Autonomous Gait Selection for Energy Efficient Walking

Sandeep Manjanna[1] and Gregory Dudek[1]

***Abstract*— In this paper, we investigate the question of how a legged robot can walk efficiently by taking advantage of its ability to alter its gait as a function of statistical (large-scale) terrain properties. One of the contributions of this paper is the algorithm to achieve real-time terrain identification and autonomous gait adaptation on a legged robot. We approach this problem by first classifying the terrains based on their proprioceptive responses and identifying the terrain in real-time. Then we choose an optimal gait to best suit the identified terrain type. We exploit our recent findings regarding gaits, estimated from terrain-contact signatures, in order to obtain an optimized mapping between terrain signatures and terrain-specific gaits. We evaluate our algorithm on synthetic data, and real robot data collected on different terrains and naturally occurring terrain transitions. Another key contribution of this work is the statistical verification that precise gait selection can lead to energy savings in practice in legged robots. This assessment of energy efficiency, achieved by gait adaptation, is among the firsts of its kind in gait adaptation literature. We also present an analysis of the effect of terrain transition frequency on our gait adaptation algorithm. Our results are supported by validation using both synthetic data and field testing.**

I. INTRODUCTION

In this paper we propose an algorithm that allows a walking robot to autonomously adapt its locomotion to its environment in real-time, and thus to be capable of navigating difficult terrains. The ability of legged robots to navigate through different kinds of rough terrains is one of their major strengths. Legged locomotion can provide robotic systems with the ability to traverse rough, rocky, sandy, steep, and other challenging terrains. Such robots can also use different kinds of leg movements to jump or step over an obstacle, climb stairs, and crawl on highly inclined surfaces. Energy efficiency while walking, the selection of an effective gait (a pattern of movement of limbs in animals, used for locomotion) to move efficiently on challenging terrains, and the maintenance of static and dynamic stability of the system while walking, are a few of the challenges faced by legged robots. We address the adaptability and walking efficiency problem through our real-time autonomous gait adaptation algorithm.

Gait selection and control is usually achieved in one of the three canonical ways: open-loop control, foot-placement modeling, and hybrid methods [1]. Clock-driven open-loop control depends on vehicle dynamics to compensate for terrain roughness [2]. Here, the system would ignore the response from the environment and continue an open-loop pattern of leg movements. In contrary, foot-placement modeling plans every foot placement carefully based on the terrain models or terrain feedback [3]. To operate with a stable gait, the system with foot-placement modeling requires a robust and detailed profiling of the ground surface. This increases the complexity and potentially makes the system fragile. We focus on the least often considered hybrid approach in which we try to merge the advantages of both the approaches. The novel contribution of this work is to modify an open-loop gait to adapt its behavior according to the terrain type.

Different terrains imply different optimal walking behaviors; a phenomenon well known to any person who has had to walk across an ice-covered sidewalk during a Canadian winter. The subsystems in an animal's brain adapt in order to achieve a stable gait on different terrains. Similarly, terrain-specific gait changes in legged robots are needed to optimize the performance. Legged platforms need to walk through tough terrains without getting unstable or painfully inefficient. A legged robot, walking with high leg rotation speed can get stuck in deep snow, but lower leg rotation speeds allow the robot to walk without digging into the snow. Contrary to that, the robot walking on hard concrete surfaces can be efficient and achieve better ground speeds at higher leg rotation speeds. These observations motivated us to study inter-dependency between gait and terrain properties [4]; and make use of these dependencies to enhance the efficiency of robot's navigation by adapting the gait parameters to changing terrain properties. We present statistical results that are consistent with our claim that gait selection based on terrain responses will increase the power efficiency in legged robots. Fig. 1 demonstrates the field trials conducted on the natural terrain transition between sand and grass to strengthen our results.



Fig. 1: Field trials conducted on continuous sand-grass terrain transition.

II. BACKGROUND

There is a body of prior work on analyzing the gaits in legged vehicles and sensing the terrain properties. The novel contribution of this paper is the use of gait and terrain

* We would like to acknowledge the NSERC Canadian Field Robotics Network (NCFRN) for its funding support.

