Benchmarking Incremental Regressors in Traversal Cost Assessment

Miloš Prágr[0000−0002−8213−893X]* and Jan Faigl[0000−0002−6193−0792]

Department of Computer Science, Faculty of Electrical Engineering
Czech Technical University in Prague
Technická 2, 166 27, Prague 6, Czech Republic
{pragrmi1,faiglj}@fel.cvut.cz,
WWW home page: https://comrob.fel.cvut.cz/

Abstract. Motivated by the deployment of multi-legged walking robots in traversing various terrain types, we benchmark existing online and unsupervised incremental learning approaches in traversal cost prediction. The traversal cost is defined by the proprioceptive signal of the robot traversal stability that is combined with appearance and geometric properties of the traversed terrains to construct the traversal cost model incrementally. In the motivational deployment, such a model is instantaneously utilized to extrapolate the traversal cost for observed areas that have not yet been visited by the robot to avoid difficult terrains in motion planning. The examined approaches are Incremental Gaussian mixture network, Growing neural gas, Improved self-organizing incremental neural network, Locally weighted projection regression, and Bayesian committee machine with Gaussian process regressors. The performance is examined using a dataset of the various traversed terrains by a real hexapod walking robot. A part of the presented benchmarking is thus a description of the dataset and also a construction of the reference traversal cost model that is used for comparison of the evaluated regressors. The reference is designed as a compound Gaussian process-based model that is learned separately over the individual terrain types. Based on the evaluation results, the best performance among the examined regressors is provided by Incremental Gaussian mixture network, Improved self-organizing incremental neural network, and Locally weighted projection regression, while the latter two have the lower computational requirements.

Keywords: Terrain Characterization, Multi-Legged Walking Robot, Incremental Learning

1 Introduction

The addressed traversal cost assessment is motivated by the deployment of mobile robots in long-term autonomous missions, where robots have to plan their motion in the environment and identify hard to traverse areas [1,2]. Our particular interest in the traversal cost assessment problem stems from the deployment

of a hexapod walking robot (see Fig. 1), that can benefit from the ability to plan the motion of each leg [20] and thus traverse rough terrains. More specifically, in this paper, we focus on traversable areas since untraversable terrains such as walls, extreme slopes, or ravines can be easily recognized in a 3D terrain map build as a part of the robot localization [9]. Thus, we investigate approaches to improve the efficiency of the robot motion over traversable areas by learning models for prediction of the traversal cost based on the perceived terrain geometry and appearance features.

In our previous work [13,14], we have deployed the Incremental Gaussian Mixture Network Model (IGMN) [11] to learn the cost of transport [17]. However, the IGMN suffers from the quadratic time complexity regarding the input dimension. On the other hand, the recent deployments of the Gaussian Process (GP) regression in robotic applications such as occupancy mapping [10] can motivate for GP-based traversal cost model, but the deployment of GPs in incremental life-long setups is hampered by their cubic learning time complexity regarding the training set size. Therefore, there is a need for efficient unsupervised incremental learning approaches that would provide competitive results to the IGMN or GPs-based models, but would be computationally less demanding.

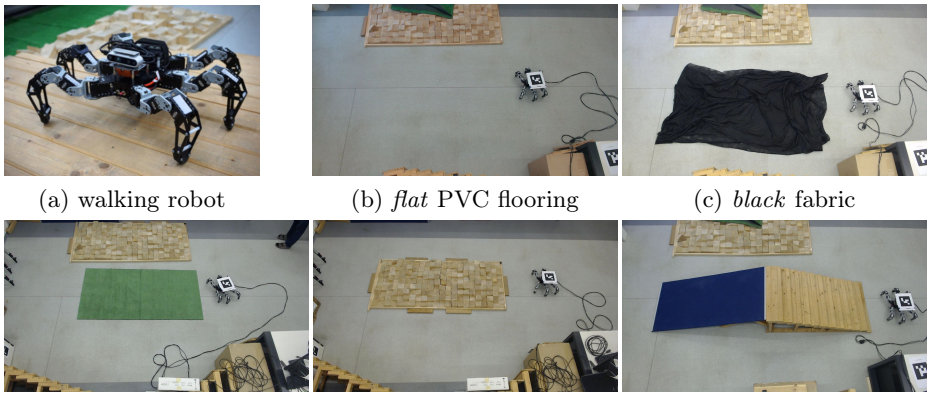
Our early results on incremental learning are reported in [4], and the herein presented work extends therein proposed evaluation, and we report on benchmarking of five incremental learning approaches for traversal cost prediction using geometry and appearance terrain descriptors. Five competing approaches include the Growing Neural Gas (GNG) [5], Improved Self-Organizing Incremental Neural Network (ISOINN) [15], Locally Weighted Projection Regression (LWPR) [18], IGMN [11], and an incrementally learned product of the GP regressor experts constructed using the Bayesian Committee Machine (BCM-GP) [16] which we have previously deployed in an exploration setup in [12]. All the regressors are evaluated using a real dataset that has been collected by a hexapod walking robot in a testing environment with seven terrain types. The performance of the incremental learners is benchmarked over areas observed but not necessarily traversed by the robot. Therefore, the traversal cost ground truth is not readily available, and the examined approaches are compared with a compound reference GP-based model that is prepared offline from the collected dataset for which an individual model is created for each particular terrain type.

