

Adaptive Traversability of Partially Occluded Obstacles

Karel Zimmermann^{1,2}, Petr Zuzánek¹, Michal Reinstein^{1,2}, Tomáš Petříček^{1,2}, Václav Hlaváč²

Abstract—Controlling mobile robots with complex articulated parts and hence many degrees of freedom generates high cognitive load on the operator, especially under demanding conditions such as in Urban Search & Rescue missions. We propose a solution based on reinforcement learning in order to accommodate the robot morphology automatically to the terrain and the obstacles it traverses. In this paper, we concentrate on the crucial issue of predicting rewards from incomplete or missing data. For this purpose we exploit the Gaussian processes as a predictor combined with decision trees. We demonstrate our achievements in a series of experiments on real data.

I. INTRODUCTION

Pushing the limits of the state-of-the-art in mobile robotics, especially when concerning the Urban Search & Rescue (USAR), nowadays requires not only close cooperation with the end-users, but also a thorough field-testing in the real environment. The motivation to our research comes from the experience of confronting our approach to adaptive traversability (AT) [1] with such a real environment. We build on our previous results and on our definition of the AT as a means of *autonomous motion control adapting the robot morphology (configuration of flippers and their compliance) to traverse an unknown complex terrain with obstacles in an optimal way*. Complex mobile platforms usually have a large number of degrees of freedom for control, causing inevitably a high level of undesired cognitive load of the operator.

In [1] we proposed a solution based on reinforcement learning, which lead to significantly lower cognitive load of the operator and we improved the overall performance of semi-autonomous traversability over complex obstacles. However, while exploiting multiple sensor modalities in natural conditions (such as forest or collapsed buildings), we often encountered the issue of missing or incomplete data that essentially spoiled our predictors of discounted rewards—the crucial part of our reinforcement learning framework. After exploring some straightforward approaches—such as least squares fit to the incomplete terrain profile—and founding them insufficient, we realized that a more robust and accurate solution has to be found.

To conclude our main contributions, in this paper, we demonstrate a significant improvement of our adaptive

The authors were supported by the Czech Science Foundation project GA14-13876S, by the SGS15/081/OHK3/1T/13, and by the project EU-FP7-ICT-609763 TRADR

¹ Authors are with the Faculty of Electrical Engineering, Czech Technical University in Prague, {zimmerk, petritol}@cmp.felk.cvut.cz, {zuzanpet, reinstein.michal}@fel.cvut.cz

² Authors are with the Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, hlavac@ciirc.cvut.cz

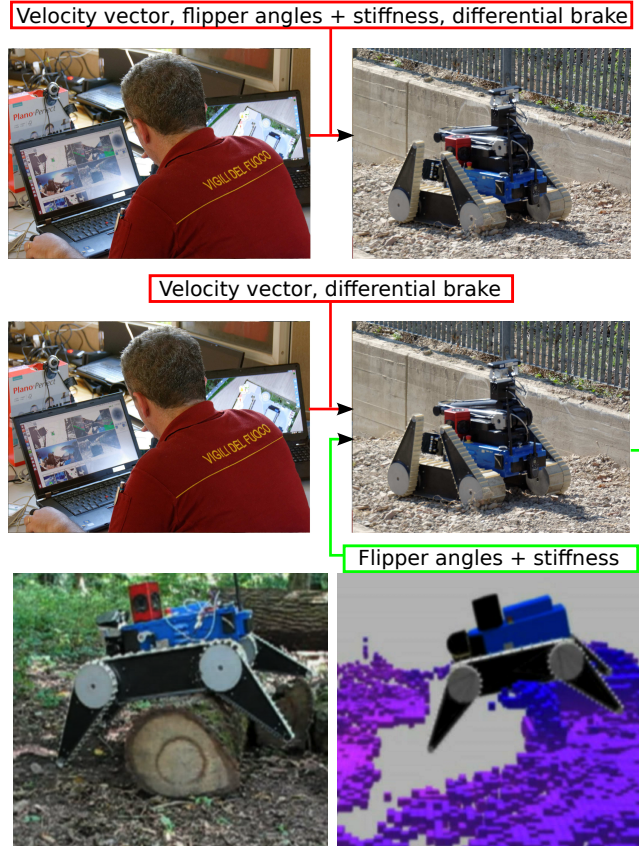


Fig. 1. Traversability with incomplete data: (top) high cognitive load of the operator during full teleoperation, (middle) low cognitive load of the operator with autonomously controlled flippers, (bottom) traversing obstacle with incomplete data.