[1]S. Manjanna and G. Dudek are with Center for Intelligent Machines (CIM), School of Computer Science, McGill University, Montreal, QC, Canada {msandeep, dudek}@cim.mcgill.ca

properties to adapt the robot's behavior in accordance with its environment in real-time. We present a gait adaptation algorithm that can be implemented on a RHex-like legged platform [2] to adapt its behavior in real-time.

Tactile sensing is an economical and robust solution for surveying the surface properties. In this work, we sense and identify the terrain through tactile feedback. One of the initial studies using whiskers to identify surface textures was conducted by Schultz et al. [5]. They were able to differentiate between smooth and rough surfaces using power spectral analysis. More recently, Giguere et al. [6], [7] used a tactile probe to analyze the acceleration patterns induced at the tip of the probe while it is dragged over different surfaces as a cue to terrain identity. Shill et al. [8] present terrain identification based on pressure images generated through direct surface contact using a robot skin constructed around a high-resolution pressure sensing array. In the aforementioned techniques, an external probe or a whisker or pressure sensing array is used to feel the surface. A more efficient way to sense a terrain is by considering the dynamics of the overall system when the robot is traversing different terrains. This can be achieved by measuring inertial estimations captured by inertial sensors mounted on the robot's body. We use this approach to record proprioceptive responses from the terrains.

The problem of terrain identification by legged vehicles was studied initially in the context of research on enhancing the mobility of planetary rovers. Krotkov et al. [9] proposed the modeling of terrain to plan the leg placements of the Ambler rover. In [10], Sadhukhan proposed a terrain classification technique using the internal sensors on a wheeled vehicle. This technique relied on vertical acceleration of the chassis of the vehicle to perform terrain identification. Giguere et al. [11] proposed an unsupervised machine learning algorithm to classify the proprioceptive data from different terrains. An advantage of this technique is that the method works by finding a set of parameter values for any user-specified classifier that minimizes a cost function. We use this methodology [11] in our real-time terrain identification for classifying the terrain response data into terrain classes.

Stable locomotion with legged vehicles on rough terrains has been the subject of study in a body of prior work. One of the approaches is to deliberately and carefully plan every footfall of the robot [3]. This approach needs a careful profiling of the terrain and involves several failure points with respect to terrain mapping, leg trajectory planning, and foot placement strategy. Weingarten et al., in their study to achieve automated gait adaptation [12], use a modified version of Nelder-Mead descent to adapt the gait parameters of the robot. This study concentrates on tuning the gait parameters for efficient locomotion of the RHex robot, but there is not much emphasis on the type of the terrain. In this work, we try to bridge the gap between terrain sensing and gait adaptation. An alternate approach for gait adaptation is to have explicitly designed gaits for specific purposes. These individual gaits are stored and activated based on the current purpose. We use a similar approach to map the gait parameters against the terrain type. Instead of purpose, we have the terrain type based on which a suitable gait-parameter value is picked. With a large set of possible gaits, the task to transition between the gaits becomes challenging. As we have a limited set of gaits, we have not yet considered the complex gait transition challenges in this work.

III. GAIT PARAMETERS AND TERRAIN PROPERTIES

In this work, we alter the gait of the robot by controlling one of the gait parameters, i.e. leg cycle-frequency (f_c) or the inverse of total stride period. For simplicity and clarity, we are considering only one parameter (f_c) in our current study. Our approach can be generalized for other gait parameters such as, leg offset, stance period, phase offset, and ground contact ratio and we plan to do it in our future work. We exploit our recent findings [4] regarding gaits estimated from terrain-contact signatures. Here, we consider optimal performance as a trade-off between the ground speed and the power consumption, high ground speeds achieved at not very expensive power consumptions. Table I presents a mapping of the terrain and the cycle-frequencies (f_c) at which the highest ground speed is attained and f_c values at which an acceptable power consumption is achieved. The fourth column presents the f_c values at which an optimal performance can be expected.

TABLE I: Inferences from the experimental observations of our previous study [4].

Terrain	f_c (max Ground Speed)	f_c (acceptable Power Consumption)	**Optimal Performance**
Concrete	1.703 Hz	1.332 Hz	$1.332 < f_c < 1.703$
Grass	1.703 Hz	1.703 Hz	$f_c = 1.703$
Dry Sand	2.05 Hz	1.703 Hz	$1.703 < f_c < 2.05$
Wet Sand	2.05 Hz	2.05 Hz	$f_c = 2.05$

We build a behavior repertoire based on these inferences such as to map terrain types with the optimal gaits (f_c). This repertoire will be used for gait look-up once the terrain type is identified. The optimal cycle frequency is computed by interpolating between the optimal f_c values in the Table I. Table II presents the behavior repertoire for the robot. The first column presents the terrain labels, their matching gait labels and gait f_c values are presented in the third and fourth columns of the table respectively.

