Neural Based Obstacle Avoidance with CPG Controlled Hexapod Walking Robot

Petr Čížek and Pavel Milička and Jan Faigl
Czech Technical University
Faculty of Electrical Engineering
Technicka 2, 166 27, Prague, Czech Republic
Email: petr.cizek@fel.cvut.cz

***Abstract*—In this work, we are proposing a collision avoidance system for a hexapod crawling robot based on the detection of intercepting objects using the Lobula giant movement detector (LGMD) connected directly to the locomotion control unit based on the Central pattern generator (CPG). We have designed and experimentally verified the proposed approach that maps the output of the LGMD directly on the locomotion control parameters of the CPG. The results of the experimental verification of the system with real mobile hexapod crawling robot support the feasibility of the proposed approach in collision avoidance scenarios.**

I. Introduction

Collision avoidance within the context of autonomous mobile robots has been a subject of studies ever since the mobile robots appear, because the ability to navigate from one place to another comes hand in hand with the need of the mobile system to interact with physical objects and entities along a robot's path. It is desirable to avoid contact with fixed or moving objects while en route because such a contact might have fatal consequences for the mission the robot is performing or even for the robotic platform itself.

In this paper, we concern a biologically inspired neural-based locomotion control to develop a collision avoidance system for a legged walking robot. In particular, we utilize a Central pattern generator (CPG) for the locomotion control together with the vision-based collision avoidance approach for the interception detection using the Lobula giant movement detector (LGMD) to enable collision avoidance behavior of a hexapod walking robot, thus enhance the capabilities of mobile robots and their autonomous operations in an unknown terrain. Moreover, the proposed combination of biologically inspired approaches is conceptually simple, easily deployable on computationally constrained robotic hardware, and it has already shown promising results when deployed on different robotic platforms [1], [2].

The overall structure of the proposed system is depicted in Fig. 1. The proposed solution builds on our previous work [3] on chaotic oscillator-based CPG for motion control of the hexapod walking robot. In [3], we propose a locomotion controller that can parametrize any given trajectory by adjusting only two input parameters, the turning radius $turn$ and the period of the oscillations p that in fact allows selecting a different type of the robot motion gait. In this work, we extend our previous work by visual feedback, which allows adjusting

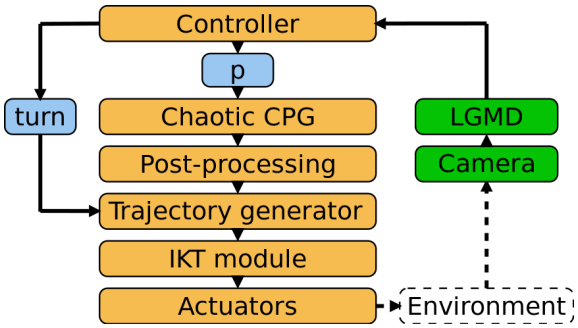

Fig. 1. Overview of the control system structure

the parameter $turn$ online as the robot is crawling through the environment, and thus prevent collisions with obstacles. The LGMD neural network processes the visual data and gets stimulated whenever there is an object interfering with the robot's path. A decision rule is applied to adapt the motion control parameter $turn$ to alter the robot behavior and avoid a collision. The proposed approach has been experimentally verified with a real walking robot while the specific discrete nature of the legged locomotion makes the task very different in comparison to wheeled [1] or flying [2] robots for which the LGMD has been already utilized. The main difference originate in the abrupt motions of the camera induced by the locomotion which negatively influence the output of the collision avoiding visual pathway.

The paper is organized as follows. Section II lists the most relevant approaches on the neural-based collision avoidance for mobile robots. Section III describes the individual building blocks of the proposed architecture in detail. Results on the experimental evaluation of the proposed system deployed on the real hexapod walking robot are detailed in Section IV. The concluding remarks and discussion are dedicated to Section V.

II. Related Work

Two basic building blocks can be identified in the structure of the proposed control system. The first one is the CPG-based locomotion controller for the hexapod walking robot based on the combination of neural based chaotic oscillator and post processor which shapes the output of the oscillator to be used for the direct control of individual robot limbs. The second

building block is the collision avoidance system based on the LGMD neural network. Since these building blocks have been pre-selected, a brief overview of the existing approaches is commented in this section to advocate a suitability and advantages of the selected approaches.

Biologically inspired strategies based on CPGs have already been utilized in a control of the legged locomotion. In general, the CPG is a neural network that produces patterned rhythmical outputs that are responsible for breathing, walking, and other repetitive processes in animals and insects [4], [5]. There are many ways of achieving patterned output [6], [7], [8], [9] which is either directly processed by neural networks with the motoneurons as the output layer [10], [11] or post-processed and transformed to the individual joint angles using inverse kinematics [12].

