CLAWAR 2020: 23rd International Conference on
Climbing and Walking Robots and the Support
Technologies for Mobile Machines,
Moscow, Russian Federation, 24-26 August 2020.
https://doi.org/10.13180/clawar.2020.24-26.08.id#

ON THE INFLUENCE OF BODY VELOCITY IN FOOTHOLD ADAPTATION FOR DYNAMIC LEGGED LOCOMOTION VIA CNNs

DOMINGO ESTEBAN, OCTAVIO VILLARREAL, SHAMEL FAHMI,
CLAUDIO SEMINI, and VICTOR BARASUOL

Dynamic Legged Systems lab, Istituto Italiano di Tecnologia, Via Morego 30, 16163, Genoa, Italy
E-mail: firstname.lastname@iit.it, Website: https://dls.iit.it/

This paper analyzes the importance of considering the body velocity for foothold adaptation during legged locomotion. We show how this velocity affects the decisions made by a foothold adaption strategy, and the number of feasible footholds computed by the approach. We extend our previous work by considering the body velocity in this foothold adaptation method and augmenting a convolutional neural network (CNN) classifier to account for the current velocity of the robot. Our results suggest that the foothold evaluation strategy has a better performance with this new CNN than with architectures that assume a constant velocity and that only consider heightmaps as input for the foothold evaluation.

1. Introduction

Legged robots require autonomy to deal with obtacles during navigation. Some approaches rely only on proprioceptive feedback.[1–3] However, they are limited to reactive adaptations and lack the benefits of anticipative actions computed using exteroceptive sensors (e.g., cameras and laser scanners). Alternatively, optimization-based algorithms[4,5] allow online body motion and foothold replanning using terrain maps but their computational complexity limits their application to slow (quasi-static) gaits. For this reason, some approaches[6–9] use machine learning models to approximate the computationally heavy operations related to terrain analysis. A convolutional neural network (CNN), for example, can be used to evaluate the geometry of terrains and identify rough areas to avoid foot and leg collisions.[7–9]

In a previous work, we proposed a CNN-based foothold adaptation strategy to infer safe foothold locations from terrain data in approximately 0.2 ms, which allowed us to apply the approach for dynamic locomotion.[7] However, this approach assumed that the robot moves with a constant velocity during the entire swing phase which has some disadvantages in scenarios similar to Fig. 1: (i) the approach will output reasonable foothold locations for that specific velocity, being less robust for higher velocities and too conservative for lower ones; (ii) in case of slow speeds, the robot behaves aggressively around sharper objects, like steps and stairs, taking larger corrections when is not needed; and (iii) if the foothold evaluation is too conservative, it does not find the safe foot placements between obstacles that are relatively too close to each other.

In this paper, we address these challenges, and propose an alternative approach to overcome them. The contributions of this paper are:

(1) An analysis of the effects of the robot's forward velocity on the evaluation of foothold adaptations based on kinematic criteria.
(2) A new formulation for the CNN-based classifier that also considers the forward velocity of the robot as input. We show that this CNN has a better performance than architectures that assume a constant velocity and consider only heightmap values.

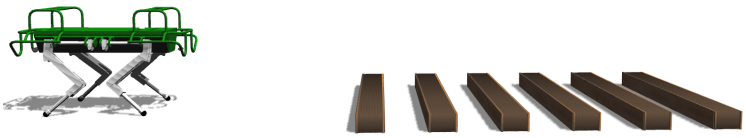

Fig. 1. *Simulated scenario for evaluating the proposed approach.*

The paper is organized as follows: in Section 2, we describe the foothold adaptation strategy considered in this paper. Later in Section 3, we analyze the effects of the robot's forward velocity on the kinematic criteria used in this framework. In Section 4, we propose a CNN that explicitly considers the forward velocity of the robot to approximate the heightmap evaluation process.