traversability by (i) proposing a better means of predicting the discounted rewards using the Gaussian processes and (ii) proposing a reliable way for completing the incomplete data using decision trees. Our proposed solution now offers a number of practical advantages. First, the mobile robot can now operate in harsh environment, where both the proprioceptive data (inertial data from the Inertial Measurement Unit (IMU), configuration of the robot morphology, currents through the main tracks and subtracks—the flippers) and exteroceptive data (provided by the rotating laser range-finder) may be missing or incomplete. This occurs usually in case of reflective surfaces such as water, in case of occluded view, presence of smoke, or deformable terrain—conditions common for USAR. Second, due to the Gaussian processes, which are an efficient solution in the context of reinforcement learning for control [2], we were able to integrate a means

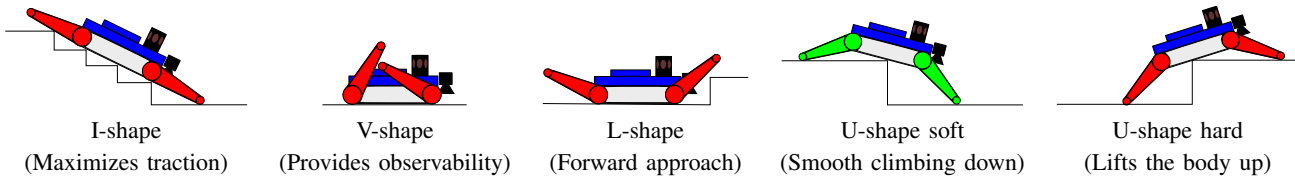


Fig. 2. Flipper configurations. Five different flipper configurations are distinguished. Red denotes low flipper compliance, green denotes high flipper compliance.

of probabilistic estimation of robot safety. This actually allows us to integrate call for the operator in potentially dangerous situations. Such supervision is crucial in order to push our framework forward towards the autonomous data collection and training. This is actually necessary for further improvement in the future, since it bypasses the need for any manual data annotations but with a sufficient guarantee of robot safety.

The structure of this paper is as follows: Section II covers the related work. Section III provides theoretical background to our approach. Section IV describes both the qualitative and quantitative experimental evaluation and Section V concludes the implications of our work.

II. RELATED WORK

Many authors estimate terrain traversability only from exteroceptive measurements (e.g. laser scans). For example Kim et al. [3] estimate whether the terrain is traversable or not and plan the trajectory over the traversable terrain. Colas et al. [4] distinguish four discrete flipper configurations; the traversability map then consists of list of feasible robot configuration along the trajectory. There are also alternative solutions based on kinematic models [5], [6] or on learning of a direct mapping between terrain features and robot actions [7], [8].

In our experience, when the robot is teleoperated in the real environment, it is not possible to plan the flipper motion in advance only from the exteroceptive measurements. The reasons are three-fold: (i) it is not known in advance, where the operator will lead the robot, (ii) exteroceptive measurements are usually partially occluded; (iii) robot-terrain interaction in a real environment cannot be inferred only from exteroceptive measurements, because the robot can for example slip or terrain can deform. Ho et al. [9] propose to predict the terrain deformation from the exteroceptive measurements. However, we argue that when the terrain collapses unexpectedly, captured terrain profile must be updated without exteroceptive measurements. Hence, rather reactive control based on all available measurements is needed.

Reactive control has been successfully used for learning the acrobatic tricks with an RC helicopter [10], [11]. Since it is possible to model the helicopter-air interactions well, one can use the model to improve the reinforcement learning. In our case, analytical modeling of the robot-terrain interaction is very difficult. On contrary to [10] we rather focus on a model-free reinforcement learning technique.

The idea of inference from incomplete data via some kind of sampling [12], [13] or EM algorithm [14] has been known

for several decades. In contrast to others, we demonstrate the proposed approach on a real robotic platform equipped with many different sensors.

