Tactile Sensing with Servo Drives Feedback only for Blind Hexapod Walking Robot

Jakub Mrva and Jan Faigl

Abstract— **In this paper, we address the problem of traversing rough terrains with hexapod walking robots. Although there can be found several approaches to deal with the terrain complexity, the proposed approach is strictly focused on a minimalist and cheap sensor equipment without additional inertial and exteroceptive sensors. The main idea of the proposed approach is to consider only the feedback from the intelligent servo drives to detect the contact point of a leg with the surface. During the leg motion, a relation of the joint torque and difference of the current and required joint positions is utilized to emulate a dedicated tactile sensor and thus the only equipment needed are the robot actuators. The proposed approach has been experimentally verified in a series of scenarios where a regular motion gait does not allow the robot to traverse the terrain while the proposed detection method enables a smooth motion of the technically blind robot in rough terrains of various difficulty.**

I. INTRODUCTION

Walking robots can operate in a much greater scope in terms of terrain diversity than classical wheeled robots originally developed to move on flat surfaces such as office floors. However, their greater motion capabilities are at the cost of increased control complexity that is primarily caused by the number degrees of freedom (DOF). For legged robots, DOF is usually significantly higher than for a car-like robot with 2 control DOF in comparison to 18 control DOF for a hexapod robot with three actuators per each leg.

One way to handle a high DOF is to generate a walking pattern—a gait [1]. A simple regular wave gait, where pairs of legs are regularly alternating, can be very efficient on flat terrains, where all legs in the support phase lie on the same plane. Moreover, for a perfect flat surface, it is just enough to rise each leg at the minimal height and move it forward. However, for rough terrains, the robot needs to traverse small obstacles and a leg can stand at a little bit different height than expected. Then, some of the legs can lose the ground support leaving them weaving uselessly in the air and the robot can stuck at that location incapable of moving towards the requested direction. Any single stair is therefore hardly traversable using a simple gait in an open-loop fashion.

The robot motion and its capability to traverse a rough terrain can be increased by closing the control loop and considering sensory information in the generation or execution of the motion gait. There can be found two complementary approaches based on exteroceptive and interoceptive sensors. The exteroceptive sensors such as range sensors can be used to build a map of the robot surroundings, which can be utilized to estimate an expected stability of the foothold locations [2]. Another way is to use a tactile sensor to ensure that a particular leg has been placed at the requested foothold location. Tactile information, e.g., using force sensors [3], is an important sense for crawling in a rough terrain because it allows to adapt the gait according to the terrain and to ensure the leg reaches the foothold.

Instead of direct force or contact sensors, an additional way how to determine the support level of the legs can be based on the feedback from the actuators provided by servo drives to prevent overloading. Palmer et al. proposed to utilize an additional passive actuator to read and control the load of the support level [4]. Information provided from the passive actuator is used to determine the joint torques and design the robot motion gaits that are independent on inertial and exteroceptive sensors, while the approach improves motion of a hexapod walking robot in rough terrains.

Based on [4], we investigated the problem of detecting the level of support, but rather than adding additional passive actuator, we directly rely on the active actuator itself. Thus, the proposed approach is even more minimalist as it does not require any sensors to provide a stable crawling on a rough terrain, where a regular motion gait does not provide the ability to traverse the terrain. Thus, the main contribution of the proposed approach is that no additional hardware or sensors adjustments are needed to make a blind hexapod robot walk smoothly over a rough terrain. We consider consumer smart serial servos to demonstrate that the proposed detection method enables usage of small and relatively cheap multi-legged robotic platforms for traversing rough terrains.

The paper is organized as follows. An overview of related work is presented in the next section. Considered assumptions and problem statement is presented in Section III. The proposed method to estimate the binary tactile information using information from the servo drive is described in Section IV. Results from real experimental deployment are reported in Section V and concluding remarks in Section VI.

II. RELATED WORK

A complex control architecture of quadruped walking robot to traverse challenging terrains has been presented in [5]. It is based on a fast locomotion controller with a selection of optimal foothold based on precise terrain templates (maps) [6], which are; however, created off-line by a precise laser scanning system.