The rest of the paper is organized as follows. Section 2 describes the used dataset of the real terrain traversal cost. The proposed benchmarking methodology for evaluation of the regressors in the addressed problem is presented in Section 3. The evaluation results together with the description of the learning approaches and their parametrizations are reported in Section 4. Finally, the paper is concluded in Section 5.

2 Terrain Traversal Dataset

The dataset for the benchmarking of the terrain traversal cost learning consists of learning data and testing data that both have been collected by the real hexapod

walking robot in a laboratory test site. The learning data, which are called trails, are sequences of terrain descriptors accompanied by the proprioceptive measure of the robot traversal cost over the terrains. The testing data are considered as terrain descriptors organized into grids that represent maps of seen environments for which the traversal cost is not available and has to be predicted.



(a) walking robot (b) *flat* PVC flooring (c) *black* fabric

(d) artificial *turf* (e) wooden *cubes* (f) *slope* terrains

Fig. 1. The (a) hexapod walking robot and some of the (b-f) traversed terrains. The *cubeblack* and *cubeturf* terrains are created by covering the (e) wooden cubes with the (c) black fabric and (d) artificial turf, respectively.

The used hexapod walking robot is shown in Fig. 1 together with the seven particular terrains denoted: *flat*, *slope*, *cubes*, *turf*, *black*, *cubeblack*, and *cubeturf*. The terrains have been set with the intention to confuse the learners. For example, using only appearance and geometric features, it is hard to distinguish whether the black fabric and artificial turf cover the cubes (*cubeblack* and *cubeturf*) or similarly the flat ground (*black* and *turf*, respectively). Moreover, the *slope* terrain consists of two distinct sloped areas that are both descended and ascended by the robot, and thus provide different robot experience.

The robot trails are sequences of terrain feature descriptors that are paired with traversal cost measurement. The terrain descriptor $\boldsymbol{d}_t = (s_1, s_2, s_3, a_1, a_2)$ is based on our previous work [13], and it comprises of a three-dimensional shape descriptor [7] and two-dimensional Lab space color descriptor. The features characterize 0.2 m radius around the robot at the particular location for which the traversal cost c is measured as the square root of the robot roll variance for a 10 s period. The traversal cost measurements are computed from inertial measurements sampled with the frequency 400 Hz. Overall, the robot terrain traversal experience is captured by the descriptor $\boldsymbol{d}_e = (s_1, s_2, s_3, a_1, a_2, c)$, where the traversal cost c is experienced over the area characterized by $\boldsymbol{d}_t$. Thus, each terrain descriptor is reported with the cost experienced over the period the robot is present at the described location.

Since some of the benchmarked approaches compute Euclidean distance over the feature space, both $\boldsymbol{d}_t$ and c are normalized to be roughly zero mean and unit variance to assure that each dimension is represented equally. Even though

the work is motivated by incremental life-long learning, where the variance of the incoming data is not known a priori, we leverage our expert knowledge of the data which are presumed to be distributed normally with $(\mu_s, \sigma_s) = (0.5, 0.2)$ for the shape features, $(\mu_a, \sigma_a) = (0.0, 10.0)$ for the appearance, and $(\mu_c, \sigma_c) = (0.02, 0.01)$ for the cost.

An individual terrain trail is collected for each particular terrain type, and therefore, the seven terrain trails are denoted $\mathcal{T}_{black}, \ldots, \mathcal{T}_{turf}$. Moreover, each terrain trail $\mathcal{T}_\mathrm{t}$ is divided into three equally length parts $\mathcal{T}_\mathrm{t} = (\mathcal{T}_\mathrm{t}^1, \mathcal{T}_\mathrm{t}^2, \mathcal{T}_\mathrm{t}^3)$, where t stands for a particular terrain type. The lengths of the individual terrain trails range from 202 descriptors in *cubeturf* to 824 in *slope*, and there are 3522 descriptors among all the trails in the total. Besides the terrain trails, four full-length (with all terrain types) trails are created that simulate the traversal of all the available terrains. The full-length trails are constructed as different orderings of the terrain trails as follows

$$\begin{aligned}
\mathcal{T}_1 &= (\mathcal{T}_{black}, \mathcal{T}_{cubeblack}, \mathcal{T}_{cubes}, \mathcal{T}_{cubeturf}, \mathcal{T}_{flat}, \mathcal{T}_{slope}, \mathcal{T}_{turf}),\\
\mathcal{T}_2 &= (\mathcal{T}_{flat}, \mathcal{T}_{black}, \mathcal{T}_{turf}, \mathcal{T}_{cubes}, \mathcal{T}_{cubeblack}, \mathcal{T}_{cubeturf}, \mathcal{T}_{slope}),\\
\mathcal{T}_3 &= (\mathcal{T}_{flat}, \mathcal{T}_{black}, \mathcal{T}_{cubeblack}, \mathcal{T}_{turf}, \mathcal{T}_{cubeturf}, \mathcal{T}_{cubes}, \mathcal{T}_{slope}),\\
\mathcal{T}_4 &= (\mathcal{T}_{black}^1, \mathcal{T}_{cubeblack}^1, \mathcal{T}_{cubes}^1, \mathcal{T}_{cubeturf}^1, \mathcal{T}_{flat}^1, \mathcal{T}_{slope}^1, \mathcal{T}_{turf}^1, \mathcal{T}_{black}^2, ..., \mathcal{T}_{turf}^3).
\end{aligned} \tag{1}$$

The testing data are terrain descriptors organized into grid maps with the size of the squared cell 0.1 m. The individual grids are denoted $\mathcal{G}_{black}, \ldots, \mathcal{G}_{turf}$. All seven individual terrain grids are merged into a single grid $\mathcal{G}_{merged}$ for evaluation of the regressors on the whole testing dataset.