TABLE II: Mapping of terrain types and optimal gaits

Terrain Label	**Terrain**	**Gait Label**	**Optimal Cycle-frequency (Hz)**
A	Concrete	Gait A	$f_c = 1.495$
B	Grass	Gait B	$f_c = 1.703$
C	Dry Sand	Gait C	$f_c = 1.922$
D	Wet Sand	Gait D	$f_c = 2.05$

IV. TERRAIN IDENTIFICATION AND GAIT ADAPTATION ALGORITHM

The block diagram in Fig. 2 shows the data flow and stages in our gait adaptation algorithm. We make an assumption that each terrain type is encountered for several consecutive foot steps, i.e. for long-term. The whole process is divided into two phases: a training phase and an execution phase. The training phase is indicated by the dotted rectangle in the Fig. 2. In the training phase, we collect proprioceptive data from different terrains and build classifiers using an unsupervised cost-based learning technique. At the end of the training phase, we have classifier parameters and terrain data with labels. In the execution phase, we classify the data coming from the robot's sensors in real-time and identify the long-term terrain class based on noisy short-term data from each footfall.

Once we know the long-term terrain, we make the decision on appropriate gait for that terrain and feed the gait back into the robot. Thus, the algorithm enhances the robot's abilities by making it capable of adapting its gait in real-time. The algorithm is semi-supervised because we use an unsupervised cost based classifier in training phase, but apply a supervised method in the execution phase with the labels from the training phase. Each stage in the diagram (Fig. 2) and the data flow between the stages is discussed further in this section. The pseudo code in Algorithm 1 presents step-wise details for the computational stages in the block diagram.

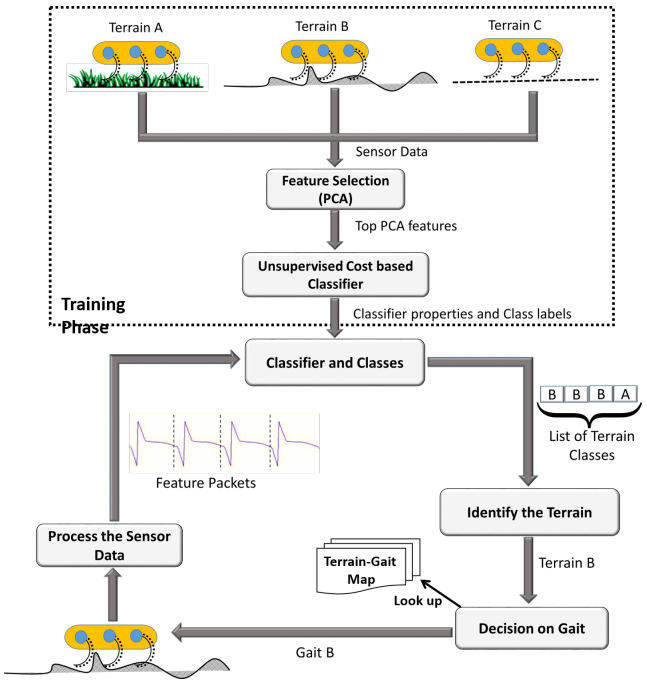


Fig. 2: Block diagram of Gait Adaptation Algorithm

A. Sensor Data

For this work, our experimental data is collected and tested using a hexapod robot from the Aqua family [13] which have a body plan based on the work of Buehler et al. [2]. The robot is made to walk on different terrains and multiple sensors on-board are used to collect proprioceptive data. We record the leg positions, electrical current through the robot's legs, linear accelerations, angular velocities, battery current and voltage drawn.

B. Feature Selection

The feature selection stage is responsible for generating a vector $\vec{x_t}$ as an input to the classifier. The data vector $\vec{x_t}$ represents the data from one complete leg cycle. The vector consists of data collected by sampling 13 sensors - comprising of 3 accelerometers, 3 rate gyroscopes, 1 leg angle encoder and 6 motor current estimators. The dimensionality of the dataset is reduced by applying Principal Component Analysis (PCA).