Jakub Mrva and Jan Faigl are with the dept. of Computer Science, Czech Technical University in Prague, Technická 2, 166 27 Prague, Czech Republic `jakub.mrva@fel.cvut.cz`, `faiglj@fel.cvut.cz`

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

Authors of [7] showed how to improve traversability of a quadruped robot by a movement of the robot body to enlarge reachable areas for particular legs. In [3], an on-line force based foothold adaptation is utilized to ensure a smooth contact of the leg with the expected foothold. The force is computed using the load and torque sensors attached to each of the robot's joints, which increases the complexity and cost of the robot. Notice, that also in these approaches, the off-line map of the environment is considered to be available.

The off-line map can be avoided by on-line map creation. A stereo-vision system attached to a quadruped platform has been utilized in [8] to detect obstacles that are considered in path planning algorithm. However, such a map provides only a rough approximation of the robot's surrounding terrain that can be improved using tactile information [9]. Moreover, the tactile information can be even used for classification of the terrain as it has been shown by the same authors in [10].

A precise elevation map built from laser scanner data of the robot surroundings has been presented in [11]. The map is used to plan each single footstep in a long trajectory while considering the stability margin and avoiding collisions. Based on the map, the authors propose traversability assessment method in [2] that provides prediction of the traversability using a motion planning technique.

A map of the environment is considered to be available or it is created on-line in the aforementioned approaches. However, the crucial part of the robot motion control is a local adaptation of the motion gait based on the tactile sensing. Thus, a local motion controller can be build on a technically blind robot [12].

Authors of [4] proposed to use passive actuators to measure the ground reaction force and thus substitute direct force sensors. They built a custom hexapod robot (with 18 DOF)—in contrast with this paper—with a passive actuator in each leg yielding 24 actuators in total. Each leg has its own force threshold-based position controller driven by and controlling the ground reaction force estimated from the passive actuator. A similar idea has been proposed in [13], where the force at the tip of the leg is estimated using the torque values in each joint. A Hall effect based sensor is used to measure the current of the motors which is utilized to estimate the torque.

In this paper, we follow the idea of [4]; however, we propose to directly use the active actuators to control the robot motion together with the estimation of the contact point of the leg with the surface. Therefore, the main difference is that our method does not need additional actuators or sensor equipment; hence, it does not increase the cost of the robot nor the complexity of its hardware parts.

III. PROBLEM STATEMENT

The main problem being addressed in this paper is to detect the surface contact point using only the actuators with smart servo drives without any additional sensors. Thus, the robot is completely blind and the only information about the outer world has to be read through the servo drives. We consider a cheap and easy-to-used platform the PhantomX Hexapod Mark II with Dynamixel AX-12A actuators.

The pentapod gait with one leg moving at a time is considered to increase the gait stability in a rough terrain [14]. The gait diagram is shown in Fig. 1, where the motion strategy (highlighted blocks) and computation steps (non-highlighted blocks) during each leg cycle are presented. The legs are alternating in a metachronal gait with a given order: LF – RR – LM – RF – LR – RM.[1] The active leg leaves its foothold, moves forward, and begins approaching the ground. During the lowering phase, the data from actuators are analyzed for a possible ground detection. When a ground is detected, the leg motion is stopped and the body position and rotation are adjusted to better suit with the new position of the feet.

A. Hexapod Structure

The used hexapod platform has six legs each with three joints formed from the Dynamixel actuators. The schema of the leg and the description of its parts is depicted in Fig. 2. The robot can traverse small obstacles up to the limits of the robot structure. We consider the robot is operating in a rough environment that satisfies the robot's construction limits and there is not a large obstacle that the robot cannot traverse. In a case of large obstacles, we consider a rough map of the terrain can be created, e.g., using a camera or laser sensor, for a high level planning to find a traversable terrain similarly as in [11]. Thus, in this paper, we focused on the local motion control and on-line adaptation of the motion gait based on the detection of the contact point of the leg and the supporting surface.