2. Vision-Based Foothold Adaptation

The short time window in which a decision needs to be made is one of the reasons why foothold adaptation in dynamic legged locomotion is a complex problem. To tackle this problem, the *Vision-based Foothold Adaptation (VFA)*[7] strategy was proposed based on using visual information about the terrain in a fast and continuous fashion. In this section, we briefly describe the VFA for a better understanding of the paper (for a detailed description refer to [7]). The foothold adaptation process is divided into four stages (see Fig. 2):

1. Foothold prediction: we estimate the landing position of the foot during the swing phase (called *nominal foothold*) as:

$$\hat{\mathbf{p}} = \bar{\mathbf{p}} + \frac{D_f \mathbf{V}_f}{2 f_s} + \Delta t \dot{\mathbf{r}}, \tag{1}$$

where $\hat{\mathbf{p}}$ is the nominal foothold, $\bar{\mathbf{p}}$ is a reference position on the ground determined by the trajectory of the leg, D_f is the duty factor, $\mathbf{V}_f$ is the commanded forward velocity, f_s is the step frequency, Δt is the remaining swing-time of the leg, and $\dot{\mathbf{r}}$ is the current body velocity.

2. Heightmap extraction: given a predicted nominal foothold, we construct a heightmap $\mathbf{H}$ with height values expressed with respect to a frame located at the hip position, and whose center corresponds to the nominal foothold.

3. Heightmap evaluation: given a heightmap $\mathbf{H}$, each cell is considered a candidate foothold $\mathbf{p}_c$ and evaluated according to four criteria (see subsection 2.1) to determine if it is feasible. We denote this evaluation as $h(\mathbf{H})$. Among all the feasible footholds, we select the closest to the nominal as the *optimal foothold*.

4. Foot trajectory adjustment: the relative displacement between the nominal foothold and the optimal foothold $\Delta\mathbf{p}$ is sent to a trajectory generator to adapt the original foot swing trajectory.

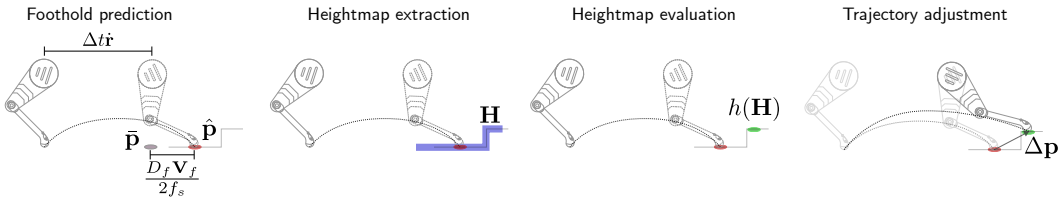

Fig. 2. *Phases of the VFA*. From left to right, initially, the nominal foothold $\hat{\mathbf{p}}$ is predicted based on the motion of the trunk and the predefined trajectory of the leg. Then, a heightmap $\mathbf{H}$ is extracted around the vicinity of the nominal foothold. From the heightmap, an optimal foothold is evaluated according to $h(\mathbf{H})$. Once the location of the optimal foothold is computed, the trajectory of the leg is continuously adjusted to adapt it to the new foothold location using vector $\Delta\mathbf{p}$.

2.1. *Heightmap Evaluation*

The VFA evaluates a heightmap centered around the nominal foothold $\hat{\mathbf{p}}$ based on the following criteria:

Terrain roughness: for each candidate foothold location $\mathbf{p}_c$ we compute the mean and the standard deviation of the slope relative to its neighborhood. We discard those footholds whose sum of mean and standard deviation is larger than a specific threshold.

Kinematic feasibility: we discard the candidate foothold locations that are outside of the workspace of the leg.

Foot frontal collision: for a given lift-off position, we discard those candidate foothold locations whose swing trajectory collides with the terrain.

Leg collision: we discard those candidate footholds that might produce a collision of the leg limbs with the terrain during the whole step cycle (i.e., stance and swing phases).

Distance to nominal foothold: in case several footholds satisfy all the previous criteria, we select the closest to the nominal foothold, namely the *optimal foothold*.

This heightmap evaluation process only considers the current heightmap in order to select the optimal foothold. This is possible thanks to a number of assumptions made about the robot's motion, e.g., that the trunk velocity remains constant during the swing phase of the legs. In the following section, we describe how the body velocity influences the result of the aforementioned criteria, and motivates its explicit consideration in the evaluation of footholds.

3. Influence of the Body Velocity on Vision-Based Foothold Adaptation

This section analyzes the influence of the body velocity on the heightmap evaluation and the number of feasible footholds computed by the VFA. The body velocity $\mathbf{V}_f$ plays a crucial role in the heightmap evaluation because this velocity changes the displacement of the hip, affecting the criteria related to the leg kinematics (namely, kinematic feasibility and leg collision). The rest of this section details the effects of the body velocity on these criteria.