C. Classification Methodology

We use a methodology developed by Giguere et al. [11] for unsupervised clustering of proprioceptive data samples. The key to this approach, and its primary advantage, is that it takes advantage of the spatial (and hence temporal) continuity of the measurements from the robot as it traverses a terrain. Since the samples are generated through a physical system interacting with a continuous or piece-wise continuous terrain, piece-wise temporal coherence will be exhibited between consecutive samples. The clustering algorithm [11] explicitly exploits this time-dependency.

In our approach, we are altering the walking behavior of the robot which in turn affects the sensor-response from the terrain. The two plots in Fig. 3 illustrate the difference in terrain responses observed by the robot sensors when the robot is operated with different f_c value. This implies that a set of classifier parameters, estimated by a robot walking with ${f_c}^1$, cannot be used to classify the terrain-data from the robot walking with ${f_c}^2$. Hence, we use gait-specific classifier, i.e. if we have n gaits, we repeat the training phase of the algorithm n times with the data generated by each of these n gaits to estimate n classifiers.

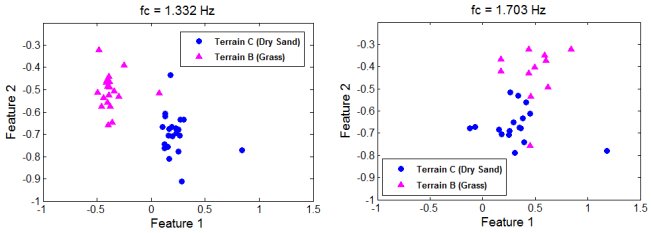


Fig. 3: Plots illustrating the effect of gait on data distribution.

D. On-line Terrain Identification

The objective of the training phase is to efficiently generate a classifier that covers the data using an associated small set of terrain class labels. In the execution phase, the aim is to identify the terrain using the real-time data from the robot when it is walking and feed back the robot with an

optimal gait. Initially, a random gait is selected for the robot to walk. The sensor data from the robot is split into feature packets, each packet comprising of data samples from one complete leg cycle. The feature packets are then fed into the classifier. The classifier classifies each feature packet into its terrain class and generates a list of class labels representing all the feature packets. Based on the current gait in use, the respective classifier is chosen to perform terrain identification.

This stage identifies the terrain by employing decision making strategies on the list of class labels. We assure the longtermness of the terrain class by considering a group of consecutive class labels for terrain identification instead of a single class label. Different decision strategies, such as maximum appearance strategy (a class label that appears most frequently in the list is decided to be the terrain class), certainty of the class strategy (a class with least uncertainty is chosen as the current terrain class), and consecutive similarity strategy (a class with maximum consecutive appearances will be chosen as the terrain class), could be employed in making the decision about the long-term terrain class. We use the maximum appearance strategy due to its simplicity and efficiency.

E. Gait Switch

This algorithm selects an optimal gait based on the identified terrain. Once we have identified the terrain, the task is to look up the behavior repertoire in Table II and pick the optimal cycle-frequency (f_c) that corresponds to the terrain type.

Different strategies can be considered to make the decision to transit between the gaits [14]. We considered the following three strategies: 1) Cost-based, 2) Utility-based, and 3) Greedy. In cost-based approach, we observed the instability during the gait switch by measuring the roll and pitch of the robot during the gait transition phase. In utility-based approach, we computed the value of gait transition in terms of an utility function, i.e. power efficiency in this case. The ratio of power efficiency in ideal case to that of the non-ideal case will represent the value of the gait-switch. In our case, as illustrated in the in Fig 4, this value gained by gait-switch between sand and grass will be $(\frac{55.83}{49.05}) = 1.14$. Both the aforementioned strategies are applicable when we have a priori knowledge, hence are not on-line. Greedy approach switches the gait as soon as a terrain change is sensed. In this paper, we are using a terrain-gait mapping (Table II) which is already optimized in terms of gait stability and power efficiency. Hence, we use the greedy approach in this work.