B. Actuators

A method how to detect the ground is essential to successfully deal with a rough terrain. Since we have neither force nor torque sensors available as in [3], the crucial part of the proposed detection method comprises the actuator used and its feedback. The robot has Dynamixel AX-12A actuators, which—according to the manufacturer—enable

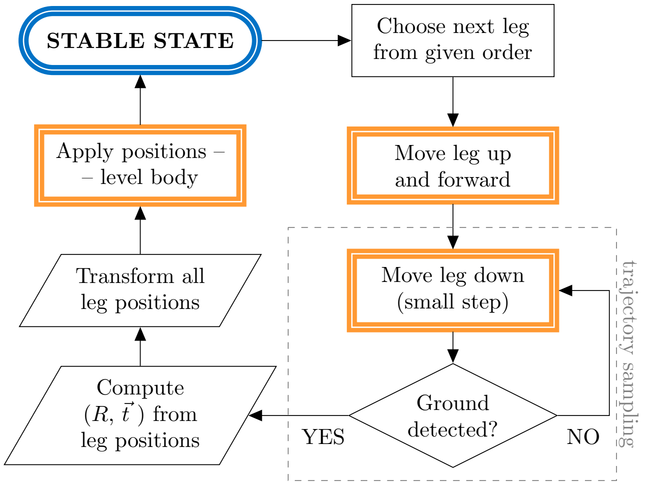


Fig. 1. Gait diagram. Only one leg is moving at a time in the swing phase during which the ground reaction force is measured. In all other phases of the cycle, all legs are in contact with the ground. R and $\vec{t}$ denote the robot rotation matrix and translation vector, respectively.

---

[1] LF = left-front, RR = right-rear, LM = left-middle, etc.

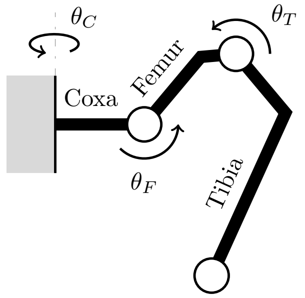


Fig. 2. Schema of the leg consisting of three parts (links)—Coxa, Femur, and Tibia. The three joints (θ_C, θ_F, and θ_T) are indexed according to the next respective link. The joint θ_C is fixed to the body with a vertical rotation axis while the other two joints have a horizontal rotation axis.

stable motions with robots designed for loads with $1/5$ or less of the stall torque. Notice, high values of the torque have to be avoided to prevent an accidental damage of the servo drives.

The actuators communicate via half-duplex serial line connected to a serial bus. The gait execution is realized in 33 ms control loop. The actuators can also be prompted to send their actual servo position; however, due to the communication limits we cannot get fresh values from more than two servos in each control loop. Therefore, we need to design the detection mechanism according to these limitations.

C. Tactile Sensing

Tactile sensing relies on measuring the counteracting force resulting from the leg contact with an obstacle. The source of the original force is the servo drive and its torque. The torque is considered to be linearly dependent on the servo position error, see Fig. 3. The actuators provide both the output position (joint angle) and the estimated torque values. Since the position values are more reliable than the torque values, we emulate the missing force (or torque) sensors by analyzing the position errors in joint angle values.

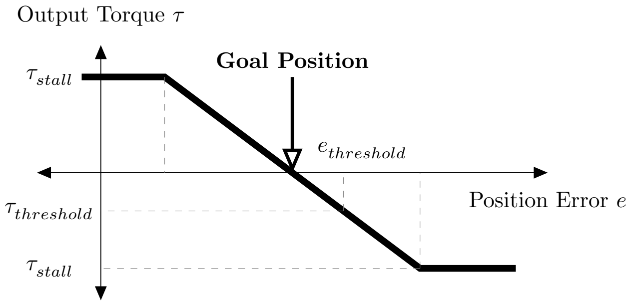


Fig. 3. The relation between the servo output torque τ and the position error e while controlling the goal position can be considered as linear. The output torque τ is limited by a stall torque value τ_{stall}.

