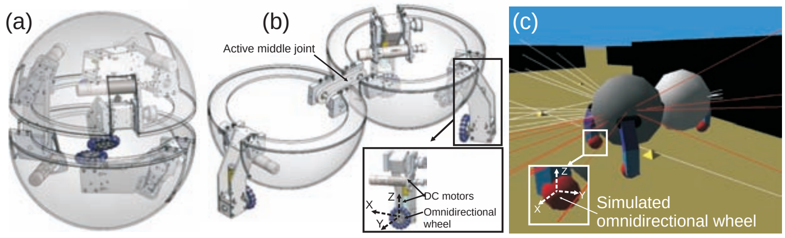

Fig. 1. (a), (b) The physical robot in dormant and transformed modes, respectively. (c) Simulated robot. To simulate the omnidirectional wheels of the robot, we set friction coefficients for two orthogonal directions (x- and y-axes) of each wheel independently. As a consequence, the wheel shows the unique property of rolling freely in the direction of its axis (i.e., freely rotating around x-axis), while operating as a normal wheel in the direction perpendicular to its axis (i.e., actively rotating around y-axis).

sequence.[1] The robot also has two infrared (IR) sensors installed at its front to detect obstacles. We use the physics simulation environment called Yet Another Robot Simulator (YARS)[6,7] to simulate the robot (Fig. 1(c)). The simulated robot is qualitatively consistent with the real one in the aspect of geometry, dimensions, mass distribution, motor torque, and sensors while frictional coefficient between the robot and ground is roughly estimated. Note that we simulate spherical shells for body parts instead of hemispherical shells since YARS has not yet provided a hemispherical geometry. However, the mass distribution and the total weight of the simulated and real robots are qualitatively consistent.

3. Neural Control for Locomotion and Obstacle Avoidance

Neural control is based on a modular structure (Fig. 2). It consists of four main modules: a minimal recurrent control network (MRC), a velocity regulating network (VRN), a neural oscillator network (abbreviated CPG, see below), and a phase switching network (PSN). The neural modules have been developed and applied to four-, six- and eight-legged robots as well as two wheeled robots in part.[6–8] Here, we for the first time combine them for wheeled and legged locomotion and a reactive obstacle avoidance behavior of a three-legged reconfigurable robot with omnidirectional wheels. We only discuss their functions used for the application here since the details of the neural modules together with the setup of their weights have already been presented in previous studies.[6–8] All neurons of the controller (i.e., hidden and motor neurons (Fig. 2)) are modelled as non-spiking neurons. Their activity develops according to $a_i(t+1) = \sum_{j=1}^{n} w_{ij}\, o_j(t) + b_i;\ i = 1, \ldots, n$

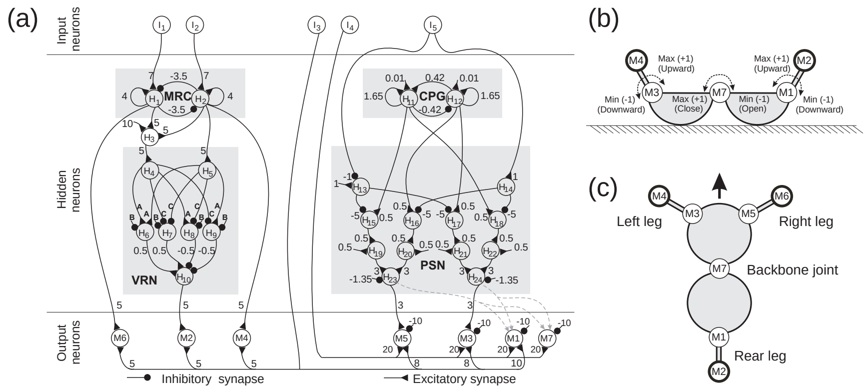

Fig. 2. (a) Neural control of the three-legged reconfigurable robot with omnidirectional wheels. There are three different neuron groups: input, hidden, and output. Input neurons I receive sensory signals controlling robot behavior ($I_{1,2,3,4,5}$). Hidden neurons H are divided into four subgroups or modules (MRC, VRN, CPG, and PSN) having different functionalities (see text for details). Output neurons are described as motor neurons ($M_{1,...,7}$) where $M_{1,3,5}$ are for controlling leg joints, $M_{2,4,6}$ are for controlling wheels, and M_7 is for controlling the body joint. All connection strengths together with bias terms are indicated by the small numbers except some parameters of the VRN given by $A = 1.7246$, $B = -2.48285$, $C = -1.7246$. Note that dashed arrows indicate additional synapses which can be added to obtain more locomotion behaviors. (b) The movements of the leg joints and the body joint. (c) The location of the motor neurons on the robot.