3 Benchmark Methodology

The addressed problem is to construct the terrain traversal cost model using a set of experience descriptors $\boldsymbol{d}_e$, and instantaneously use the model for predicting the traversal cost c using the terrain descriptor $\boldsymbol{d}_t$ determined from the available model of the environment. In the motivational deployment scenario, the model is constructed incrementally as new data about the terrain traversability are collected during the robot movement over the terrain, and thus the model is sequentially learned from descriptors of the trail $\mathcal{T}$. Thus the model $\mathcal{M}(\mathcal{T}, k)$ at the learning step k incorporates the k-th observation $\boldsymbol{d}_e^k$ from the trail $\mathcal{T}$, i.e., $\boldsymbol{d}_e^k \in \mathcal{T}$, and the learning process can be defined as the iterative update

$$\mathcal{M}(\mathcal{T}, k) \leftarrow update(\mathcal{M}(\mathcal{T}, k-1), \boldsymbol{d}_e^k | \boldsymbol{d}_e^k \in \mathcal{T}), \tag{2}$$

where the initial model $\mathcal{M}(\mathcal{T}, 0)$ carries no information for any trail $\mathcal{T}$.

The prediction of each examined regressor is evaluated over the terrain descriptor grid maps. However, the robot has not traversed all the represented areas, and thus the ground truth traversal cost is not available for the grid maps. Therefore, the quality of the individual models is evaluated using a reference model $\mathcal{M}_\mathrm{ref}$, and we follow the evaluation based on the correctness ratio [4].

The cost prediction $pred(\mathcal{M}, \boldsymbol{d}_t)$ of the model $\mathcal{M}$ for the terrain descriptor $\boldsymbol{d}_t$ is considered correct if it corresponds to 95 % confidence interval of the reference model $\mathcal{M}_{\text{ref}}$, and the correctness $correct(\mathcal{M}, \mathcal{M}_{\text{ref}}, \boldsymbol{d}_t)$ is defined as

$$correct(\mathcal{M}, \mathcal{M}_{\text{ref}}, \boldsymbol{d}_t) = \begin{cases} 1 & \text{if} \quad |pred(\mathcal{M}, \boldsymbol{d}_t) - \mu(\mathcal{M}_{\text{ref}}, \boldsymbol{d}_t)| < 2\sigma(\mathcal{M}_{\text{ref}}, \boldsymbol{d}_t) \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

where $\mu(\mathcal{M}_{\text{ref}}, \boldsymbol{d}_t)$ and $\sigma(\mathcal{M}_{\text{ref}}, \boldsymbol{d}_t)$ are the predictive mean and square root of the predictive variance of the reference $\mathcal{M}_{\text{ref}}$. The predicted cost is assumed to be a random variable that is modeled by the mean and variance, and therefore, the reference model is based on the individual GP for each particular terrain type. A particular reference model for the dataset described in Section 2 is thus a compound model of seven GPs.

The prediction of each examined regressor is evaluated over the terrain descriptor grid maps. In particular, each terrain grid map $\mathcal{G}_{\text{t}}$ is associated with the reference GP-based model $\mathcal{M}_{\mathcal{G}_{\text{t}}}$, which is learned using the particular terrain type trail $\mathcal{T}_{\text{t}}$. Although the individual terrain trails are ordered differently in the four full-length trails $\mathcal{T}_1, ..., \mathcal{T}_4$, it does not affect the reference models, because the reference models are not learned incrementally.

The prediction of the model $\mathcal{M}$ over $\mathcal{G}$ is quantified by the correctness ratio $\mathcal{R}$

$$\mathcal{R}(\mathcal{M}, \mathcal{G}) = \frac{\sum_{\boldsymbol{d}_t \in \mathcal{G}} correct(\mathcal{M}, \mathcal{M}_{\mathcal{G}}, \boldsymbol{d}_t)}{|\mathcal{G}|}, \tag{4}$$

where $|\mathcal{G}|$ is the number of descriptors in the grid $\mathcal{G}$.

4 Evaluation Results

Five incremental learning algorithms are benchmarked for the terrain traversal cost modeling with the hexapod walking robot. Namely, we use the Incremental Gaussian Mixture Network Model (IGMN) [11], the Robust Bayesian Committee Machine with Gaussian Process Regressors (BCM-GP) [3], the Growing Neural Gas (GNG) [5], the Improved Self-Organizing Incremental Neural Network (ISOINN) [15], and the Locally Weighted Projection Regression (LWPR) [18]. In the rest of this section, the used parametrizations of the individual models are described, and we report on the performance of the models in terms of their correctness ratios and interpret their traversal cost predictions.