Most commonly used CPG implementations are based on non-linear oscillators (NLO), which are not strictly biologically based, but share many common characteristics with biophysical models. Among them the Matsuoka NLO [13], implementing the half center principle: extensor and flexor neurons inhibiting each other with an adaptation mechanism, is the prevalent one. The Matsuoka model was successfully simulated and implemented in hexapod [12], [14], quadruped [15], and biped walking [16]. Our CPG-based locomotion controller is based on an NLO proposed in [10] but uses different post-processing for shaping the output signal [3].

The most relevant neural-based collision avoidance systems include earlier work on LGMD published in [1], [17], and [2]. In [1], the authors describe the LGMD model for a collision avoidance and deploy the system on a wheeled robot moving in an arena. The behavior of the LGMD is studied for different speeds of the mobile robot. In the recent work [17], the LGMD is compared to the directional selective neurons (DSN) which are both to be found in the visual pathways of insects, in the ability to avoid collisions. The presented results show that the LGMD can be trained to outperform the DSN in the collision recognition ability. The LGMD collision avoidance has also been considered for the collision avoidance of a Blimp UAV in [2]. Note, all of the above mentioned approaches experimentally verify the collision avoidance with a real robot either in a closed arena where it is necessary to avoid collisions with walls or in a scenario where a static robot is supposed to detect an intercepting object. In this paper, we consider the LGMD on a different robot (hexapod walking robot) and in a different setup, the robot is requested to reach its goal location and avoid obstacles on its pathway.

Regarding our target scenario, the most relevant approach to the proposed solution has been presented in [18]. The authors use a bio-inspired collision avoidance approach based on the extraction of nearness information from the visual motion to detect obstacles and avoid collisions. The whole system allows a simulated hexapod robot to navigate cluttered environment while actively avoiding obstacles. In comparison to their work, we are using a different neural-based approach for detecting a possible collision. Moreover, we utilize a neural based locomotion controller. We also emphasize the real-world practical verification of our approach while the dynamics of the walking robot makes the whole task more complicated than the simulation framework used in [18].

III. Proposed Solution

The proposed solution of the reactive hexapod agent consists of two main parts. The first part is the hexapod locomotion control based on the chaotic oscillator [3], [10], [19] controlling the walking pattern and solving the kinematics. It allows to change the type of the motion gait and steer the robot motion according to the input signal defining the turning radius. The second part is the LGMD neural network, which is a biologically inspired system for avoiding approaching objects and triggering escape behavior [20], [21]. The main idea of the proposed approach is to use the LGMD for setting the hexapod control parameters, in particular, the turning radius of the robot.

Fig. 1 shows the overall structure of the proposed solution, where the orange blocks belong to the locomotion control system whereas the green blocks represent the visual pathway for the obstacle avoidance mechanism. The goal of the proposed solution is to close the feedback loop from controlling the actuators (orange blocks) with the observations of the environment (green blocks) to achieve an obstacle avoidance behavior. Therefore, the most important part of the proposed approach is to set parameters of the LGMD for the particular walking pattern that would allow the hexapod to safely avoid obstacles on its path. Along with the LGMD parameters, a suitable mapping function Φ has to be found, which transforms the LGMD output to the locomotion control parameters

$$\Phi : \mathcal{X} \rightarrow turn, \tag{1}$$

where $\mathcal{X}$ is the range of the vision system output and $turn$ is the locomotion control parameter determining the turning radius of the hexapod. The individual building blocks and the proposed solutions are detailed in the following sections.

A. Hexapod Locomotion Control

The locomotion control is based on a chaotic CPG consisting of two interconnected neurons with a control input computed solely based on the input period p [3]. The control input stabilizes a periodic orbit of p from the chaotic oscillation so the output is a discrete periodic signal. Period $p \in \{4, 6, 8, 12\}$ directly determines the resulting walking pattern (gait): tripod, ripple, tetrapod, and wave, respectively [22].

The output of the chaotic oscillator is shaped and post-processed in order to obtain signal usable for a trajectory generator and to determine the phase of individual legs, i.e., whether the leg is swinging or supporting the body. Fig. 2 visualizes the output of the chaotic CPG together with individual steps of the post-processing pipeline. The top plot shows the oscillations before stabilization (blue lines) and after stabilizing the periodic orbit (red line) where the dots are the individual states of the oscillator (for $p = 6$ there are 6 states) for the first $x_1(t)$ and second $x_2(t)$ neuron. For further processing, the difference of the second neuron output