III. THEORY

A. Reinforcement Learning Framework

For the sake of completeness, we briefly summarize our adaptive traversability (AT) method, see [1] for more details. We solve the AT problem for a tracked robot equipped with four flippers, lockable differential, rotating 2D laser scanner (SICK LMS-151) mounted in front of the robot, an IMU and GPS, see Fig. 1. We assume that the speed and the heading of the robot is controlled by the operator, and hence the AT is used to control the configuration of the four flippers and their compliance. Compliance of flippers is obtained by limiting the maximum allowed current in flipper drives.

To simplify this 8-dimensional task, we defined five discrete flipper modes specifying the angle and the compliance for all four flippers. The task is to switch between these flipper modes (denoted by $c \in \mathbb{C} = \{1 \dots 5\}$) in order to collect maximum sum of *rewards* while traversing the obstacle. We define the reward function $r(c, \mathbf{x}) : (\mathbb{C} \times \mathbb{R}^n) \rightarrow \mathbb{R}$, which assigns a real-valued reward for achieving state \mathbf{x} (defined in the following paragraph) while using mode c . It is expressed as a weighted sum of (i) user-denoted penalty (reward) specifying that the state is (not) dangerous, (ii) high pitch/roll angle penalty (preventing robot's flip-over), (iii) penalty for excessively switching the modes, (iv) robot forward speed reward (for making progress in traversing), and (v) the motion roughness (smoothness) penalty (reward) measured by accelerometers.

We represent the mutual state of the robot and the local neighboring terrain as n -dimensional feature vector $\mathbf{x} \in \mathbb{R}^n$ which consists of exteroceptive and proprioceptive features.

Exteroceptive features: We merge online individual scans into a point cloud 3D map using the ICP algorithm [15]. A point cloud map in the local neighborhood of the robot is further transformed into the Digital Elevation Map (DEM), see Fig. 3, capturing the local spatial representation of the terrain. Heights in bins are directly used as exteroceptive features.

Proprioceptive features: These consist of the robot speed (both actual and requested by the operator), the pose (pitch, roll), the flipper angles, the compliance thresholds and the actual flipper configuration.

We further define function $Q(c, \mathbf{x}) : (\mathbb{C} \times \mathbb{R}^n) \rightarrow \mathbb{R}$, which estimates expected sum of discounted rewards, when the robot is in state \mathbf{x} and flippers are set to mode c and the

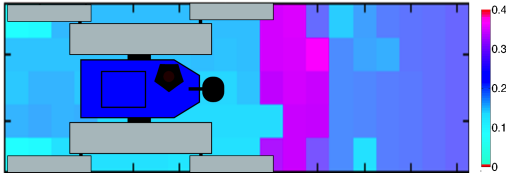


Fig. 3. An example of the Digital Elevation Map (DEM) together with the robot. Color encodes terrain heights in particular bins. The heights in the respective locations are directly used as state description features.

robot will be controlled optimally from the following state onwards. Such function allows for the following recursive definition:

$$Q(c, \mathbf{x}) = \sum_{\mathbf{x}'} p(\mathbf{x}'|c, \mathbf{x}) \left[r(c, \mathbf{x}) + \gamma \max_{c'} Q(c', \mathbf{x}') \right]$$

where $p(\mathbf{x}'|c, \mathbf{x})$ is the transition probability that the robot in state \mathbf{x} with flippers set to mode c will get to the following state \mathbf{x}' . Discount factor $\gamma \in [0, 1]$ is used to reduce the influence of distant future rewards. The Q-function is modeled by the Gaussian processes learned on training sets collected over several episodes by a standard Q-learning method [16]. Once the Q-function is known, the optimal flipper mode c^* for the robot in the state \mathbf{x} is chosen as follows:

$$c^* = \arg \max_c Q(c, \mathbf{x}).$$

B. Gaussian Processes

Gaussian processes (GPs) [2] are the extension of multivariate Gaussians to infinite-size collections of real valued variables. In general, GPs can be understood as distributions over random functions. Therefore, instead of describing the distribution by a mean vector and covariance matrix, in GPs we define a mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}, \mathbf{x}')$ —the kernel. The kernel captures the correlation between the function values of any input pair $(\mathbf{x}, \mathbf{x}')$, leading to the definition:

$$f(\mathbf{x}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')).$$

Since the essential part of the learning is given by the kernel function, the mean function is often set to zero, as it is in our case. After empirically evaluating a number of kernel functions used in common practice, in our implementation, we exploit the rational quadratic function. We modify it for the purpose of *automatic relevance detection*, which can be interpreted as embedded feature selection performed automatically when optimizing over the parameters of the kernel (to maximize the likelihood):