where n denotes the number of units, b_i represents a fixed internal bias term of neuron i, a_i their activity, w_{ij} the synaptic strength of the connection from neuron j to neuron i. The neuron output o_i is given by a hyperbolic tangent (tanh) transfer function $o_i = \tanh(a_i) = \frac{2}{1+e^{-2a_i}} - 1$. Input neurons $I_{1,2,3,4,5}$ are here configured as linear buffers ($a_i = o_i$).

The MRC module has been originally evolved through the evolutionary algorithm ENS[3] for generating obstacle avoidance behavior of a miniature Khepera robot with two wheels.[6] Due to its recurrent connections, the MRC exhibits hysteresis effects which allow an agent to keep on doing a task till the task is completed even if the stimulus has decayed or is removed. The hysteresis phenomena have already been discussed as models for short-term memory.[9] Without such memory, the agent might switch between tasks reactively without completing any of them and, thus fails to complete tasks. Here, we apply the MRC to directly drive the two front wheels (M_4, M_6) of our robot and exploit its hysteresis effects for controlling obstacle avoidance behavior and filtering sensory noise. We use two IR sensor signals $IR_{1,2}$ for obstacle detection at the front of the robot. They are transmitted to the inputs $I_{1,2}$ projecting to the MRC. I_1 corresponds to the left IR sensor signal and I_2 to that of the right one. Applying the output signals of H_1 and H_2 directly to their target motor neurons M_6, M_4 and indirectly to the motor neuron M_2 (Fig. 2) via the VRN module enables the robot to autonomously change its motion. For instance, the robot changes from moving forward to turning left when there are obstacles on the right, and vice versa. This way, it can avoid obstacles and escape from corners as well as deadlock situations.

The VRN module[7] basically changes the rotational direction of M_2 with respect to turning motion. The network approximately works as a multiplication operator. It was constructed as a feedforward network with two input, four hidden, and one output neurons. It was trained by using the backpropagation algorithm. The neuron H_3 is added to combine both outputs of the MRC. The output of H_3 projects to the input neuron H_4 of the VRN. Another input neuron H_5 of the VRN receives its input from the neuron H_2 of the MRC. By doing so, the VRN drives the rear wheel M_2 with low ≈ -1 activation to rotate clockwise leading to a right turn when there are obstacles on the left (i.e., the output of H_1 of the MRC is $\approx +1$ while the output of H_2 of the MRC is ≈ -1) and vice versa. In case no obstacle is detected (i.e., the outputs of $H_{1,2}$ are ≈ -1), the rear wheel will be inactive (zero activation) leading to forward motion.

The CPG module[7] generates basic rhythmic signals which control leg movements (M_3, M_5) in directly through the PSN module. The leg movements result in sidestepping. The CPG is realized by using the dynamics of a simple 2-neuron network with full connectivity and biases. Its weights were adjusted such that the network generates periodic attractors.[7] The network with the resulting weights produces rhythmic outputs that differ in phase by $\pi/2$ with a frequency of approximately 0.8 Hz. According to the robot configuration, making the front legs moves out of phase to each other by $\pi/2$, we obtain sidestepping.

The PSN module[8] is used to steer the sidestepping directions (i.e., lateral motions to the left and right). The network basically switches the phase of the two rhythmic signals originally coming from the CPG to lead or lag behind each other by $\pi/2$ in phase when the input I_5 is changed from 0 to 1 and vice versa. By applying this network property, the movements of the left and right legs will be reversed corresponding to the modification of I_5. Consequently, the robot will change its sidestepping directions from the right to the left and vice versa. The PSN is a hand-designed feedforward network consisting of four hierarchical layers with 12 neurons. The synaptic weights and bias terms of the network were determined in a way that they do not change the periodic form of its input signals and keep the amplitude of the signals as high as possible.