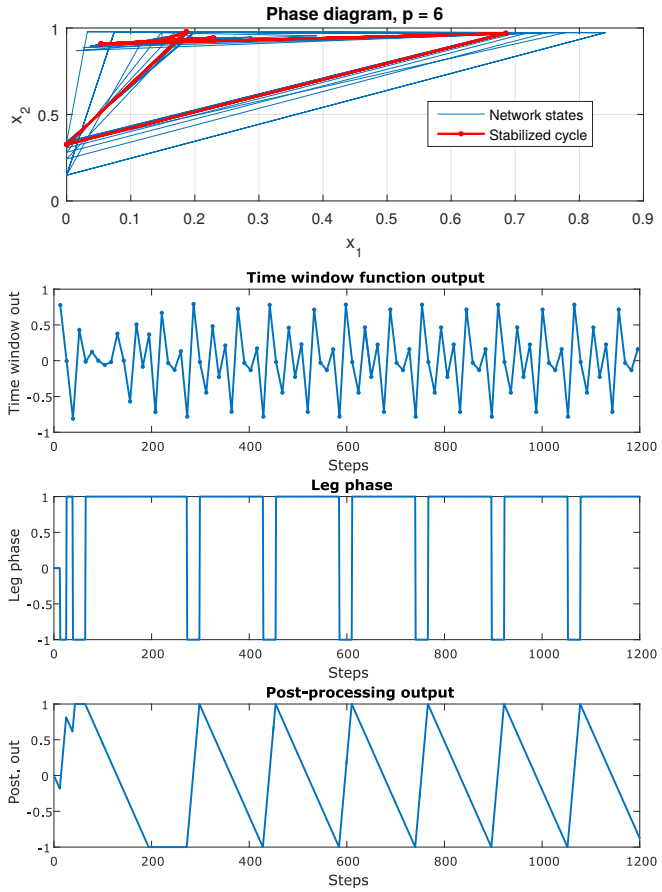

Fig. 2. CPG patterned output generation and post-processing

$x_2(t)$ and its delayed value $x_2(t-1)$ is passed through a time window function, which samples and holds the value only if $\mod(t, 13) = 0$ to achieve smoother leg movement. Then, the output is thresholded and a triangle wave alternating between -1 and 1 is produced, where the upslope (swing phase) is a constant and the downslope (support phase) depends on the period p. Based on the leg coordination rules [23], individual delays are applied to the triangular wave per each leg to produce the rhythmic pattern for each leg based on one CPG. In other words, there is only one CPG for the robot and by delaying and shaping its output a control signal is produced for each leg.

The result of the post-processing module is fed into a trajectory generator, which determines the position of foot-tips according to the input signal along with the parameter $turn$, which is obtained by the mapping function Φ. The $turn$ parameter is equal to the distance (in millimeters) from the robot center to the turning center on a line perpendicular to the heading of the robot connecting the default positions of the middle legs. The inputs of trajectory generator uniquely determine the foot-tip positions of each leg on the constructed arcs which are limited by the angle α. The value of α is computed from the distance of the furthest leg from the pivotal point established by $turn$ and the maximum step size y_{max}. The idea of the trajectory generator is visualized in Fig. 3.

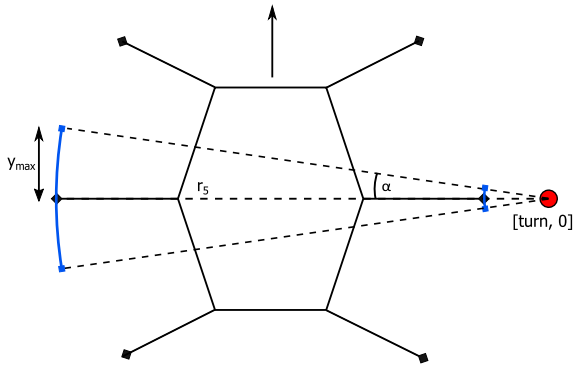

Fig. 3. Trajectory generation - the turning point denoted as red circle is given by $turn$ parameter. α is computed as the maximum angle given the turning radius and the maximum step size y_{max}.

The output of the trajectory generator is transformed into the joint space using the inverse kinematics module (IKT) and then sent to the robot actuators. The robot motion is performed by the gait type according to the period p, which moves the robot body forward at a particular speed defined by the gait type while the robot angular velocity is controlled by the $turn$ parameter, which is adjusted by the LGMD described in the following section.

B. Lobula Giant Movement Detector

The obstacle avoiding behavior is achieved by utilizing the LGMD neural network. The LGMD is a system of neural layers found in the visual pathways of insects, such as locusts, which responds selectively to objects approaching the animal on a collision course. The neural network is composed of four groups of cells: *Photoreceptive*, *Excitatory*, *Inhibitory*, and *Summation*; and two individual cells: *Feed-forward inhibitory* and *Lobula Giant Movement Detector*.