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \left(1 + \frac{1}{2\alpha} ((\mathbf{x} - \mathbf{x}')^T \mathbf{\Lambda}^{-2} (\mathbf{x} - \mathbf{x}')) \right)^{-\alpha}$$

where σ_f^2 and α are the kernel parameters, and $\mathbf{\Lambda}$ is a diagonal matrix containing weights to perform the automatic relevance detection.

Since GP is a stochastic process such that any finite subset of random variables has a multivariate Gaussian distribution,

having a finite training set, we can easily use GPs for regression of discounted sum of rewards; i.e. predicting both mean and variance of the testing data.

C. AT with Incomplete Data

The proposed AT is based on a model-free reinforcement learning technique with the action-value Q-function learned on a training set collected over several episodes. Since autonomous collection of training samples may cause destruction of the robot, expensive manual operation and annotation of the data is needed and the resulting training set is rather limited. Since we observed that such a limited training set does not capture all the mutual dependencies which we need to reconstruct the missing data, we learn the model for reconstruction of missing data from a significantly larger training set, captured almost autonomously and without any manual annotations. We also add synthetically generated partial occlusions of the DEM. As a result, huge training set consisting from hundreds of thousands of training samples is obtained.

To tackle the training from such a huge training set in a reasonable time, we model conditional probability distributions for each feature separately by the regression forest [17]. It is known that decision trees provide good performance when huge training set is available [18] while allowing for online learning (meaning that the training time is linear in the size of the training set).

Let us denote n -dimensional vector of features $\mathbf{x} = (x_1 \dots x_n)^T$. Conditional probability $p(x_i|x_J)$, $J = \{1 \dots n\} \setminus i$ of i -th feature is represented by 24 binary regression trees¹ with stump decision rules. To train the tree, we are given training set consisting of m training samples $\mathbf{x}^1, \dots, \mathbf{x}^m$.

Each tree is learned by the greedy algorithm which selects the splitting variable $j \in J$ and split point s that minimize the weighted variance in the left and right sub-tree in each node as follows:

$$\arg \min_{(j,s)} |R_1| \cdot \text{var}_{k \in R_1(s,j)}(x_i^k) + |R_2| \cdot \text{var}_{k \in R_2(s,j)}(x_i^k),$$

where $R_1(s, j) = \{k \mid x_j^k \leq s\}$ is the set of indices going to the left sub-tree, $R_2(s, j) = \{k \mid x_j^k > s\}$ is the set of indices descending into the right sub-tree and $|R_1|$, $|R_2|$ denotes number of samples descending into the left and right sub-tree, respectively. Especially, if the value of the splitting feature is unknown (e.g. occluded), then the training sample descends into both sub-trees. The leaves of the regression tree contain discretized conditional probability $p(x_i|x_J)$ estimated on training data. Conditional probability of the feature is computed as the mean conditional probability over all the leaves in the forest reached by the respective sample. To reconstruct the missing data, we perform the Gibbs sampling [19] from conditional probability models of missing features to obtain samples from their joint probability. These samples are further used (i) to reconstruct the

¹The number of trees was chosen to be suitable for the parallelization on our grid.