Algorithm 1 Semi-supervised Gait Adaptation Algorithm

Input: Set of n terrain classes, $T = \{t_1, t_2, ...t_n\}$.
Set of n gaits, $G = \{g_1, g_2, ...g_n\}$.
$\forall\ g_i$ in G, $\vec{X}_G = \{\vec{X}_{g_1}, \vec{X}_{g_2}, ...\vec{X}_{g_n}\}$.
Where, $\vec{X}_{g_i} = \{\vec{X}_{t_1}, \vec{X}_{t_2}, ..., \vec{X}_{t_n}\}$, $\forall\ t_i$ in T.
Map$\{T \leftarrow G\}$ ▷ A mapping of Gaits to Terrains
Output: t_{curr} : Current terrain identified.
g_{new} : Optimal gait chosen for terrain t_{curr}

Training Phase:

▷ Every gait is assigned its own classifier
for each g_i in G **do**
▷ We use a cost based unsupervised methodology [11] to obtain the classifier parameters.
classifier param: $\vec{\theta}_{g_i} \leftarrow$CostBasedClassifier($\vec{X}_{g_i}$)
$c_{g_i} \leftarrow$ Classifier ($\vec{\theta}_{g_i}$,T)
$g_i \leftarrow \{g_i, c_{g_i}\}$
end for

Execution Phase:

initialize current gait: $g_{curr} \leftarrow$ Random(G)
define: switchThreshold
while $(True)$ **do**
sensor data from terrain: $\vec{X}_{t_i} = \{\vec{x}_1, \vec{x}_2, .., \vec{x}_m\}$
▷ k samples per leg rotation.
feature packets:$\vec{F}$:$\{f_1, f_2, ..., f_{m/k}\} \leftarrow$ sample($\vec{X}_{t_i}$)
terrain labels: $\vec{t}$: $\{t_1, t_2, ..., t_{m/k}\} \leftarrow$ classify($\vec{F}$)
current terrain: $t_{curr} \leftarrow$ maxRepeat($\vec{t}$)
$g_{new} \leftarrow$ lookUp(Map$\{T \leftarrow G\}$,t_{curr})
if (switchVal($g_{new}, g_{curr}, t_{curr}$) $\geq$ switchThreshold) **then**
$g_{curr} \leftarrow g_{new}$
end if
end while

V. RESULTS AND OBSERVATIONS

We begin this section by presenting the statistically significant results indicating the effectiveness of our algorithm on energy efficiency of the legged platform. These results validate the motivation behind this work by strengthening our argument that choosing terrain-optimal gait leads to efficient walking behavior in the legged robots. Then we present results from the validation of our on-line gait adaptation algorithm.

We validate our gait adaptation algorithm with three types of datasets: synthetic datasets, real robot datasets from two terrains juxtaposed together to simulate terrain transition, and real robot datasets on naturally occurring continuous terrain transition (Fig. 1). The synthetic dataset includes data samples generated by two Gaussian distributions. We set the parameters of our synthetic distributions to be qualitatively similar to the kinds of terrain-depending signature changes we see in practice. The samples in these data are segmented and stitched to create scenarios of transitions between two terrains. We conducted multiple experiments with each dataset by changing the frequency of transitions between terrain classes.

The real robot dataset includes data samples collected by running the robot through different terrains. The data is collected from the robot at a rate of 50 Hz, i.e. 50 readings of sensor data are recorded per second, although the raw sensor measurements used are collected at a higher rate (e.g. 100 kHz for the IMU measurements). The data vector $\vec{x_t}$ consists of data collected by sampling 13 sensors at a rate

of 30 samples per one cycle of the leg. Thus, each feature vector $\vec{x_t}$ is of size 13x30.

In our algorithm, the terrain is not identified on every data packet, i.e. the terrain identity is not decided on every step of the robot. Instead, we consider a group of packets to decide on a particular terrain identity. Hence, we refer to it as a long-term terrain label. The number of data packets in each group is referred to as data packet length (DP_s). We use DP_s = 3 for the results presented here.