3.1. *Influence on the Heightmap Evaluation*

Kinematic feasibility: a candidate foothold $\mathbf{p}_c$ is kinematically feasible if it lies inside the leg workspace during the whole stance period. During the stance period, the hip of the leg moves with respect to its foothold, changing its workspace accordingly. Hence, to evaluate the kinematic feasibility criteria, we need to consider the leg workspace during the touchdown and the next lift-off of the leg (at the beginning and the end of the stance period). The kinematically feasible candidate footholds are the footholds that overlap the leg workspaces during touchdown and next lift-off. This overlapping is inversely related to the body velocity $\mathbf{V}_f$ because an increment in $\mathbf{V}_f$ increases the hip displacement, decreasing the overlapping. Figure 3a illustrates this correlation between kinematic feasibility and body velocity. It shows the configuration of the leg during the touchdown and the next lift-off, and the corresponding overlapping workspaces, for two different velocities ($\mathbf{V}_{f1}$ and $\mathbf{V}_{f2}$). As shown, the kinematically feasible footholds (the horizontal red and blue lines) are inversely proportional to the body velocity.

Leg collision: a candidate foothold $\mathbf{p}_c$ is collision free if the configuration of the leg does not collide with the terrain during the step cycle. The body velocity $\mathbf{V}_f$ changes the hip displacement, which might result in configurations where the leg collides with the environment. For example, with an increment in body velocity, the shin might become closer to the

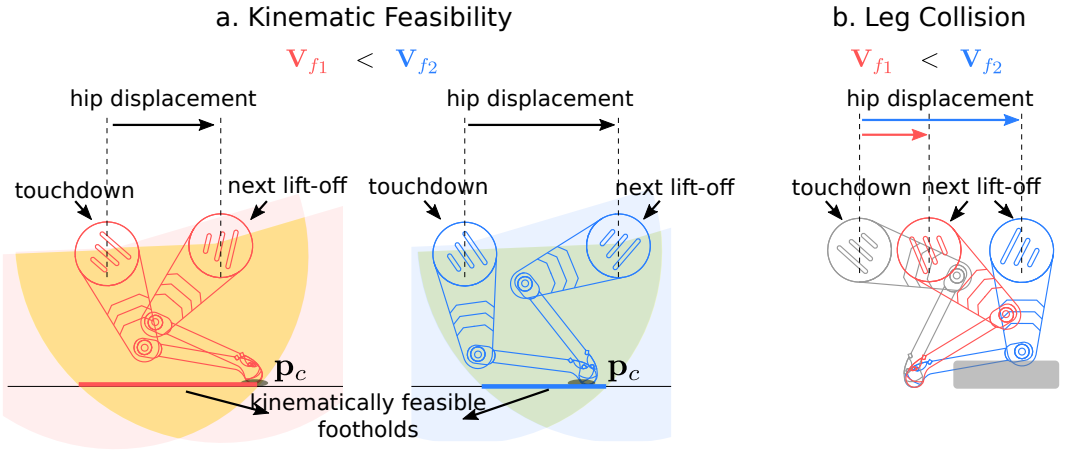

Fig. 3. *An overview of the influence of the body velocity on two foothold evaluation criteria: (a) kinematic feasibility and (b) leg collision for a low ($\mathbf{V}_{f1}$) and a high ($\mathbf{V}_{f2}$) body velocity.* The shaded regions represent the workspace of the robot. The variable $\mathbf{p}_c$ is a candidate foothold. The horizontal red and blue lines represent the kinematically feasible footholds.

ground, as shown in Fig. 3b. This situation reduces the number of feasible footholds due to the leg collision criterion.

3.2. *Influence on the Number of Feasible Footholds*

The number of feasible footholds is implicitly affected by the body velocity, because it depends on the evaluation criteria mentioned in Section 3.1. To verify this, we performed two simulations in the scenario shown in Fig. 1, and collected the heightmaps resulting from these simulations. In the two simulations, we use body velocities of 0.2 m/s, and 0.5 m/s, respectively, while keeping the other settings identical (step frequency of 1.4 Hz, duty factor of 0.6, and step height of 0.12 m). After collecting the heightmaps, we evaluate the VFA offline using the criteria detailed in Section 2.1 while considering the body velocity as an input.