The *Photoreceptive layer* processes the sensory input from the camera. Its output is the difference between two successive frames grabbed from the camera computed as

$$P_f(x, y) = L_f(x, y) - L_{f-1}(x, y), \tag{2}$$

where f denotes the current frame and L is the intensity of the (x, y) pixel in grayscale. In principle, the *Photoreceptive layer* detects changes in intensity values and forms the input to another two layers – the *Inhibition layer* and *Summation layer*.

The response of the *Inhibition layer* is computed as

$$I_f(x, y) = \sum_{i=-n}^{n} \sum_{j=-n}^{n} P_{f-1}(x+i, y+j) w_I(i, j) \tag{3}$$
$$(i \neq j, \text{ if } i = 0),$$

where w_I are the inhibition weights set as

$$w_I = \begin{bmatrix} 0.125 & 0.250 & 0.125 \\ 0.250 & 0 & 0.250 \\ 0.125 & 0.250 & 0.125 \end{bmatrix}. \tag{4}$$

The *Inhibition layer* is essentially smoothing the *Photoreceptive layer* output values filtering those caused by noise or camera imperfections.

The response of the *Summation layer* is computed as

$$S_f(x,y) = \text{abs}(P_f(x,y)) - \text{abs}(I_f(x,y))W_I, \quad (5)$$

where W_I is the global inhibition weight.

Let S'_f be a matrix, where each value exceeding the threshold T_r is passed and any lower value is set to 0

$$S'_f(x,y) = \begin{cases} S_f(x,y) & \text{if } S_f(x,y) \geq T_r \\ 0 & \text{otherwise} \end{cases}. \quad (6)$$

Then, we can compute an excitation of the *LGMD* cell as

$$U_f = \sum_{x=1}^{k} \sum_{y=1}^{l} \text{abs}(S'_f(x,y)) \quad (7)$$

and finally, the *LGMD* cell output is

$$u_f = (1 + \exp^{-U_f n_{cell}^{-1}})^{-1}, \quad (8)$$

where n_{cell} is the total number of cells (the number of pixels). Note, the output of u_f is in the interval $u_f \in [0.5, 1]$.

Typically, the LGMD neural network contains *Feed-forward* cells which suppress the output of the *LGMD* cell in a case of camera movement. However, the experimental evaluation showed that the *Feed-forward* cells are not necessary and in several cases restrict the collision avoidance behavior. In principle, the LGMD neural network reacts on the lateral movement of significant vertical edges in the environment regardless their depth in the scene. In a case of camera pan the further edges might play a more prevalent role in the decision of the LGMD than the closer less significant ones. The purpose of the Feed forward cell is to suppress the output of the LGMD in such a case which is; however, not desirable when the robot moves continuously as in our case.

Our vision system consists of two separate LGMDs to detect the direction of interception, and thus be able to steer the robot in the opposite direction to achieve the desired obstacle avoiding behavior. The input image is split into two left and right parts with the overlapping center part. The output of the system is the difference between the left and right LGMD responses. The difference is mapped to $turn$ parameter by the proposed Φ mapping function. The particular function Φ has been designed experimentally and it has the form

$$\Phi(e) = \begin{cases} 100/e & \text{for } \text{abs}(e) \geq 0.2 \\ 10000 \cdot \text{sgn}(e) & \text{for } \text{abs}(e) < 0.2 \end{cases}, \quad (9)$$

where e is the difference of the LGMD outputs

$$e = u_f^{left} - u_f^{right}. \quad (10)$$

The feasibility of the proposed solution to provide a collision avoiding behavior has been experimentally verified using a real hexapod walking robot. The results of the performed experiments are reported in the next section.

IV. Experimental Evaluation

The proposed neural-based architecture for the collision avoiding behavior has been experimentally verified in a set of experiments. We are emphasizing the practical verification with a real walking robot to thoroughly test the proposed solution and provide insights on the achieved performance. Fig. 4 shows two test tracks used for the evaluation of the obstacle avoiding behavior.

Fig. 4. Experimental setup in the hallway and in the lab environment

The experimental evaluation has been considered with the PhantomX hexapod walking robot. The robot has six legs attached to the trunk that hosts the sensors. In particular, two cameras have been attached to the robot trunk. The Logitech C920 camera with the Field of view (FOV) 78° to provide the LGMD with the visual input and Asus Xtion Pro Live RGB-D camera to provide a reliable tracking and ground truth for the experiment based on the localization technique [24]. The image data fed into the LGMD neural network has been captured by the Logitech C920 camera, subsampled to the resolution of 176×144 pixels and finally divided into two parts overlapping in 10% of the image area.