A. Energy conservation by Gait Adaptation

In this experiment, we made the robot walk five meters continuously between two terrains (grass and dry sand) with an equal distance of 2.5m on each terrain as illustrated in Fig. 1. We estimated the root mean square of power consumption per unit walk of the robot by collecting the battery voltage and battery current data. In the first two sets of experiments, we do not change the gait when transitioning into different terrain, i.e. the robot walks with optimal gait for grass (Gait B in Table II) over full path, and then the robot walks the full path with optimal gait for dry sand (Gait C in Table II). As a third set, we used gait adaptation when the terrain transition happens. Here, the first 2.5m of the path is traversed with the optimal gait for sand (Gait C), and the next 2.5m of the path is traversed with the optimal gait for grass (Gait B). The last set of experiments were performed with intentionally mismatched pairings of gait and terrain, i.e. Gait C on grass and Gait B on sand. The intent was to definitively verify the impact of selecting suboptimal gaits, and thus to determine the cost of any gait adaptation errors. Three trials of each experiment set were conducted and the plot in Fig. 4 displays standard error between the trials.

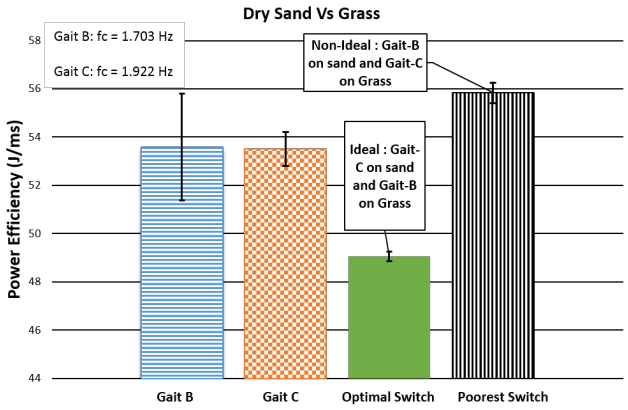


Fig. 4: Energy efficiency achieved by gait adaptation.

The results in Fig. 4 show that adapting the gait of the robot according to the terrain is 8.5% more power efficient than using a fixed gait on different terrains. The plot also compares the power consumption between the ideal gait adaptation and wrong gait adaptation. As seen from the results, we can reduce power consumption up to 12% by choosing right gait based on the terrain properties. These results are a major contribution of this work as they are among the firsts of their kind in the gait adaptation literature.

B. Experiments with Synthetic Dataset

In these experiments, the dataset is generated by two Gaussian distributions, each representing a terrain. Here, we are simulating the scenario of robot traversing between two terrains. The Gaussian distributions used to generate this dataset are derived from the real sensor data collected from the robot running on two different terrains - dry sand and grass. The two distributions used are shown in Fig. 5. We consider three different cases, in which the transition between the terrains is at different step lengths. For example, if the robot is walking on grass and enters a tilled area after 50 steps, then the transition step length has a value of 50. We use the notation T_s to represent the transition step length. Four cases we consider are with values T_s = {100, 3, 2}. T_s gives a notion of the frequency of terrain transitions, lower the value of T_s higher will be the terrain transition frequency.

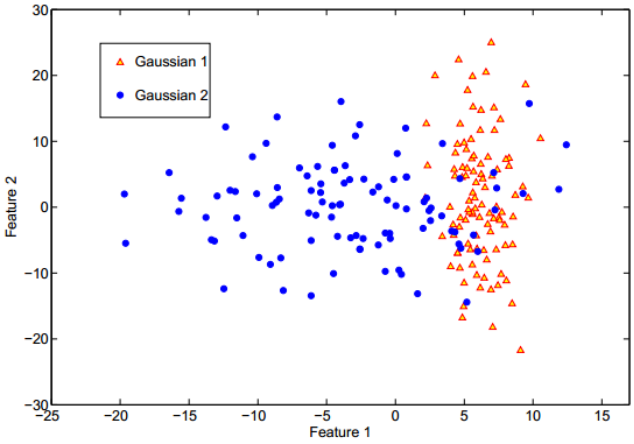


Fig. 5: Data simulating two terrains.

The results are illustrated by plotting the class labels : from ground truth, classifier output and gait adaptation algorithm output, against the time samples. Fig. 6 shows the data samples over 200 time samples. In this example, a transition between two terrains is shown by a switch after 100 time samples. An overlap between terrain classes results in classifier errors. The gait adaptation algorithm, however, performs efficiently as the gait decision is made by considering 3 consecutive data packets (DP_s) and not every data packet.