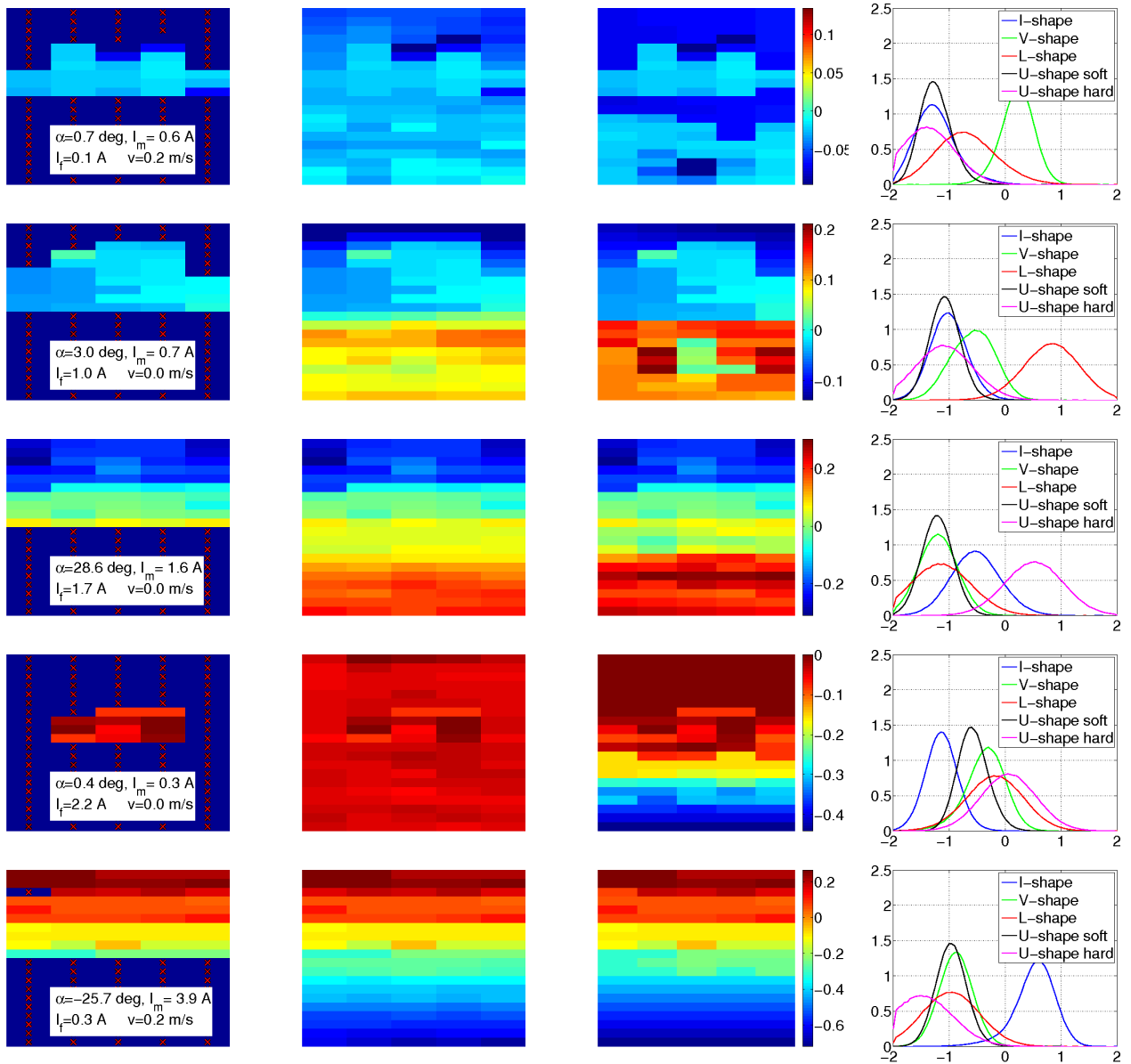


Fig. 4. Qualitative evaluation. See Section IV-A for corresponding description. Size of bins in the DEM is 10 cm \times 10 cm.

missing data (e.g. for the reconstruction of training data or for the visualization purposes) and (ii) to estimate probability distribution function of the Q-function (we will further refer to it as QPDF). To speed up the burn-in we replace the random initialization of the missing features by the one based on conditional probabilities. We initialize each missing feature by estimating its conditional probability given all other features. When other missing features are needed we send the sample to both sub-trees, the conditional probability is then estimated as the weighted average over all leaves it reached. The missing feature is initialized by the value maximizing the conditional probability.