The outcome of these simulations is shown in Fig. 4 and Fig. 5. In Fig. 4, we plot the number of feasible footholds computed by evaluating the VFA for the four legs for the two velocities. As shown in the figure, lower body velocities can result in higher number of feasible footholds. In Fig. 5, we plot the *distance to nominal* of the four legs for the two velocities. The distance to nominal is the distance between the optimal foothold from the VFA, and the nominal foothold $\hat{\mathbf{p}}$. We use this metric to indicate how far the optimal foothold is from the nominal one. The shorter the distance to the nominal foothold, the smaller the magnitude of the disturbance on the robot due to the foothold adaptation. As shown in the figure, lower body velocities can result in smaller values of the distance to nominal.

Figures 4 and 5 show that some body velocities can give more feasible solutions to the robot that are closer to the nominal footholds. To elaborate, in the simulation where the robot was trotting with a forward velocity of 0.2 m/s, if the VFA of our previous work was used (where the forward velocity was assumed to be constant of 0.5 m/s in the VFA evaluation), it will result in less number of feasible footholds that are further away from the nominal foothold. This can put the robot in situations with less alternatives that might lead to unsafe footholds if there is a large uncertainty in the heightmap.

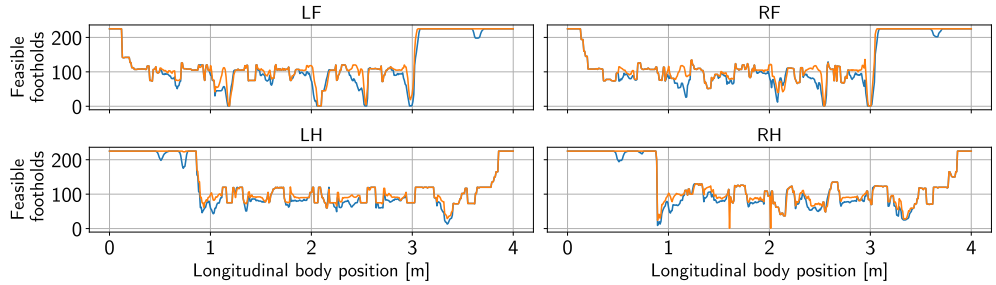

Fig. 4. *Number of feasible footholds.* The number of feasible footholds according to the evaluation criteria for a body velocity of 0.2 m/s (orange), and 0.5 m/s (blue) using the scenario shown in Fig. 1.

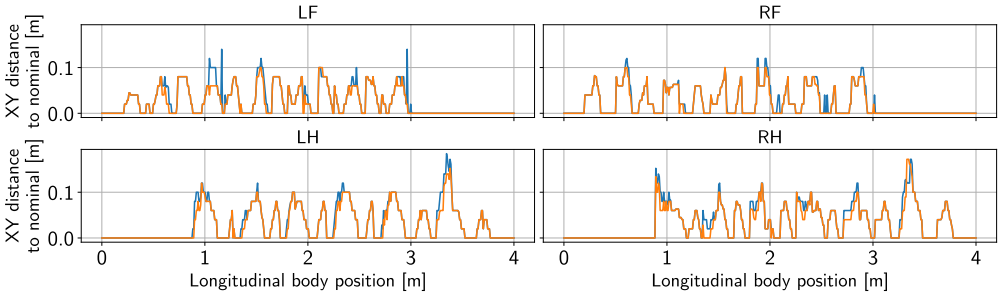

Fig. 5. *Distance to nominal.* The distance to nominal (the distance between the optimal foothold from the VFA and the nominal foothold) for a body velocity of 0.2 m/s (orange), and 0.5 m/s (blue) using the scenario shown in Fig. 1.

4. Velocity-Aware Foothold Evaluation with Convolutional Neural Networks

The VFA was devised to provide safe foothold locations during dynamic locomotion. This means that the foothold adaptation should be performed continuously in short periods of time. However, the heightmap evaluation (Section 2) is a computationally demanding operation. As a solution, the VFA approximates the evaluation $h(\mathbf{H})$ with a CNN to infer the optimal foothold in a faster way[7] (see Fig. 6a). This network consists of two convolutional layers with 5×5 kernels, 2×2 padding, ReLU activation and 2×2 max-pooling operation, and two fully-connected layers with ReLU and softmax activations. This CNN only considers a heightmap as input, reason why it is trained with foothold evaluation data that assumes that the robot moves with a constant velocity.