The first experimental setup consists of 3 m length path in the lab environment with only one obstacle placed directly in front of the robot, whereas the second setup in a hallway consists of 3 obstacles as visualized in Fig. 4. The lab experiment tests the ability of the system to deal with more cluttered environment with a narrow passage only slightly exceeding the robot's outline. The hallway experiment has been conducted to verify that the robot can navigate between multiple obstacles.

During the experiments, the robot was started from approx. the same place with heading directly intercepting the first obstacle and was reactively guided by the proposed controller through the experimental test track. Note, the forward speed of the robot is given by the used gait (parameter p), thus only $turn$ parameter is adjusted during the traversal.

During the traversal, RGB-D data has been logged and later post-processed by the RGB-D Simultaneous localization and mapping (SLAM) algorithm [25] to recover the executed trajectory. We have used this approach instead of external localization to precisely evaluate the position of the obstacles with respect to the robot concerning its current FOV. Altogether four trials $s_{1..4}$ in the lab and five trials denoted as $t_{1..5}$ on hallway setup have been performed. The experimental results are detailed in the rest of this section.

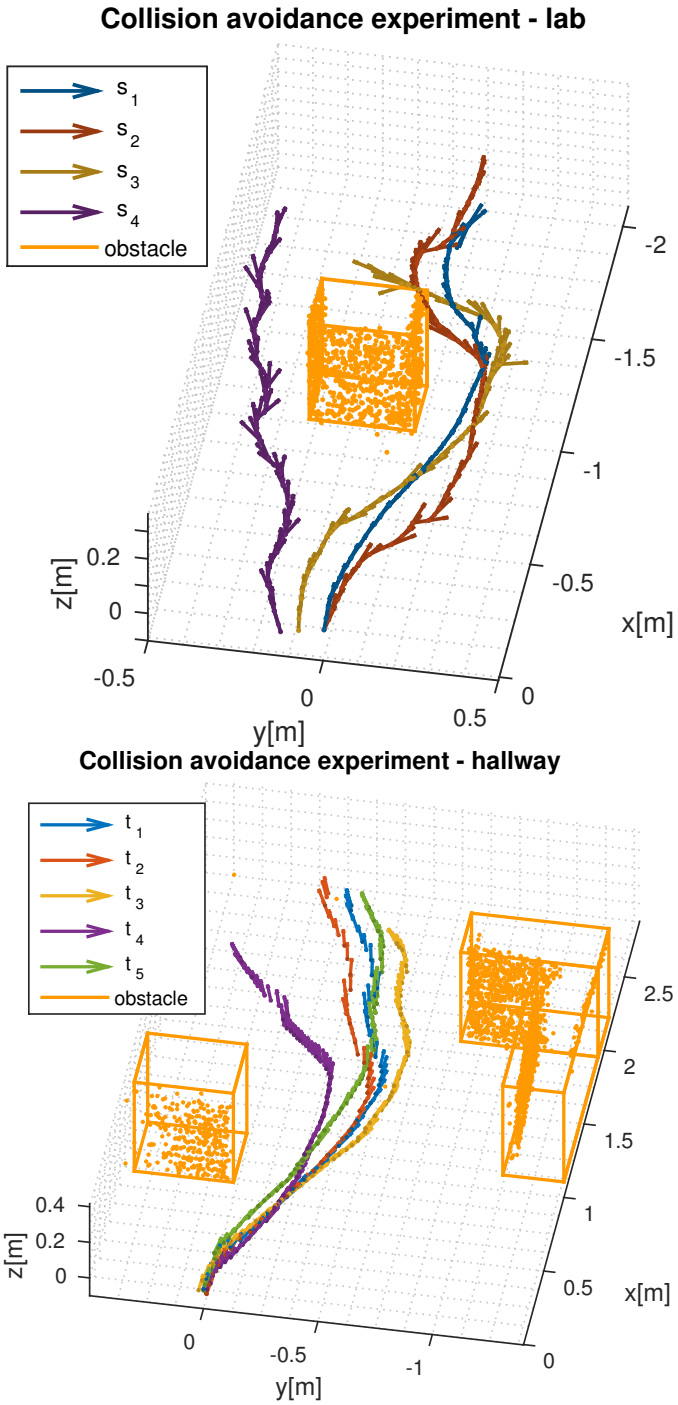

Fig. 5. Trajectories with obstacle and overlying point cloud recovered by the RGB-D SLAM

A. Results

Fig. 5 shows all the resulting trajectories together with an outline of the obstacles with a part of the overlying point cloud recovered by the RGB-D SLAM. As can be seen, in all nine cases, the robot was able to avoid the obstacles and reach the end of the path.