4. Experiments and Results

The first experiment presents locomotion behaviors of the robot using legs and wheels in the physics simulator (YARS). In this case, all input neurons ($I_{1,2,3,4,5}$, Fig. 2) were set manually to clearly observe the robot locomotion behaviors. The inputs $I_{1,2}$ were set to -1 and the input I_4 was set to 0 (i.e., the transformed mode with forward motion). We let the robot move over flat terrain and continuously changed inputs $I_{3,5}$ to investigate its basic locomotion. By simply controlling the input I_3, the robot changes its locomotion from using wheels to legs and vice versa. During legged locomotion, changing the input parameter I_5 from zero to one leads to sidestepping to the left and changing I_5 back to zero leads to sidestepping to the right. The input parameters and motor signals during the experiment are shown in Fig. 3. The video clip of this experiment showing forward motion using wheels and sidestepping using legs can be found at http://www.manoonpong.com/CLAWAR2013/S1.wmv.

The second experiment shows the performance of the controller implemented on the real robot. Figure 4(a) shows transformation behavior where the robot transformed from the dormant mode (Fig. 1(a)) to the transformed mode (Fig. 1(b)). In this case, the input I_4 was set from 1 to 0 while the other inputs were set to 0. Note that the expansion is automatically stopped as soon as the legs reach a desired position determined by the potentiometers of the leg joints. After this transformation, the robot will be able to locomote using wheels or legs. Figure 4(b) shows wheeled locomotion with a reactive obstacle avoidance behavior. For this scenario, the inputs $I_{1,2}$ received IR sensory signals while the inputs $I_{3,4,5}$ were set to 0. The robot moved toward a corner and then autonomously turned left since its IR sensor detected the right wall. Due to the hysteresis effects of the MRC, the effects allow the robot to keep on turning for longer than the stimulus itself (i.e., IR signal). This way, the robot performed a large turning angle; thereby easily avoiding the corner. In contrast, using finite state control or classical Braitenberg control[10] without state memory the robot needs to turn several times in order to avoid obstacles or it sometimes gets stuck. Figure 4(c) exemplifies legged locomotion (i.e., sidestepping to the right). The robot performed rhythmic leg movements where the inputs were set as $I_{1,2} = -1$, $I_3 = 1$, $I_{4,5} = 0$. Due to mechanical problems of this first prototype robot (i.e., backlash and slip of the leg driving mechanisms),

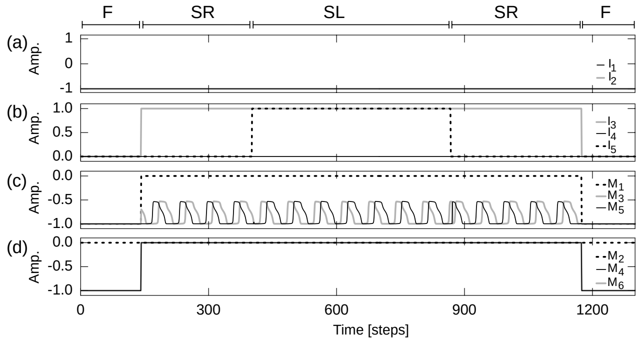

Fig. 3. (a) Inputs $I_{1,2}$ were set to -1 at all times resulting in only forward motion. (b) Inputs $I_{3,4,5}$. I_3 was used to switch between wheeled ($I_3 = 0$) and legged ($I_3 = 1$) locomotion. I_4 was here set to 0 in order to keep the robot in the transformed mode. Setting I_4 to 1 leads to the dormant mode. I_5 is used to steer the sidestepping directions. Setting I_5 to 0 leads to the sidestepping to the right SR and setting it to 1 leads to the left SL. (c) Motor signals at the leg joints ($M_{1,3,5}$). The motors $M_{3,5}$ show the periodic signals when the legged locomotion mode was activated. One can observe that when the robot steps sideways to its right the periodic signal of M_5 leads the one of M_3 by $\pi/2$ in phase and vice versa when the robot steps sideways to its left. (d) Motor signals at the wheels ($M_{2,4,6}$). Low ≈ -1 activation drives the wheels such that the robot moves forward F while zero activation means the wheels roll freely. Backward motion using the wheels can be achieved by setting $I_{1,2}$ to 1 and $I_{3,4,5}$ to 0.

its legs cannot follow the motor commands all the times as expected. As a result, sidestepping using its legs cannot be effectively performed.