A moment of the contact is detected by exceeding the $e_{threshold}$ value, which is illustrated in Fig. 3. This value depends on the overall robot weight and the distance between the footholds, which affects the acting momentum. Although the momentum acts on both θ_F and θ_T joints (see Fig. 2), the joint θ_F reflects the most of the overall ground reaction force and thus it is sufficient to use solely this particular servo in the emulation of a tactile sensor.

IV. SURFACE DETECTION AND ADAPTIVE MOTION GAIT

The proposed surface detection is based on the adaptive motion gait that determines new footholds considering the level of support (the load) of a leg in such a new position. We consider that the robot weight should be distributed among all legs, i.e., all legs should bear the same (or similar) load to support the robot stable state. In the pentapod motion gait, only one leg is moved, and the swinging leg starts to increase its load when it comes in a contact with the surface. Then, at some point, the actual load can be used to detect the ground.

The leg load is used to detect the surface contact point while a particular load per each leg is desired to support the robot stable state. It is not necessary the average load per each leg is the same as the load rate required for the ground detection. However, if the leg continues to lower with the load growing beyond the average, the robot would elevate on the leg; thus, causing some of the other legs to lose their ground support, which could furthermore lead to a slippage. Since we want to avoid such behavior, we tend not to exceed the load rate beyond the common average load during the ground sensing and the motion is stopped just when the leg reaches the common load rate.

Based on this idea of the expected load rate, we proposed the motion control, which can be divided into two particularly separate parts. Firstly, we need to handle the detection of the surface during the leg swing phase, which is described in Section IV-A. Secondly, due to the uncertainty in the final position in which the swinging leg finally ends, it is necessary to ensure the body heeds the current footholds. The proposed approach to address this issue is described in Section IV-B.

A. Ground Sensing and Detection

The ground detection is based on setting appropriate threshold values $e_{threshold}$ (see Fig. 3). These values are the expected values corresponding to the ground reaction force at the moment of the ground detection. We can infer such values from the average load rate of the legs for a robot in a default (starting) position on a perfectly flat terrain. For the considered hexapod robot, we can approximate the rate as

$$e_{threshold} \approx \frac{1}{6} \sum_{i=1}^{6} e_{default_i}, \tag{1}$$

where $e_{default}$ is the servo position error for the default robot configuration with all legs on the ground.

The proposed ground sensing algorithm works in a closed loop, which is shown in Fig. 1 as the dashed block—constantly switching between moving and comparing the received data with the threshold value obtained by (1) until the ground is detected. The used control loop frequency is related to the communication period $T = 33$ ms, and therefore, the motion of the swinging leg and sensor data are considered at the discrete time steps with the same period.

The data reading provides us the real joint angle values. Using the direct kinematics, we can recompute the real foot position (in the world coordinates) from the θ_F and θ_T joints

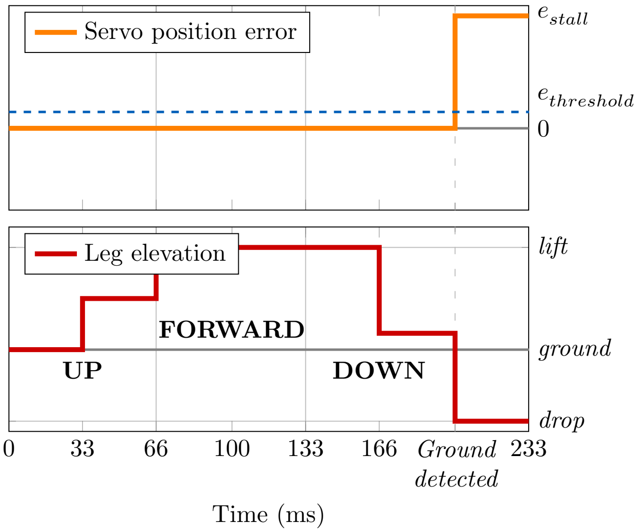


Fig. 4. A theoretical example of the servo position error dependency on the leg motion with only two discrete steps for particular leg motion phases and without considering dynamics. If each of the motion phases (up, forward, and down) are performed only in two steps, the ground is detected after the leg has reached its lowest position, which can cause a very high torque value (possibly near the stall torque) and thus it can cause serious damage.