4.1 Parametrization of the Examined Traversal Cost Models

Except for the LWPR, which performs better when tuned manually, the parametrizations of the individual models are selected by maximizing the correctness ratio $R(\mathcal{M}(\mathcal{T}_1), \mathcal{G}_{merged})$ evaluated on the merged grid map $\mathcal{G}_{merged}$ using a grid search. Therefore, the selected parametrizations may not suit the other trails perfectly; however, we consider this transfer of parametrization to be a part of the evaluation of the incremental learners.

The IGMN is an incremental approximation of the EM algorithm which learns a set of Gaussian components. The algorithm is parameterized with the grace period $v_{\min} = 5$, scaling factor $\delta = 0.1$, and is allowed up to 100 components. The minimal accumulated posterior is fixed to $sp_{\min} = 3$ to enforce that any $sp_{\min}$ and $v_{\min}$ combination allows adding new components. The IGMN has been implemented in Python as all the other algorithms.

The framework [19] implemented in Python has been used for the GNG [5] and ISOINN [15] algorithms. The GNG is parameterized with the learning step $\lambda = 10$ and maximal age $a_{\max} = 10$. The framework utilizes Gaussian kernel smoother parameterized with $K = 1000$ smoothing neurons and smooth parameter $smooth = 0$. The ISOINN is parameterized with the learning step $\lambda = 10$ and maximal age $a_{\max} = 1000$, and it also utilizes smoother with $K = 1000$, but with the smooth parameter $smooth = -0.75$. It is worth noting that $a_{\max} = 1000$ effectively inhibits edge deletion given the lengths of the trails.

Python bindings of the LWPR implementation [8] are used to compute the LWPR models. In the best performing parametrization on $R(\mathcal{M}(\mathcal{T}_1), \mathcal{G}_{merged})$, the initial distance metric is set as $\mathbf{D}_{\text{init}} = 0.1\mathbf{I}$, where $\mathbf{I}$ is an identity matrix, and the distance metric learning rate is $\alpha_{\text{init}} = 1000$. However, such parametrization is prone to overfitting when learning on other trails, and exhibits poor performance overall. Therefore, we further report on results for the manually tuned parametrization $\mathbf{D}_{\text{init}} = 10\mathbf{I}$ and $\alpha_{\text{init}} = 10$ that is denoted LWPR$^+$.

The BCM-GP [16] learns Gaussian Process regressor experts, which are combined using the Robust Bayesian committee machine. Experts are constructed incrementally one at a time, and after the expert is constructed, it is not further modified. The BCM-GP is parameterized with the maximal expert size of 25 observations and uses the Matern $\frac{3}{2}$ kernel. The GPy toolbox [6] is used to learn the individual experts and the BCM-GP is also implemented in Python.

Finally, the reference GP-based models are learned using the GPy toolbox with the RBF kernel.

4.2 Correctness Ratios of the Examined Regressors

The correctness ratios R of the examined regressor on $\mathcal{G}_{merged}$ for four trails $\mathcal{T}_1, \ldots, \mathcal{T}_4$ are depicted in Table 1 and they range between approx. 55 % and 75 %. The IGMN and hand-tuned LWPR$^+$ appear to be the most stable incremental

Table 1. Correctness ratio R on the *merged* grid $\mathcal{G}_{merged}$ for different trails

Trail	BCM-GP	GNG	IGMN	ISOINN	LWPR	LWPR$^+$
$\mathcal{T}_1$	0.59	0.74	0.70	0.77	0.72	0.68
$\mathcal{T}_2$	0.62	0.57	0.71	0.59	0.48	0.69
$\mathcal{T}_3$	0.57	0.57	0.71	0.64	0.58	0.69
$\mathcal{T}_4$	0.72	0.63	0.67	0.69	0.70	0.66

regressors as their results are similar regardless of the trail ordering used for learning. The GNG and ISOINN appear to be much more affected by the trail

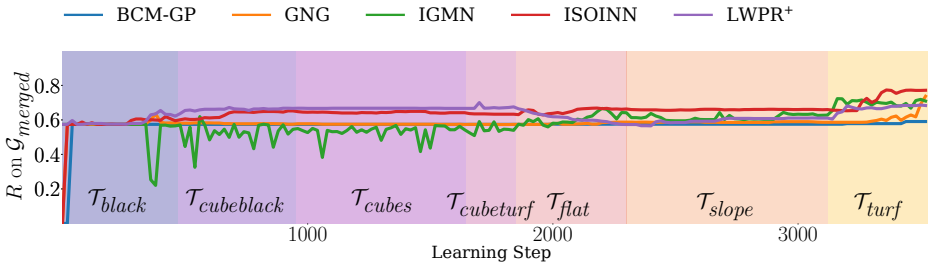


Fig. 2. The evolution of the correctness ratio R computed on $\mathcal{G}_{merged}$ for the incrementally learned traversal cost models on $\mathcal{T}_1$.

ordering and are outperformed by the IGMN for $\mathcal{T}_2$ and $\mathcal{T}_3$. On the other hand, the GNG and ISOINN outperform the IGMN for $\mathcal{T}_1$, and ISOINN also performs slightly better for $\mathcal{T}_4$. The best performance for $\mathcal{T}_4$ is provided by the BCM-GP, but its overall performance is not convincing, especially when compared to the IGMN, ISOINN, and LWPR$^+$.