To approximate a heightmap evaluation that explicitly considers the forward velocity of the robot, $h(\mathbf{H}, \mathbf{V}_f)$, we slightly modified the aforementioned CNN by concatenating the velocity value to the one-dimensional feature vector obtained from the convolutional layers (see Fig. 6b). Our decision is based on: (i) exploiting an architecture that has already shown a good performance in the extraction of features from heightmaps, (ii) capturing the non-linear dependence of the forward velocity in the foothold evaluation, thus evaluating it with at least two fully-connected layers, and (iii) making a minimal increment in the number of parameters (8237 parameters for the velocity-aware CNN compared to 8217 parameters for the previous CNN).

5. Results

In this section, we evaluate the effect of including the forward velocity on the prediction accuracy of the CNNs. Later, we analyze how considering this forward velocity affects the

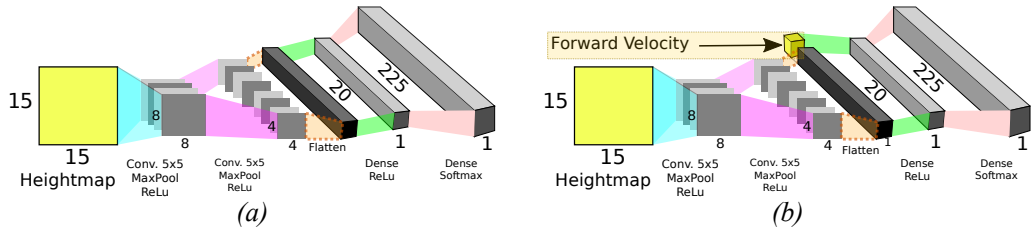

Fig. 6. *Convolutional neural networks that approximate the evaluation of a heightmap. (a) Heightmap-only CNN.* This model considers only current heightmap values and predicts optimal footholds assumming the robot moves with a constant velocity. *(b) Velocity-aware CNN.* This neural network explicitly considers the current forward velocity as input. It has an architecture similar to the CNN in *(a)*. Note, however, that the forward velocity is concatenated with the feature vector obtained from the convolutional layers before passing through the fully-connected ones.

foothold adaptation of a legged robot in a simulation experiment.

5.1. *CNN Prediction*

We trained three different types of CNNs. The first two, denoted by $\hat{h}_{0.2}$ and $\hat{h}_{0.5}$, were trained using $\mathbf{V}_f = 0.2$ m/s and $\mathbf{V}_f = 0.5$ m/s fixed forward velocity, respectively. The architecture of these CNNs is shown in Figure 6a. The third type of CNN, denoted by $\hat{h}_{\mathbf{V}}$, has a velocity-aware architecture (Figure 6b) and it is trained with evaluations that consider velocities between 0.0 m/s and 0.6 m/s. Therefore, in total we trained 12 different CNNs, one per each leg and CNN type.

The parameters of all the CNNs are optimized to minimize the cross-entropy loss of classifying a candidate foothold location as optimal according to the evaluation criteria described in Section 2.1. We used the Adam optimizer with a learning rate of 0.001, and 16316 heightmaps collected from simulation, which are split in 80% as training set and 20% as validation set. We have performed the training process with five different random seeds, and the CNNs with better performance in the validation set are the ones considered in this section. To analyze their performance, we had a test set (1480 new heightmaps) of evaluations with velocities of 0.2 m/s, 0.5 m/s, and 12 equally spaced velocities in the range of 0.0 m/s and 0.6 m/s.

The performances of the trained CNNs in predicting optimal-footholds are summarized in Table 1. The $\hat{h}_{\mathbf{V}}$ CNNs have, on average, a better optimal-foothold prediction than the $\hat{h}_{0.2}$ and $\hat{h}_{0.5}$ networks. We believe that this performance is due to the decoupling of the velocity-dependent features that should be obtained from the convolutional layers. That is, concatenating the velocity after the convolutional layers allows them to focus on the construction of features relevant to the foothold evaluation but not related to the velocity. The performance of $\hat{h}_{\mathbf{V}}$ is also better in giving solutions that are feasible and closer to the true optimal footholds.