Fig. 6 and 7 show the individual trajectories s_2, s_3 s_4 and t_3, t_4, t_5 respectively from the aerial view with overlaid output of the LGMD which shows the strength of the turning at a given point. Note, for a better readability the graph includes the difference of the LGMD outputs e given by (10) and not the actual value of the Φ mapping function. Regarding the proposed Φ, a particular value of the $turn$ parameter is inversely proportional to the strength of the LGMD output, i.e., as the parameter $turn$ gets higher, the robot walks straight. A very high value of the $turn$ parameter means the robot is crawling along a curve with a high curvature, and thus it moves straight ahead. The direction of the LGMD output determines the direction in which the LGMD is trying to avoid the obstacle given the heading of the robot.

Fig. 8 shows a plot of distances to the obstacles along the path together with the LGMD output e. The distances to the obstacles which are currently in the field of view of the camera are taken into account. The vertical axis of the plot discriminates the direction to the obstacle given the heading of the robot. Although there are obstacles in a closer distance to the robot, only the ones that are currently in the field of view are influencing the output of the LGMD; hence, the heading of the robot. The phenomena can be observed in Fig. 9 which shows the images captured by the camera along the path.

B. Discussion of the Results

The results indicate that the proposed neural-based locomotion controller with the collision avoidance feedback provided by the LGMD neural network is feasible.

Concerning the laboratory test track, the robot was able to navigate the path without collisions; however, it can be seen that a more cluttered environment has a substantial effect on the reliability of the algorithm and smoothness of the motion. In a case of the hallway experiment, the main conclusions can be drawn from Fig. 7 and Fig. 8. It can be seen that the majority of the LGMD actions are in the opposite direction to the closest obstacle unlike in case of the lab test track. The reason behind this behavior is, that the LGMD neural network reacts on any moving vertical edge in the image. This can be seen in t_3, t_4, and t_5 sequences at about 80% of the trajectory which corresponds to Fig. 9f. During this part, the doors to the hallway forms the most prevalent edge in the environment which influences the turning of the mobile robot. Whenever the robot ceases to see the door, the most prevalent edge in the empty hallway is the jamb on the right side of the door at the end of the hallway. At that moment, the LGMD starts to push the robot to the left which is clearly visible in Fig. 8. However, that is the anticipated behavior of the LGMD-based collision avoidance, e.g., whenever the robot is in a closed arena, the influence of the closest obstacle gets inevitably the prevalent role in the direction designation. This phenomena causes the LGMD output of trajectories s_2, s_3, and s_4 less smooth with significant turns on the spot.

It also has to be noted that the output of the LGMD is considerably influenced by the movement of the robot. As has been said, we did not incorporate any suppressing mechanism to the output of the LGMD. It has shown that the legged

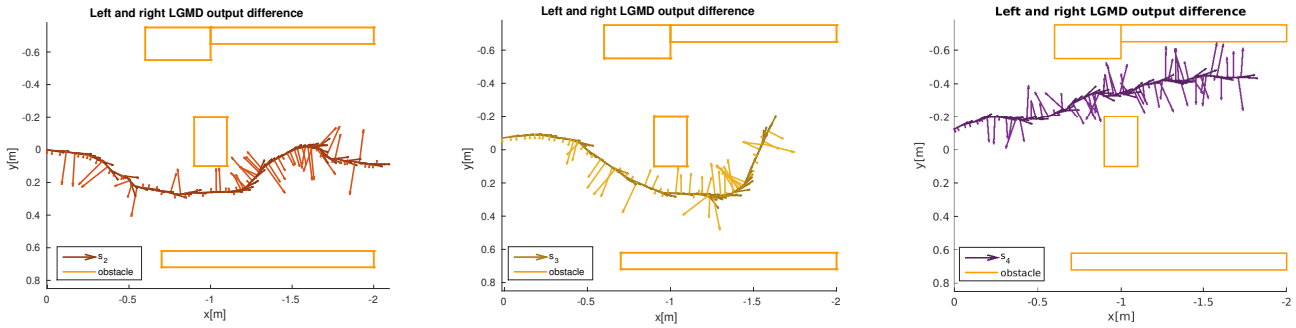

Fig. 6. Individual collision avoiding trajectories for experiments s_2, s_3 and s_4 with the overlayed LGMD output e