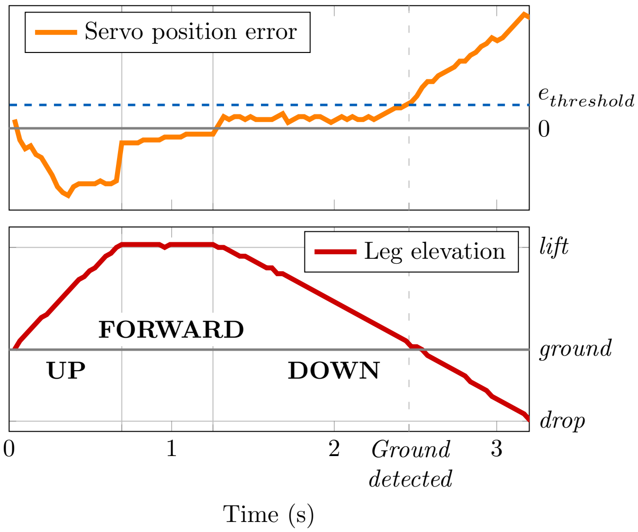


Fig. 5. An influence of the real motion dynamics to the position error during the leg motion. A moment of inertia of the leg acts in the opposite direction to the leg movement, which affects the position error. Raising a leg up is faster than its lowering in accordance with the measured servo position error in an active-leg cycle. When the leg is approaching the ground, the influence of the moment of inertia on the position error is significant with respect to the value of $e_{threshold}$.

(see Fig. 2), and compare it with the desired one to get the foot vertical displacement. However, the θ_F joint angle displacement shares the same tendency, therefore, only the displacement of θ_F is used as a sufficient approximation of the ground reaction force acting on the lowering leg.

For an idealized case, we can imagine an evolution of the position error during the full leg cycle as in Fig. 4, which does not provide a safe way to detect the contact point. Therefore, we need to slow down the motion of the leg during the down phase a bit and discretize the required drop value into a sequence of particular small steps. This allows to read the data and acts appropriately to avoid damage of the leg and safely detect the contact of the leg with the ground.

Fig. 5 shows the real data of the servo position error during particular phases of the leg cycle. The leg elevation plot traces the desired height of the foot above the ground. We can see that when the leg hits the ground (leg elevation crosses the zero value), the displacement rises. If the motion is not halted in that moment, the displacement continues to increase as the whole robot elevates itself on the moving leg.

The ground detection algorithm is now a simple threshold-exceeding detection as it is indicated in Fig. 5. The new foothold position (in the world coordinates) is remembered and used for computing the new body position.

B. Body Leveling and Movement

The body motion is separated from the leg motion according to the diagram in Fig. 1. Therefore, after a leg changed its foothold, the body has to counteract these changes by shifting and rotating into a more suitable position (in other words, to follow the legs). An optimal position can be very hard to find considering all DOF because each body position offers different possibilities of the movement depending on how close to the working space limits the legs are. Since the robot has to walk over a rough terrain without any perception about the terrain ahead, there is no option to choose any preferable body position in order to prepare for the oncoming terrain. Therefore, there has to exist an *equilibrium body position*, which offers balanced possibilities of the movement in all directions. The proposed body movement to its new position can be performed in the two following steps.

Firstly, we have to rotate the body to adapt to the new foothold positions. For this purpose, we use a simple linear regression. Having the foot positions[2], we can determine parameters a, b, and c of the plane (with the equation $z = ax+by+c$) that fits the foot positions, i.e., their squared distance from that plane is minimized. Then, the new body position is transformed to be parallel with this plane.