The reconstruction of the missing data is obtained as the mean of generated samples, since the reliable estimate of the high dimensional joint probability of all missing data requires

a lot of training samples. QPDF is computed for each flipper configuration separately as the mean of Gaussian distributions estimated by the Gaussian processes corresponding to the Q-function of the particular configuration.²

In the teleoperated mode, we choose only from configurations achieving positive discounted rewards with probability higher than 80%. If more configurations exist, we chose the one which yields the highest average sum of discounted

²Note, that we could alternatively also generate samples from the joint probability of missing features from the Gaussian process directly, however, our tree based conditional probabilities are trained on significantly larger unannotated training set, with missing features which allows to capture non-Gaussian dependencies among missing features. Please also note, that in our case, existence of a unique joint probability of all missing features consistent with all conditional probabilities is not guaranteed, however, plausible results are achieved in our experiments.

rewards. If none of the configurations satisfies the safety condition, the robot is stopped and the operator takes over the flipper control.

IV. EXPERIMENTAL EVALUATION

Experiments are organized as follows: Section IV-A provides qualitative evaluation by showing examples with missing data, their reconstructions, and probability distribution of Q-functions (QPDF). Section IV-B presents quantitative evaluation of the robustness to missing features. Section IV-C evaluates the reconstruction error of the missing data.

A. Qualitative Evaluation

Fig. 4 shows some typical results including the one with incorrect reconstruction and inability to control the flippers autonomously. Rows correspond to different situations. The first column shows the input data; the missing heights in the DEM are outlined by red X in a blue rectangle, pitch is denoted by α , mean absolute current over both main tracks is denoted I_m , mean absolute current in the engines lifting the front flippers is denoted by I_f . The second column corresponds to our mean reconstruction of missing heights. The third column shows the ground-truth data of the terrain profiles measured by our laser scanner under ideal observation conditions. Scale is given in meters. The last column shows the probability distribution of the Q-function for five different flipper configurations. The horizontal axis corresponds to the discounted sum of rewards (the higher the better), vertical axis contains probability distribution function (i.e. integral over each function is equal to one) of the discounted sum of rewards (i.e. the Q-values); we will refer to this graph as QPDF.

The first row depicts the robot on a flat terrain: currents are low, since there is no obstacle which the robot could bounce into; pitch is also close to zero. The reconstruction of the missing flat terrain is correct. The most suitable configuration is the V-shape. Of course, in this particular case all configuration are possible, but we trained the Q-learning to use this configuration exclusively on the flat terrain since it provides the best observation conditions (i.e. flippers do not cause occlusions in the scene sensed by the laser and the omnidirectional camera).

The second row shows the robot in front of an obstacle. Pitch is still close to zero and the obstacle is not visible in the input DEM. Nevertheless, the reconstruction is correct, since the robot senses the obstacle using the front flippers. QPDF shows that the only suitable configuration is the L-shape, which is also correct since

The third row corresponds to the state in which the robot's pitch is almost 30° and the robot is lifting its body up on the top of the obstacle. The reasonable mode is the U-shape-hard since we need rear flippers to support the whole robot body. Reconstruction of the obstacle is also correct.

The fourth row demonstrates the inability to reconstruct the step down in front of the robot. The input is only a small piece of flat terrain with zero pitch, however the robot is still in the U-shape-hard configuration causing high current in the

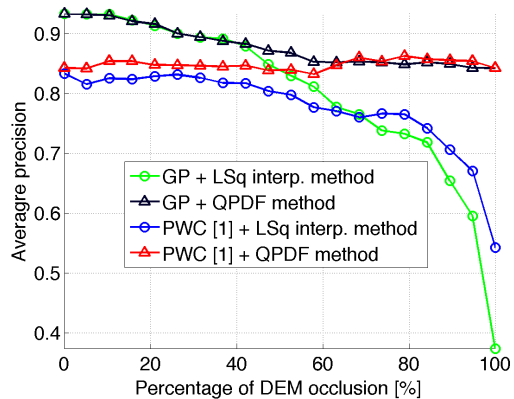


Fig. 5. Robustness to DEM occlusion. The experiment shows that the proposed QPDF marginalization preserves high precision even for 100% occlusion (i.e. fully occluded DEM).

front and rear flippers. One possible reconstruction is a flat terrain, but the U-shape-hard is prohibited configuration for such terrain and there is no permitted way by which the robot could get into such state. Another possibility is step-down or step-up. QPDF reveals that none of the configurations is sufficiently safe. This is the typical case in which the operator has to make a safe decision.