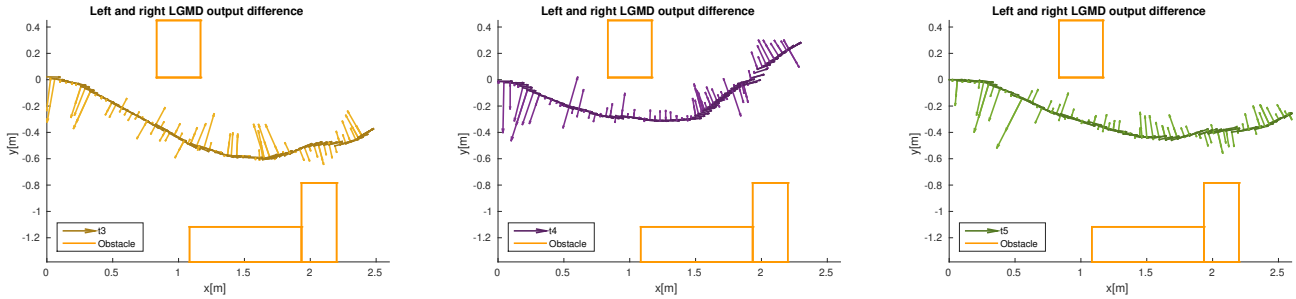

Fig. 7. Individual collision avoiding trajectories for experiments t_3, t_4 and t_5 with the overlayed LGMD output e

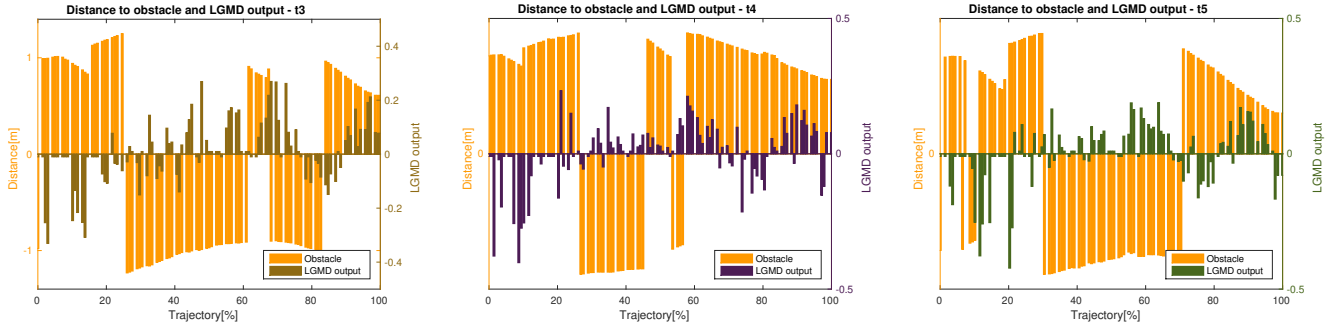

Fig. 8. Graph of the distance to a nearest obstacle in the FOV overlayed with the LGMD output e

Fig. 9. Images taken by the robot during the traversing the t_3 sequence

locomotion does not influence the output of the LGMD in a markable way, therefore in less cluttered environments, the suppressing mechanism is not necessary. On the other hand, we tested the collision avoiding behavior also in heavily cluttered environment where the robot was unable to avoid obstacles in a continuous motion due to a lot of stimuli from the distinctive edges in far distance from the robot because of the locomotion.

It is also worth noting that we tested the LGMD collision avoidance also with a fish-eye lens camera with FOV of 178° which also did not work well because of a fast movement of the edges the robot is passing by. A solution might be to learn the weights of the individual neurons in LGMD layers using reinforcement learning or evolutionary technique which we consider as a subject of our future work. Such an approach might neglect the influence of distinctive far vertical edges which mostly impacts the collision avoiding behavior in a cluttered environment. Moreover, as the reaction of the robot is based solely on the current observation of the environment it is necessary to incorporate a memory time windowing mechanism which would prevent the robot hitting obstacles from the side that has successfully avoided earlier.

V. Conclusion

In this paper, we propose a neural-based locomotion controller with integrated obstacle avoiding behavior based on the visual feedback. We have implemented and experimentally verified the proposed approach with a real walking robot in a collision avoidance scenario. The experimental results support feasibility of the proposed approach. We have experimentally verified that the collision avoidance behavior is triggered correctly by the closest obstacles when the environment is less cluttered. Whenever the environment is heavily cluttered, the LGMD output gets more influenced by distant objects and the image motion induced by the locomotion. In that case, it is necessary to adapt the controller to incorporate mechanisms for the LGMD output suppressing and weighting. Addressing this issue is considered for future work.

Acknowledgments – The presented work was supported by the Czech Science Foundation (GAČR) under research project No. 15-09600Y.

References

[1] M. Blanchard, F. Rind, and P. F. M. J. Verschure, "Collision avoidance using a model of the locust LGMD neuron," *Robotics and Autonomous Systems*, vol. 30, no. 1–2, pp. 17–38, 2000.

[2] S. B. i Badia, P. Pyk, and P. F. M. J. Verschure, "A Biologically Based Flight Control System for a Blimp-based UAV," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2005, pp. 3053–3059.