Secondly, we have to shift the rotated body to improve stability and leg working space margins. We can average the foot positions to get their "center", which we consider as the *equilibrium body position*. Notice, we are considering only the x and y coordinates of the new rotated plane. The new body $[x, y]$ position can therefore be expressed as the average of the rotated foot $[x, y]$ positions. The body height (z coordinate) is then adapted to keep the body at the default height 0.1 m above the estimated plane. The transformation of the body coordinates can be expressed as

$$\begin{bmatrix} x'_B \\ y'_B \\ z'_B \\ 1 \end{bmatrix} = \begin{bmatrix} R & R\vec{t} \\ 0\ 0\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_B \\ y_B \\ z_B \\ 1 \end{bmatrix} . \tag{2}$$

---

[2] The world coordinate system xyz is oriented as follows: z-axis is pointing vertically upwards, the x-axis is heading in the robot's forward direction and the y-axis is pointing to the left; hence, they form a right-handed coordinate system.

Note that the translation is preceded by the rotation; hence, the translation vector $\vec{t}$ is multiplied by the rotation matrix R.

Actually, to move the body, we need to move all the legs into the opposite direction. Therefore, the new foot positions can be expressed by an inverse transformation

$$\begin{bmatrix} x'_i \\ y'_i \\ z'_i \\ 1 \end{bmatrix} = \begin{bmatrix} R^T & -\vec{t} \\ 0\;0\;0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}, \qquad (3)$$

where R can be created by setting up its basic vectors individually. For a better readability, we create an orthogonal (not orthonormal) basis $\begin{bmatrix} \vec{b}_x & \vec{b}_y & \vec{b}_z \end{bmatrix}$ first, and norm its vectors later. Since the forward walking direction has to be preserved, the first basic vector $\vec{b}_x$ can be formed directly from the regression plane as $\vec{b}_x = \begin{bmatrix} 1 & 0 & a \end{bmatrix}^T$. The third basic vector $\vec{b}_z$ is heading upward and it is perpendicular to the regression plane and thus it can be obtained directly from its general form $0 = ax + by - z + c$. Hence, we get $\vec{b}_z = \begin{bmatrix} -a & -b & 1 \end{bmatrix}^T$. The last basic vector is simply any vector that is linearly independent. Such a vector that completes an orthogonal basis is $\vec{b}_y = \begin{bmatrix} -ab & a^2+1 & b \end{bmatrix}^T$. The resulting orthonormal rotation matrix R is created from the basic vectors dividing them by their norms as follows

$$R = \left[\begin{array}{c|c|c} 1 & -ab & -a \\ 0 & a^2+1 & -b \\ a & b & 1 \end{array}\right] \begin{bmatrix} ||\vec{b}_x|| & 0 & 0 \\ 0 & ||\vec{b}_y|| & 0 \\ 0 & 0 & ||\vec{b}_z|| \end{bmatrix}^{-1}. \qquad (4)$$

The translation vector can be expressed by rewriting (3) line by line as follows

$$x'_i = \frac{\vec{b}_x \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}}{||\vec{b}_x||} - t_x, \qquad (5)$$

$$y'_i = \frac{\vec{b}_y \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}}{||\vec{b}_y||} - t_y, \qquad (6)$$

$$z'_i = \frac{\vec{b}_z \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}}{||\vec{b}_z||} - t_z. \qquad (7)$$

Note that the $[x', y']$ coordinates are the new foot positions, which are designed in a way the body position is computed from their average. Hence, from the sum of (5) and (6), we can directly express the parameters t_x and t_y. The last coordinate of the translation vector $\vec{t}$ has to compensate the change in the body height h above the ground. It can be computed easily using the similarity of triangles as $\frac{1}{||\vec{b}_z||} = \frac{t_z - h}{c}$. The translation vector is therefore

$$\vec{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} \frac{\sum x_i + a \sum z_i}{6||\vec{b}_x||} \\ \frac{-ab \sum x_i + (a^2+1) \sum y_i + b \sum z_i}{6||\vec{b}_y||} \\ \frac{c}{||\vec{b}_z||} - h \end{bmatrix}. \qquad (8)$$