An unexpected result is that despite the $\hat{h}_{0.2}$ CNNs were trained with evaluation data that considers a velocity of 0.2 m/s, their performance is worse than CNNs trained with a velocity of 0.5 m/s (see column $\mathbf{V}_f = 0.2$ m/s in Table 1). As we could see in Section 3, lower forward velocities make the locations of the initial foothold and the hip of the robot in the heightmap closer to the nominal foothold, and thus closer to its center. If we also consider that in most cases, the optimal-foothold is even closer to the center, we have a significant amount of optimal-footholds closer to the nominal. This produces a CNN slightly biased towards solutions closer to the center and that ignores solutions far from it.

Table 1. *Optimal-foothold prediction of three different CNNs.* The table shows the performance on the test set for evaluations with velocities of 0.2 m/s, 0.5 m/s, and 12 velocities in the range of 0.0 m/s and 0.6 m/s. The left-front, right-front, left-hind and right-hind legs are denoted by LF, RF, LH and RH, respectively. The values reported in the perfect-match and feasible footholds rows are expressed in percentage (%) over the total size of the test set. On the other hand, the values of the distance to optimal (last row) are in cm. A CNN that considers the velocity as input has a better performance, in terms of perfect-matching, prediction of feasible footholds, and distance to the true optimal. See subsection 5.1 for further discussion.

		$\mathbf{V}_f = 0.2$m/s			$\mathbf{V}_f = 0.5$m/s			$\mathbf{V}_f \in [0.0, 0.6]$ m/s		
	Leg	$\hat{h}_{0.2}$	$\hat{h}_{0.5}$	$\hat{h}_{\mathbf{V}}$	$\hat{h}_{0.2}$	$\hat{h}_{0.5}$	$\hat{h}_{\mathbf{V}}$	$\hat{h}_{0.2}$	$\hat{h}_{0.5}$	$\hat{h}_{\mathbf{V}}$
Perfect match	LF	87.70	88.90	**95.30**	87.30	89.20	**94.40**	87.56	88.85	**94.87**
	RF	87.70	91.80	**96.00**	87.30	92.00	**95.10**	87.46	91.68	**95.59**
	LH	85.10	89.10	**93.10**	85.50	90.10	**93.90**	85.34	89.43	**93.37**
	RH	86.50	89.70	**93.50**	87.00	91.10	**94.20**	86.82	90.21	**93.81**
Feasible foothold	LF	95.60	96.60	**98.50**	95.10	96.50	**98.50**	95.43	96.48	**98.45**
	RF	96.10	97.40	**98.80**	95.60	97.20	**98.20**	95.88	97.26	**98.59**
	LH	95.30	**97.90**	97.60	94.40	97.60	**98.30**	94.88	97.74	**97.88**
	RH	95.50	**97.40**	**97.40**	94.40	97.30	**98.50**	95.01	97.30	**97.97**
Distance to optimal	LF	0.61	0.63	**0.20**	0.67	0.57	**0.44**	0.64	0.62	**0.22**
	RF	0.62	0.51	**0.16**	0.68	0.45	**0.27**	0.65	0.50	**0.19**
	LH	0.72	0.62	**0.29**	0.70	0.56	**0.33**	0.72	0.60	**0.28**
	RH	0.71	0.56	**0.25**	0.67	0.50	**0.27**	0.69	0.54	**0.27**

5.2. *Simulations*

We evaluated the proposed improvements to the VFA and the accuracy of the trained CNN on the simulation scenario depicted in Fig. 1. It is comprised by a series of bars of 15 cm high, 20 cm wide, equally spaced 30 cm. We chose this type of scenario because the spacing between the bars makes it necessary for the robot to step on the ground when crossing from one bar to another. This situation has high chances to lead to leg collisions, which is the evaluation criterion that has the more relevant effect when considering the body velocity to select safe footholds.

The robot is commanded to trot at a forward velocity $\mathbf{V}_f$ of 0.2 m/s, a step frequency f_s of 1.4 Hz and a duty factor D_f of 0.6. The body velocity $\dot{\mathbf{r}}$ is fed into the CNN continuously. We are mostly interested in performing locomotion at slow velocities, since it is when the improvement in performance is most noticeable, as stated in Section 3. However, we are also interested to show how the constant update of the forward velocity plays a role in the selection of future foothold locations.