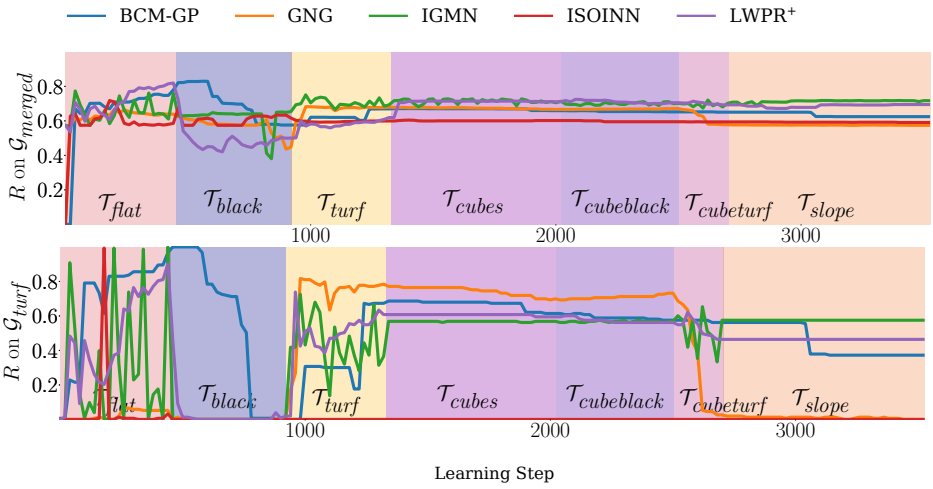


Fig. 3. The evolution of the correctness ratio R computed on $\mathcal{G}_{merged}$ and $\mathcal{G}_{turf}$ for the incrementally learned traversal cost models on $\mathcal{T}_2$.

The incremental nature of the learners can be observed in the evolution of the correctness ratio R presented in Figs. 2–5. A drop of the correctness ratio is expected behavior because of new information is incorporated into the models. It is especially prevalent for the performance evaluation on the individual terrain grids, where the GNG notably suffers by the dropout. Figs. 3 and 4 show a significant drop in the GNG performance on the $\mathcal{G}_{turf}$ grid after learning the model from $\mathcal{T}_{cubeturf}$. Arguably, the performance of some of the other learners on $\mathcal{G}_{turf}$ is also relatively poor. The showed evolution of R reinforces the claim that the IGMN is the most stable approach, as new data rarely cause a drop in the overall correctness, although there are some short term fluctuations.

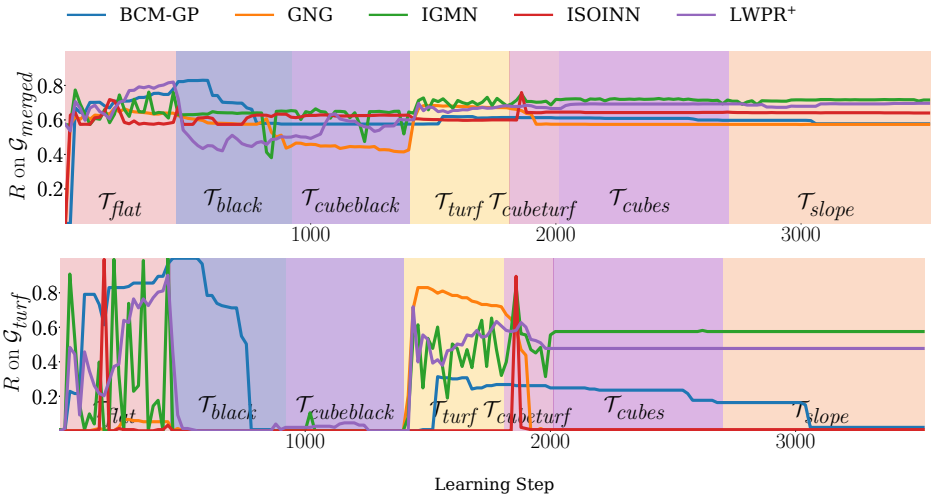


Fig. 4. The evolution of the correctness ratio R computed on $\mathcal{G}_{merged}$ and $\mathcal{G}_{turf}$ for the incrementally learned traversal cost models on $\mathcal{T}_3$.

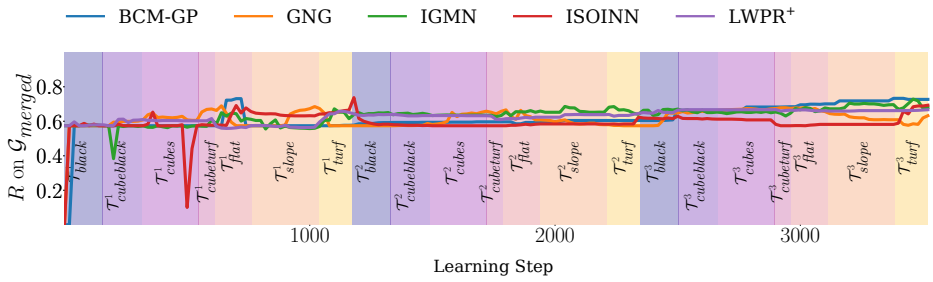


Fig. 5. The evolution of the correctness ratio R computed on $\mathcal{G}_{merged}$ for the incrementally learned traversal cost models on $\mathcal{T}_4$.