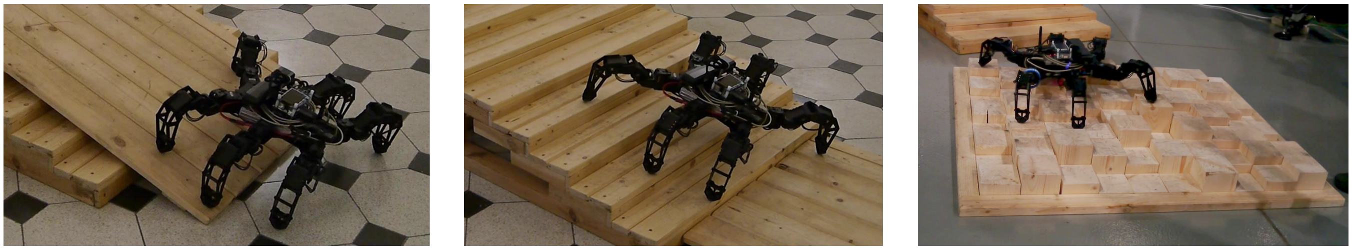
The body move is achieved by applying the transformation of all leg coordinates from (3) and execution of the motion to get the legs to their new positions. Since the legs are always moving a bit forward—though the distance between the new and old foot positions is variable—and the body position is computed as an average of the foot positions, the body is therefore following the legs, no matter which leg or how far the leg is moving. This small moves ensure the whole body movement and thus the movement of the whole robot.

V. EXPERIMENTAL RESULTS

The proposed detection of the surface contact point accompanied with the developed enhanced gait has been verified in a series of experimental scenarios. First, a flat floor has been considered to ensure that there is not a significant drop in the robot's ability to traverse a simple terrain. In this scenario, the ground detection threshold (see Fig. 5) has been carefully set with respect to the robot's weight to react appropriately on the ground contact. Then, we consider three scenarios with a rough terrain in which the robot is unable to traverse them by the default gait albeit all of the scenarios were prepared with respect to the dimensions of the hexapod. The scenarios are depicted in Fig. 6 and it consists of the inclined plane, stairs, and a set of blocks with various height. The proposed detection mechanism utilized in the developed adaptive motion gait allows the robot to traverse all of the terrains smoothly and the particular performance of the robot has been as follows:

1) The inclined plane scenarios shown in Fig. 6a does not provide significant difficulties to traverse the breaking point even for a slope greater than 20°. An example of the motion is captured in Fig. 7. Although the sloped terrain (made of wood) is a bit slippery for our robot, the proper ground detection allows to avoid trying to lift the robot body, which could accidentally yield in a loss of support of several legs and thus sliding the legs down the inclined plane. On the other hand, the robot with the regular motion gait remained stuck at the edge of the inclined plane with no further progress despite the continuous gait execution, which has been observed even for a slightly inclined plane, i.e., about 10°.

2) In the next scenario shown in Fig. 6b, the robot has been able to successfully climb up the stairs. The terrain is more challenging because the stairs provide less feasible footholds than a simple plane and edges of the stairs are particularly difficult for the robot. Though, the occurrence of a slippage is less likely to appear due to the horizontal surface of each stair. The main issue in this scenario has been observed when the robot stepped on an edge. Despite it did not cause a downfall immediately, when the leg became more loaded due to the other legs movement, its foot fell one step down. This case resulted in a slight loss of stability with several legs hanging in the air (the robot has always support of at least three legs, naturally). Because the robot is technically blind, such an accident is not avoidable, though. But, the robot is able to regain its lost stability within a few following steps



(a) Inclined plane (10° / 20° slope) (b) Stairs (4 cm height) (c) 10x10 cm blocks of variable height 3-15 cm

Fig. 6. Testing scenarios.

Fig. 7. Hexapod traversing an inclined plane (20° slope).

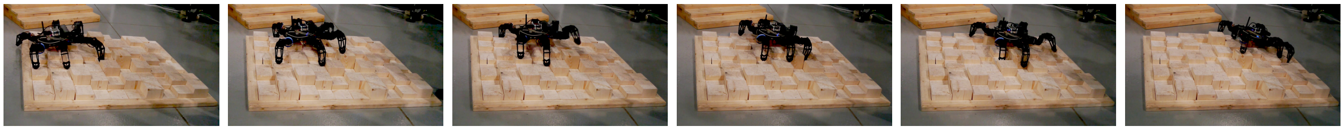


Fig. 8. Hexapod traversing wooden blocks.

due to the adaptive behavior of the leg ground approaching method. Soon or later, the robot has a five-leg support again.