Figure 7 shows a comparison of the body velocity in the x direction of the body, when crossing the scenario depicted in Fig. 1, using three different CNNs. As it can be seen, the variation of the body velocity as the robot crosses the scenario is less when the CNN is trained for different forward velocities. This is because the selection of the foothold reduces the number of impacts with the bars. In the case of the CNN trained considering a constant forward velocity of 0.2 m/s, although that is the commanded velocity, the actual robot's velocity suffers variations caused by the changes in foothold location with respect to the nominal ones. These changes perturb the robot and consequently change its acceleration. The variations in velocity lead to kinematic configurations that were not considered during training, which leads to leg collisions. On the other hand, the CNN trained assuming 0.5 m/s forward velocity, generates aggressive actions, either to far from the obstacle, or continuously trying to overcome it. Effectively, these large displacements with respect to the position of the nominal foothold increase the size of the disturbance on the trunk, leading to higher variation in forward velocity.

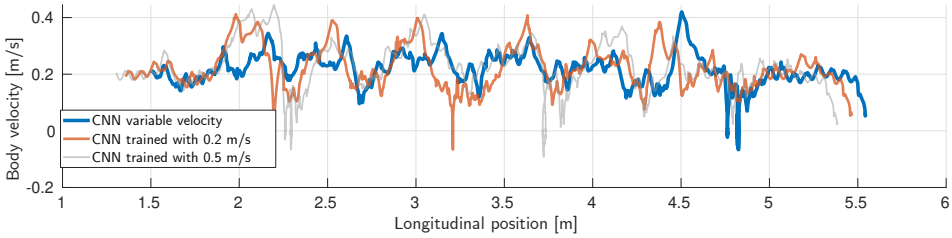

Fig. 7. *Comparison of body velocity using three different CNNs while crossing the scenario depicted in Fig. 1.* The CNNs were trained considering three conditions for the body velocity: constant at 0.2 m/s (red), constant at 0.5 m/s and variable body velocity (blue). Five trials were performed for each CNN and we show the average of them.

6. Conclusions

In this paper, we have analyzed the influence of the body velocity in foothold adaptation during dynamic legged locomotion. We have shown how this velocity is directly related to hip displacement. Therefore, it alters the evaluation of the kinematic criteria in a framework that selects feasible footholds based on them. In addition, we have proposed a CNN classifier that explicitly considers the velocity of the robot and approximates the foothold evaluation process. Our results show that our foothold evaluation strategy has a better performance with this CNN than with architectures that assume a constant velocity and that only consider heightmaps as input. We plan to improve the performance of the CNN by considering additional variables, testing new architectures and training with data collected from the real robot.

References

1. V. Barasuol, J. Buchli, C. Semini, M. Frigerio, E. R. De Pieri, and D. G. Caldwell, "A reactive controller framework for quadrupedal locomotion on challenging terrain," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2554–2561, 2013.
2. C. D. Bellicoso, F. Jenelten, C. Gehring, and M. Hutter, "Dynamic locomotion through online nonlinear motion optimization quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2261–2268, 2018.
3. J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, "Dynamic locomotion in the mit cheetah 3 through convex model-predictive control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 7440–7447.
4. S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, and R. Tedrake, "Optimization-based Locomotion Planning, Estimation, and Control Design for the Atlas Humanoid Robot," *Autonomous Robots*, vol. 40, no. 3, pp. 429–455, 2016.
5. A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, "Gait and Trajectory Optimization for Legged Systems through Phase-based End-Effector Parameterization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1560–1567, 2018.
6. V. Barasuol, M. Camurri, S. Bazeille, D. G. Caldwell, and C. Semini, "Reactive trotting with foot placement corrections through visual pattern classification," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5734–5741.
7. O. A. Villarreal-Magaña, V. Barasuol, M. Camurri, L. Franceschi, M. Focchi, M. Pontil, D. G. Caldwell, and C. Semini, "Fast and Continuous Foothold Adaptation for Dynamic Locomotion Through CNNs," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2140–2147, 2019.
8. D. Belter, J. Bednarek, H. Lin, G. Xin, and M. Mistry, "Single-shot Foothold Selection and Constraint Evaluation for Quadruped Locomotion," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7441–7447.
9. L. Chen, S. Ye, C. Sun, A. Zhang, G. Deng, T. Liao, and J. Sun, "CNNs based Foothold Selection for Energy-Efficient Quadruped Locomotion over Rough Terrains," in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2019, pp. 1115–1120.