4.3 Qualitative Analysis

In our experience, the most significant ability of a traversal cost model is the inference of the traversal cost values that can be used to identify hard to traverse terrains. The adherence to the GP-based reference traversal cost model is not necessary, even though the reference model provides baseline predictions. Therefore, we investigate the predicted values over individual terrains.

The grid maps in Figs. 6 and 8 indicate that the reference, IGMN, ISOINN, and LWPR$^+$ learned on $\mathcal{T}_1$ predict a higher cost on $\mathcal{G}_{cubes}$ than on $\mathcal{G}_{flat}$, and thus they provide desired information for path planning to avoid difficult terrains.

The differences between predictions on $\mathcal{G}_{slope}$ showed in Fig. 7 are not surprising given that the predictive variance of the reference is higher than on other terrains. Notice, the ISOINN and LWPR$^+$ predictions are noisy over all the ter-

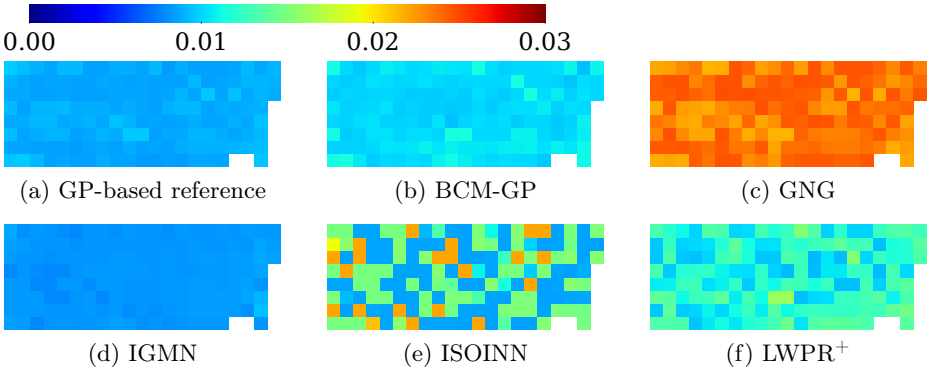


Fig. 6. The (a) mean of the GP reference model, and (b) BCM-GP, (c) GNG, (d) IGMN, (e) ISOINN, (f) LWPR$^+$ predictions over $\mathcal{G}_{flat}$ learned on $\mathcal{T}_1$. Only grid cells corresponding to the spatial allocation of the *flat* terrain are shown.

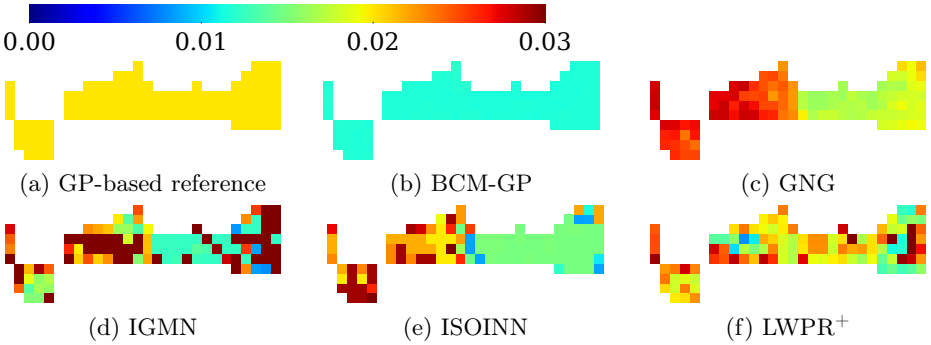
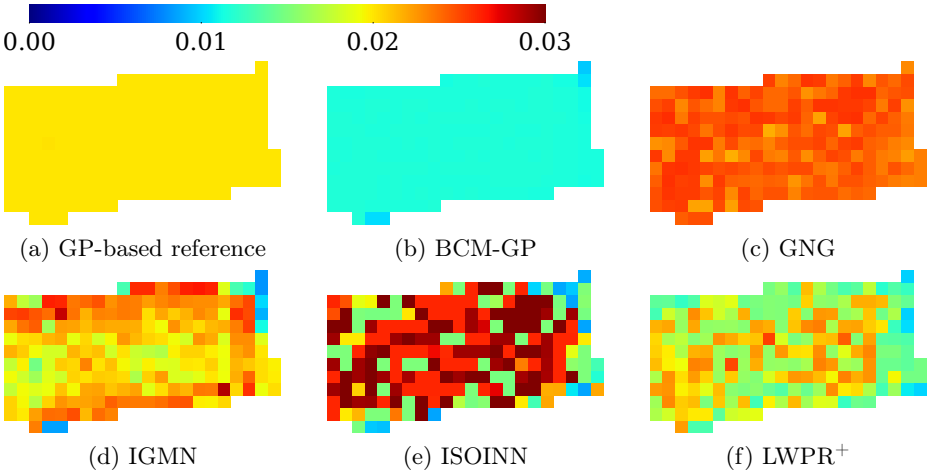