[3] P. Milička, P. Čížek, and J. Faigl, "On chaotic oscillator-based central pattern generator for motion control of hexapod walking robot," in *ITAT*, vol. 1649, 2016, pp. 131–137.

[4] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: a review," *Neural Networks*, vol. 21, no. 4, pp. 642–653, 2008.

[5] T. G. Brown, "On the nature of the fundamental activity of the nervous centres; together with an analysis of the conditioning of rhythmic activity in progression, and a theory of the evolution of function in the nervous system," *The Journal of Physiology*, vol. 48, no. 1, pp. 18–46, 1914.

[6] E. Marder and D. Bucher, "Central pattern generators and the control of rhythmic movements," *Current biology*, vol. 11, no. 23, pp. R986–R996, 2001.

[7] J. Yu, M. Tan, J. Chen, and J. Zhang, "A survey on cpg-inspired control models and system implementation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 3, pp. 441–456, 2014.

[8] R. Haschke, "Bifurcations in discrete-time neural networks: controlling complex network behaviour with inputs," Ph.D. dissertation, Bielefield University, 2003.

[9] F. Pasemann, "Complex dynamics and the structure of small neural networks," *Network: Computation in neural systems*, vol. 13, no. 2, pp. 195–216, 2002.

[10] S. Steingrube, M. Timme, F. Wörgötter, and P. Manoonpong, "Self-organized adaptation of a simple neural circuit enables complex robot behaviour," *Nature physics*, vol. 6, no. 3, pp. 224–230, 2010.

[11] T. Geng, B. Porr, and F. Wörgötter, "Fast Biped Walking with a Sensor-driven Neuronal Controller and Real-time Online Learning," *International Journal of Robotics Research*, vol. 25, no. 3, pp. 243–259, 2006.

[12] W. Chen, G. Ren, J. Wang, and D. Liu, "An adaptive locomotion controller for a hexapod robot: CPG, kinematics and force feedback," *Science China Information Sciences*, vol. 57, no. 11, pp. 1–18, 2014.

[13] K. Matsuoka, "Sustained oscillations generated by mutually inhibiting neurons with adaptation," *Biological cybernetics*, vol. 52, no. 6, pp. 367–376, 1985.

[14] L. Xu, W. Liu, Z. Wang, and W. Xu, "Gait planning method of a hexapod robot based on the central pattern generators: Simulation and experiment," in *IEEE International Conference on Robotics and Biomimetics*, 2013, pp. 698–703.

[15] T. Ishii, S. Masakado, and K. Ishii, "Locomotion of a quadruped robot using cpg," in *IEEE International Joint Conference on Neural Networks (IJCNN)*, vol. 4, 2004, pp. 3179–3184.

[16] G. L. Liu, M. K. Habib, K. Watanabe, and K. Izumi, "Central pattern generators based on matsuoka oscillators for the locomotion of biped robots," *Artificial Life and Robotics*, vol. 12, no. 1-2, pp. 264–269, 2008.

[17] S. Yue and F. C. Rind, "Redundant neural vision systems – competing for collision recognition roles," *IEEE Transactions on Autonomous Mental Development*, vol. 5, no. 2, pp. 173–186, 2013.

[18] H. G. Meyer, O. J. N. Bertrand, J. Paskarbeit, J. P. Lindemann, A. Schneider, and M. Egelhaaf, "A bio-inspired model for visual collision avoidance on a hexapod walking robot," in *5th International Conference on Biomimetic and Biohybrid Systems*, 2016, pp. 167–178.

[19] W. Chen, G. Ren, J. Wang, and D. Liu, "An adaptive locomotion controller for a hexapod robot," *Science China Information Sciences*, vol. vol. 57, no. issue 11, pp. 1–18, 2014.

[20] S. Yue and F. C. Rind, "Collision detection in complex dynamic scenes using an LGMD-based visual neural network with feature enhancement," *IEEE Transactions on Neural Networks*, vol. 17, no. 3, pp. 705–716, 2006.

[21] S. Yue and F. Rind, "Near range path navigation using LGMD visual neural networks," in *2nd IEEE International Conference on Computer Science and Information Technology*, 2009, pp. 105–109.

[22] N. Porcino, "Hexapod gait control by a neural network," in *International Joint Conference on Neural Networks*. IEEE, 1990, pp. 189–194.

[23] D. M. Wilson, "Insect walking," *Annual review of entomology*, vol. 11, no. 1, pp. 103–122, 1966.

[24] P. Čížek and J. Faigl, "On localization and mapping with RGB-D sensor and hexapod walking robot in rough terrains," in *Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics*, 2016, pp. 2273–2278.

[25] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the RGB-D SLAM system," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 1691–1696.