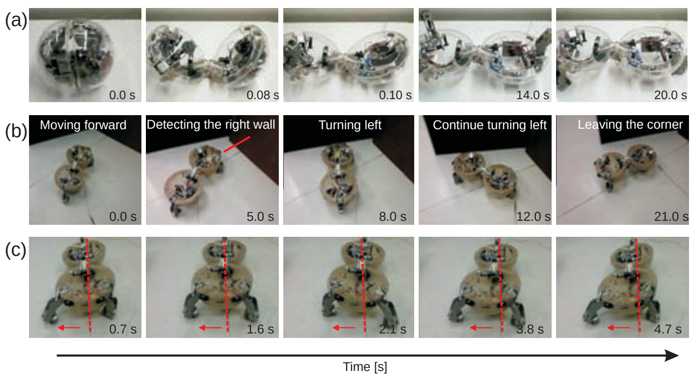

Fig. 4. (a) Transformation from the dormant mode to the transformed mode. (b) Obstacle avoidance behavior using wheels. (c) Sidestepping to the right using legs. The video clip of the tests including passive rolling motion can be see at http://www.manoonpong.com/CLAWAR2013/S2.wmv.

5. Conclusion

We presented neural control of a leg-wheel robot having two (hemi) spherical body shells and three legs with omnidirectional wheels. The controller was designed as a modular structure composed of four modules (MRC, VRN, CPG, and PSN). The MRC and CPG modules were developed by realizing dynamical properties of recurrent neural networks while the VRN and PSN modules were developed as feedforward networks. This neural controller generates active locomotion behaviors using wheels (i.e., forward, backward, turn left/right) and legs (i.e., sidestepping) as well as a reactive obstacle avoidance behavior. The behaviors were activated through the five inputs of the controller. Besides the active locomotion behaviors, the robot can perform passive rolling using its closed-spherical body.

Our next step will be the improvement of the leg driving mechanisms to obtain a better legged locomotion. In addition to this, we will use proprioceptive sensors (i.e., rotational sensors of wheels and joint angle sensors of leg joints) for damage detection and apply neural learning[11] to find behaviorally useful motor responses after damage.

Acknowledgments: This research was supported by Emmy Noether grant MA4464/3-1, BCCN grant 01GQ1005A (project D1), and the Office of the Higher Education Commission, Thailand.

References

1. N. Chadil, M. Phadoognsidhi, K. Suwannasit, P. Manoonpong and P. Laksanacharoen, A reconfigurable spherical robot, in *Proc. IEEE Int. on Robotics and Automation (ICRA)*, 2011.
2. Y. Kim, S. Ahn and Y. Lee, Kisbot: new spherical robot with arms, in *Proc. of Int. on Robotics, control and manufacturing technology (WSEAS)*, 2010.
3. M. Eich, F. Grimminger, S. Bosse, D. Spenneberg and F. Kirchner, Asguard: A hybrid legged wheel security and sar-robot using bio-inspired locomotion for rough terrain, in *Proc. of the IARP/EURON Workshop*, 2008.
4. S. Nakajima and E. Nakano, *J. Robot Mechatronics* **20**, 912 (2008).
5. R. Armour and J. Vincent, *J. Bionic Eng.* **3**, 195 (2006).
6. M. Hülse, S. Wischmann and F. Pasemann, *Connect. Sci.* **16**, 249 (2004).
7. P. Manoonpong, *Neural preprocessing and control of reactive walking machines: Towards versatile artificial perceptionaction systems* (Springer, 2007).
8. P. Manoonpong, F. Pasemann and F. Wörgötter, *Robot. Auton. Syst.* **56**, 265 (2008).
9. E. Harth, T. Csermely, B. Beek and R. Lindsay, *J. Theor. Biol.* **26**, 93 (1970).
10. V. Braitenberg, *Vehicles: Experiments in synthetic psychology* (MIT, 1984).
11. P. Manoonpong, C. Kolodziejski, F. Wörgötter and J. Morimoto, *Adv. Complex Syst.* (2013).