Fig. 7. The (a) mean of the GP reference model, and (b) BCM-GP, (c) GNG, (d) IGMN, (e) ISOINN, (f) LWPR$^+$ predictions over $\mathcal{G}_{slope}$ learned on $\mathcal{T}_1$. Only grid cells corresponding to the spatial allocation of the *slope* terrain are shown.

rains, but the median values are applicable to path planning. However, several failed predictions on $\mathcal{G}_{flat}$ can be seen in Fig. 9. These are especially prevalent when learning on $\mathcal{T}_2$ and $\mathcal{T}_3$, where $\mathcal{T}_{flat}$ is traversed first. The overall performance of the GNG is considered mediocre, and it fails to predict distinctively low values on $\mathcal{G}_{flat}$ even when learned on $\mathcal{T}_1$. Nevertheless, its predictions on $\mathcal{G}_{cubes}$ and $\mathcal{G}_{slope}$ are considered well-suited for path planning. Finally, the BCM-GP fails to provide distinctive costs for most terrains, and therefore, it cannot discriminate the hard to traverse areas.

4.4 Discussion

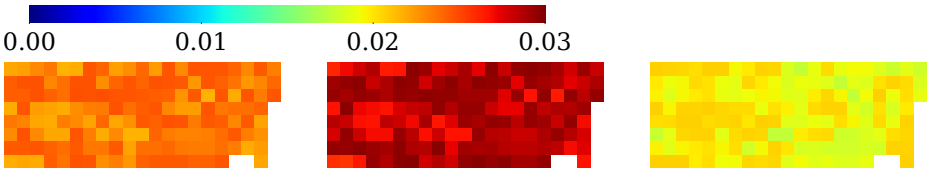
Based on the reported results, the IGMN, ISOINN, and LWPR$^+$ provide satisfiable performance in traversal cost inference for the considered hexapod walking robot, although ISOINN is more prone to forgetting or concept drift, see Fig. 9.



(a) GP-based reference (b) BCM-GP (c) GNG

(d) IGMN (e) ISOINN (f) LWPR$^+$

Fig. 8. The (a) mean of the GP reference model, and (b) BCM-GP, (c) GNG, (d) IGMN, (e) ISOINN, (f) LWPR$^+$ predictions over $\mathcal{G}_{cubes}$ learned on $\mathcal{T}_1$. Only grid cells corresponding to the spatial allocation of the *cubes* terrain are shown.



(a) GNG learned on $\mathcal{T}_1$ (b) GNG learned on $\mathcal{T}_2$ (c) ISOINN learned on $\mathcal{T}_2$

Fig. 9. Examples of high-cost predictions over $\mathcal{G}_{flat}$ which make it hard to discriminate the flat terrain from rough terrains.

The BCM-GP and GNG provide competitive results in terms of the correctness ratio and outperform the IGMN and LWPR$^+$ on some trails, but they do not provide predictions applicable to path planning. A close inspection indicates that the individual BCM-GP predictions are almost uniform and correspond to the most commonly observed traversal cost, i.e., the cost over the various easy to traverse terrains. Considering the previous deployment of the BCM-GP model in the exploration task in [12], these results are somewhat underwhelming. However, in the exploration task, the robot actively perceives the environment until all observed terrains are sufficiently known. In our case, the terrains on the test grids may remain unknown to the model. Nevertheless, the inability of the BCM-GP to extrapolate the predictions for such terrains is disappointing.

Table 2. $\mathcal{T}_1$ learning and $\mathcal{G}_{merged}$ prediction wall times T_{wall} [s]. (Intel i5-4460 CPU)

Method/Phase	BCM-GP	GNG	IGMN	ISOINN	LWPR$^+$	Compound GP
Learning	5.27	1.14	3.95	1.26	0.71	8.94
Prediction	22.44	0.62	4.54	0.60	0.59	8.64

The best performing regressors among the examined approaches are considered IGMN, ISOINN, and LWPR$^+$. The final IGMN and LWPR$^+$ predictions are the most stable, exhibiting only a very small difference when learning over permuted trails. The ISOINN outperforms the IGMN in few tested trails in terms of the correctness ratio. Moreover, according to the learning and inference times reported in Table 2, the ISOINN and LWPR$^+$ offer performance speedup over the IGMN in high dimensional learning, as they do not suffer from the quadratic time complexity with regards to the input dimension which hampers the IGMN.

5 Conclusion

In this paper, we present results on benchmarking five incremental learning approaches in traversal cost estimation of the hexapod walking robot. The main motivation of the presented work is to find alternatives to the IGMN algorithm we have used previously. The examined approaches include the Growing Neural Gas, Improved Self-Organizing Incremental Neural Network, Bayesian Committee Machine with Gaussian Process regressors, and Locally Weighted Projection Regression. Based on the presented results, the BCM-GP performs poorly, contradicting our initial intuition. The GNG performance is ambiguous, and it seems the used approach suffers from the concept drift for certain ordering of the learning sequences. The ISOINN and the hand-tuned LWPR$^+$ perform similarly to the IGMN in terms of the prediction correctness ratio, and all the three methods provide distinguishable cost prediction that is applicable in path planning. However, a possible drawback of the LWPR is the necessity of tuning the parameters and the grid search on $\mathcal{G}_{merged}$ provides significantly worse results in the case of $\mathcal{T}_2$ and $\mathcal{T}_3$ trails. Moreover, the ISOINN and LWPR offer better potential in high dimensional feature spaces, where the IGMN suffers from quadratic time complexity. Therefore, we consider the ISOINN and LWPR as the primary traversal cost learners for further deployments.