3) Finally, in the last scenario shown in Fig. 6c, the robot exhibits a similar behavior as for the stairs scenario and it is able to successfully traverse the wooden blocks repeatably. The only issue is related to the height of the blocks, where the tallest blocks cannot be just next to the lowest ones because the robot is physically incapable to traverse such obstacles due to the length of its legs.

VI. CONCLUSION

We propose a method to detect the surface contact point that is solely based on the feedback provided by the smart servo drive. The method is employed in the developed adaptive motion gait that allows the hexapod walking robot to traverse rough terrains using a pentapod gait. The proposed approach does not rely on any additional sensors and thus its main benefit is in easy deployment on cheap platforms that are basically composed only from a body, legs, and servo drives. Although the approach does not provide motion capabilities for challenging rough terrains, it enhances the robot motion that is basically limited only to flat surfaces. Thus, we believe the proposed simple way how to substitute a dedicated tactile sensor enables deployment of cheap hexapod walking robots in further research and applications.

REFERENCES

[1] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*. New York, NY, USA: Cambridge University Press, 2000.

[2] D. Belter, P. Łabęcki, and P. Skrzypczyński, "An exploration-based approach to terrain traversability assessment for a walking robot," in *Proceedings of the IEEE International Conference on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2013.

[3] A. Winkler, I. Havoutis, S. Bazeille, J. Ortiz, M. Focchi, R. Dillmann, D. Caldwell, and C. Semini, "Path planning with force-based foothold adaptation and virtual model control for torque controlled quadruped robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 6476–6482.

[4] L. Palmer and M. Palankar, "Blind hexapod walking over uneven terrain using only local feedback," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2011, pp. 1603–1608.

[5] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal, "Learning, planning, and control for quadruped locomotion over challenging terrain," *The International Journal of Robotics Research*, vol. 30, no. 2, pp. 236–258, February 2011.

[6] M. Kalakrishnan, J. Buchli, P. Pastor, and S. Schaal, "Learning locomotion over rough terrain using terrain templates," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, October 2009, pp. 167–172.

[7] V.-G. Loc, I. M. Koo, D. T. Tran, S. Park, H. Moon, and H. R. Choi, "Improving traversability of quadruped walking robots using body movement in 3d rough terrains," *Robotics and Autonomous Systems*, vol. 59, no. 12, pp. 1036–1048, 2011.

[8] X. Shao, Y. Yang, and W. Wang, "Obstacle crossing with stereo vision for a quadruped robot," in *International Conference on Mechatronics and Automation (ICMA)*, 2012, pp. 1738–1743.

[9] K. Walas, "Improving accuracy of local maps with active haptic sensing," in *Robot Motion and Control 2011*, ser. Lecture Notes in Control and Information Sciences, K. Kozlowski, Ed. Springer London, 2012, vol. 422, pp. 137–146.

[10] ——, "Tactile sensing for ground classification," *Journal of Automation, Mobile Robotics & Intelligent Systems*, vol. 7, no. 2, pp. 18–23, June 2013.

[11] D. Belter and P. Skrzypczyński, "Rough terrain mapping and classification for foothold selection in a walking robot," *Journal of Field Robotics*, vol. 28, no. 4, pp. 497–528, July 2011.

[12] M. Palankar, "A distributed local-leg feedback algorithm for robust walking on uneven terrain," Ph.D. dissertation, University of South Florida, May 2013.

[13] K. Walas and D. Belter, "Supporting locomotive functions of a six-legged walking robot," *International Journal of Applied Mathematics and Computer Science*, vol. 21, no. 2, pp. 363–377, June 2011.

[14] D. Belter, "Gait modification strategy for a six-legged robot walking on rough terrain," *Proceedings of the 15th International Conference on Climbing and Walking Robots, Adaptive Mobile Robotics, World Scientific, A. Azad et al. (Eds.), Singapore*, vol. 23, pp. 367–374, 2012.