The fifth row shows climbing down the stairs with pitch almost -30° . The only suitable configuration is the I-shape, which maximizes the friction.

B. Quantitative Evaluation

In this experiment we quantitatively evaluate robustness to DEM occlusions of (i) the proposed QPDF method (denoted by triangles) and (ii) the Least Squares (LSq) interpolation method (denoted by circles). We also compare the Gaussian process Q-function (GP) proposed in this work with the piece-wise constant Q-function (PWC) proposed in [1].

The QPDF method first computes the QPDF distributions (see Section III-C for details) and choose the configuration guaranteeing maximal mean discounted sum of rewards. The LSq interpolation method first interpolates the missing data, then it computes the response of the Q-function on the interpolated data and chooses the configuration with the highest response. Each method has its own Q-function trained on the data reconstructed by the corresponding reconstruction method.

The robustness to DEM occlusions is demonstrated by synthetically occluding the DEM from the front of the robot. For each occlusion, the average precision is computed as the "number of states in which the method selects permitted flipper configuration" divided by "the total number of testing states". Fig. 5 shows superiority of the QPDF method over the LSq interpolation method. In particular, we see that up to 40% of DEM occlusion, the LSq and QPDF methods behave comparably. The reasons are two-fold: (i) the part of the occluded DEM is far in front of the robot and there is no way to sense it from the proprioceptive measurements, (ii) the obstacle hidden in this part of DEM is usually far enough,

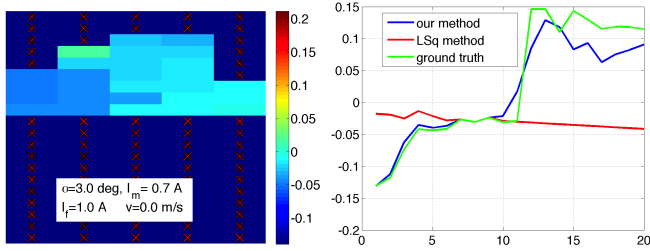


Fig. 6. Reconstruction of the Digital Elevation Map: (left) input data, with red crosses denoting the missing data, (right) the profile of the reconstructed and ground-truth terrain. Size of bins in the DEM is 10 cm \times 10 cm, terrain height in particular bins is measured in meters. Unit of the horizontal axis of the right figure is DEM bin.

therefore the V-shape configuration (the one for flat terrain) is still allowed in most of the testing data. When more than 40% are hidden, performance of the LSq interpolation method drops rapidly down towards 0.4 – 0.5 precision (i.e. 40% – 50% of states in which the permitted configuration is selected) for both GP and PWC, while the QPDF method preserves the high precision.

C. Reconstruction Error

Finally, we present the reconstruction error of our method and LSq interpolation method for different DEM occlusions, see Table I. When front 25% of the DEM is occluded, we can see that reconstruction errors are the same, since the terrain is too far from the robot to be reflected in proprioceptive measurements. When the occlusion is larger, our method obviously dominates. We do not evaluate the reconstruction error of the LSq method for the full occlusion, since if the whole DEM is occluded, there are no data to interpolate from. Fig. 6 presents a typical case in which our method reconstructs the stair-like terrain shape correctly while the LSq method fails.

TABLE I
RECONSTRUCTION ERROR

occlusion	25%	50%	75%	90%	100%
our method	2.8 cm	5.5 cm	6.7 cm	7.8 cm	8.7 cm
LSq method	2.8 cm	9.2 cm	24.7 cm	94.7 cm	-

V. CONCLUSIONS

We have proposed a reinforcement learning based method for the adaptive traversability and extended our original contribution to handle incomplete or missing data coming from partially occluded obstacles. We proved that it is possible to reconstruct the missing exteroceptive data from the proprioceptive data, and preserve reasonable precision even for full occlusion of the digital elevation map (DEM). Furthermore, we have also shown how to estimate the safety of our proposed autonomous control method and for dangerous cases then execute a call for the operator to avoid damage to the platform. In our approach we greatly benefit from the incorporation of the regression trees, which

model the probability distribution of missing data and do not require any manual annotations—in contrary to learning the Q-function. Therefore, a significantly larger training set can be provided. For the actual prediction of rewards, we exploited the Gaussian processes framework that proved to be superior to standard piecewise constant regression.