Acknowledgments

This work was supported by the Czech Science Foundation under research project No.18-18858S.

References

1. Bartoszyk, S., Kasprzak, P., Belter, D.: Terrain-aware motion planning for a walking robot. In: International Workshop on Robot Motion and Control (RoMoCo). pp. 29–34. IEEE (2017). doi: 10.1109/RoMoCo.2017.8003889
2. Brunner, M., Brüggemann, B., Schulz, D.: Rough Terrain Motion Planning for Actuated, Tracked Robots. In: Intl. Conf. Agents and Artificial Intelligence (ICAART). pp. 40–61 (2013). doi: 10.1007/978-3-662-44440-5_3
3. Deisenroth, M.P., Ng, J.W.: Distributed gaussian processes. In: Intl. Conf. International Conference on Machine Learning (ICML). pp. 1481–1490 (2015)

4. Faigl, J., Prágr, M.: Incremental traversability assessment learning using growing neural gas algorithm. In: Advances in Self-Organizing Maps, Learning Vector Quantization, Clustering and Data Visualization. pp. 166–176 (2020). doi: 10.1007/978-3-030-19642-4_17
5. Fritzke, B.: A growing neural gas network learns topologies. In: Neural Information Processing Systems (NIPS). pp. 625–632 (1994)
6. GPy: A Gaussian process framework in python. http://github.com/SheffieldML/GPy (since 2012), cited on 2019-03-28
7. Kragh, M., Jørgensen, R.N., Pedersen, H.: Object Detection and Terrain Classification in Agricultural Fields Using 3d Lidar Data. In: Int. Conf. Computer Vision Systems (ICVS). vol. 9163, pp. 188–197 (2015). doi: 10.1007/978-3-319-20904-3_18
8. LWPR library. https://github.com/jdlangs/lwpr (since 2007), cited on 2019-05-28
9. Nowicki, M.R., Belter, D., Kostusiak, A., Čížek, P., Faigl, J., Skrzypczynski, P.: An experimental study on feature-based SLAM for multi-legged robots with RGB-D sensors. Industrial Robot **44**(4), 428–441 (2017). doi: 10.1108/IR-11-2016-0340
10. O'Callaghan, S., Ramos, F.T., Durrant-Whyte, H.: Contextual occupancy maps using Gaussian processes. In: IEEE Int. Conf. Robotics and Automation (ICRA). pp. 1054–1060 (2009). doi: 10.1109/ROBOT.2009.5152754
11. Pinto, R., Engel, P., Alegre, P.: A Fast Incremental Gaussian Mixture Model. PLOS p. e0141942 (2015). doi: 10.1371/journal.pone.0139931
12. Prágr, M., Čížek, P., Bayer, J., Faigl, J.: Online Incremental Learning of the Terrain Traversal Cost in Autonomous Exploration. In: Robotics: Science and Systems (RSS) (2019). doi: 10.15607/RSS.2019.XV.040
13. Prágr, M., Čížek, P., Faigl, J.: Cost of Transport Estimation for Legged Robot Based on Terrain Features Inference from Aerial Scan. In: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS). pp. 1745–1750 (2018). doi: 10.1109/IROS.2018.8593374
14. Prágr, M., Čížek, P., Faigl, J.: Incremental Learning of Traversability Cost for Aerial Reconnaissance Support to Ground Units. In: Modelling and Simulation for Autonomous Systems (MESAS). pp. 412–421 (2019). doi: 10.1007/978-3-030-14984-0_30
15. Shen, F., Yu, H., Sakurai, K., Hasegawa, O.: An incremental online semi-supervised active learning algorithm based on self-organizing incremental neural network. Neural Computing and Applications **20**(7), 1061–1074 (2011). doi: 10.1007/s00521-010-0428-y
16. Tresp, V.: A Bayesian Committee Machine. Neural Computation **12**(11), 2719–2741 (Nov 2000). doi: 10.1162/089976600300014908
17. Tucker, V.A.: The Energetic Cost of Moving About: Walking and running are extremely inefficient forms of locomotion. Much greater efficiency is achieved by birds, fish—and bicyclists. American Scientist **63**(4), 413–419 (1975)
18. Vijayakumar, S., Schaal, S.: Locally Weighted Projection Regression : An O(n) Algorithm for Incremental Real Time Learning in High Dimensional Space. In: Intl. Conf. International Conference on Machine Learning (ICML). pp. 1079–1086 (2000)
19. Xiang, Z., Xiao, Z., Wang, D., Xiao, J.: Gaussian kernel smooth regression with topology learning neural networks and Python implementation. Neurocomputing **260**, 1–4 (2017). doi: 10.1016/j.neucom.2017.01.051
20. Čížek, P., Masri, D., Faigl, J.: Foothold Placement Planning with a Hexapod Crawling Robot. In: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS). pp. 4096–4101 (2017). doi: 10.1109/IROS.2017.8206267