REFERENCES

- [1] K. Zimmermann, P. Zuzánek, M. Reinstein, and V. Hlaváč, “Adaptive traversability of unknown complex terrain with obstacles for mobile robots,” in *IEEE International Conference on Robotics and Automation*, 2014, pp. 5177–5182.
- [2] M. Deisenroth, D. Fox, and C. Rasmussen, “Gaussian processes for data-efficient learning in robotics and control,” *Transaction on Pattern Analysis Machine Intelligence*, 2009, to be published.
- [3] D. Kim, J. Sun, S. Min, O. James, M. Rehg, and A. F. Bobick, “Traversability classification using unsupervised on-line visual learning for outdoor robot navigation,” in *In Proc. of International Conference on Robotics and Automation (ICRA)*, 2006, pp. 518–525.
- [4] F. Colas, S. Mahesh, F. Pomerleau, M. Liu, and R. Siegwart, “3d path planning and execution for search and rescue ground robots,” in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, Nov 2013, pp. 722–727.
- [5] Y. Okada, K. Nagatani, K. Yoshida, S. Tadokoro, T. Yoshida, and E. Koyanagi, “Shared autonomy system for tracked vehicles on rough terrain based on continuous three-dimensional terrain scanning,” *J. Field Robot.*, vol. 28, no. 6, pp. 875–893, 2011. [Online]. Available: <http://dx.doi.org/10.1002/rob.20416>
- [6] K. Ohno, S. Morimura, S. Tadokoro, E. Koyanagi, and T. Yoshida, “Semi-autonomous control system of rescue crawler robot having flippers for getting over unknown-steps,” in *Intelligent Robots and Systems (IROS), 2007. IEEE/RSJ Int. Conf. on*, 2007, pp. 3012–3018.
- [7] R. Sheh, B. Hengst, and C. Sammut, “Behavioural cloning for driving robots over rough terrain,” in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011, pp. 732–737.
- [8] E. Uğur and E. Şahin, “Traversability: A case study for learning and perceiving affordances in robots,” *Adaptive Behavior*, vol. 18, no. 3–4, pp. 258–284, 2010.
- [9] K. Ho, T. Peynot, and S. Sukkarieh, “A near-to-far non-parametric learning approach for estimating traversability in deformable terrain,” in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, Nov 2013, pp. 2827–2833.
- [10] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, “An application of reinforcement learning to aerobatic helicopter flight,” in *In Advances in Neural Information Processing Systems 19*. MIT Press, 2007, p. 2007.
- [11] P. Abbeel and A. Y. Ng, “Exploration and apprenticeship learning in reinforcement learning,” in *Proc. of the 22nd int. conf. on Machine learning*, ser. ICML ’05. New York, NY, USA: ACM, 2005, pp. 1–8.
- [12] D. J. Lizotte, L. Gunter, E. Laber, and S. A. Murphy, “Missing data and uncertainty in batch reinforcement learning,” in *Neural Information Processing Systems (NIPS)*, 2008.
- [13] M. A. Tanner and W. Wong, “The calculation of posterior distributions by data augmentation,” *Journal of the American Statistical Association*, vol. 82, pp. 528–540, 1987.
- [14] Z. Ghahramani and M. I. Jordan, “Supervised learning from incomplete data via an em approach,” in *Neural Information Processing Systems (NIPS)*. Morgan Kaufmann, 1994, pp. 120–127.
- [15] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, “Comparing ICP Variants on Real-World Data Sets,” *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, Feb. 2013.
- [16] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.
- [17] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, ser. Springer series in statistics. Springer, 2001. [Online]. Available: <http://books.google.cz/books?id=VRzITwgNV2UC>
- [18] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, “Real-time human pose recognition in parts from single depth images,” in *CVPR*, 2011.
- [19] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.