Data samples are altered by splitting and stitching to simulate a scenario of frequent terrain change. The results for experiments using dataset with a value of T_s = 3 is shown in Fig. 7. It is observed that the gait adaptation algorithm performs exceedingly well when the transition step is set equal to data packet length (T_s = DP_s). The gait adaptation algorithm performs poorly if the terrain transition occurs more frequently than the length of the data packet list considered. That is, if $T_s < DP_s$, the algorithm fails to choose the appropriate gait. This can be observed in Fig. 8.

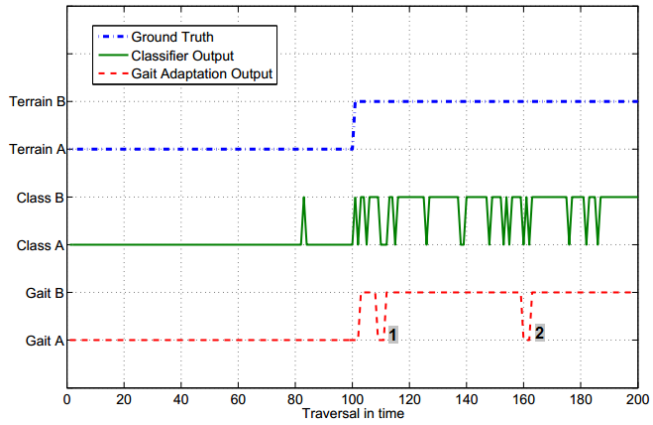


Fig. 6: Robot traversing from terrain A to terrain B only once. Gait selection is successful except at two time slots, 1 and 2.

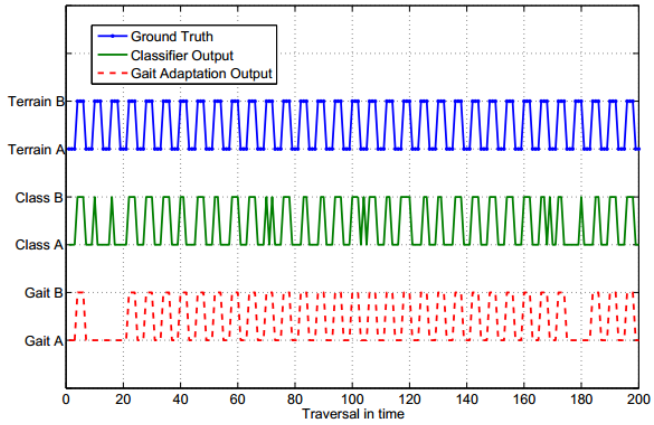


Fig. 7: Terrain transition for every 3 steps. In this test, transition step length is equal to the data packet length considered by gait adaptation algorithm (T_s = DP_s = 3). The success rate for gait selection is high.

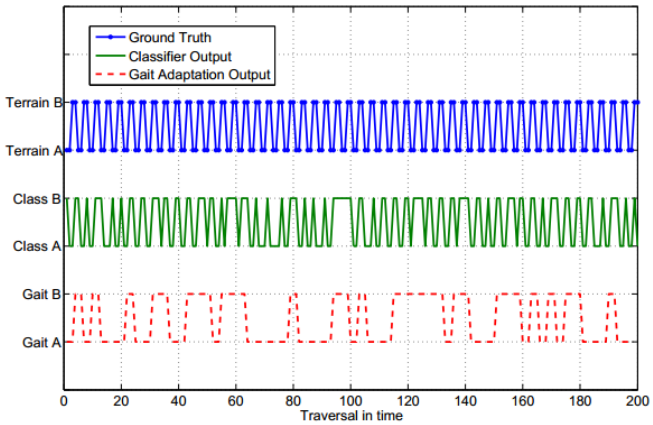


Fig. 8: Terrain transition for every 2 steps (T_s = 2). In this test, transition step length is smaller than the data packet length considered by gait adaptation algorithm. Gait selection fails.

We evaluated the performance of the gait adaptation algorithm and plotted the percentage error in the choice of gait against the terrain transition step lengths (T_s). Fig. 9 presents the variation of percentage error (rate of failure) in gait adaptation with terrain transition step length. It is observed that the error in the choice of gait increases with the reduction in terrain transition step length, that is, the performance of the algorithm is better if the frequency of terrain transition is low. The plot also illustrates a drop in the error when the transition step length is equal to data packet length (T_s = DP_s = 3).

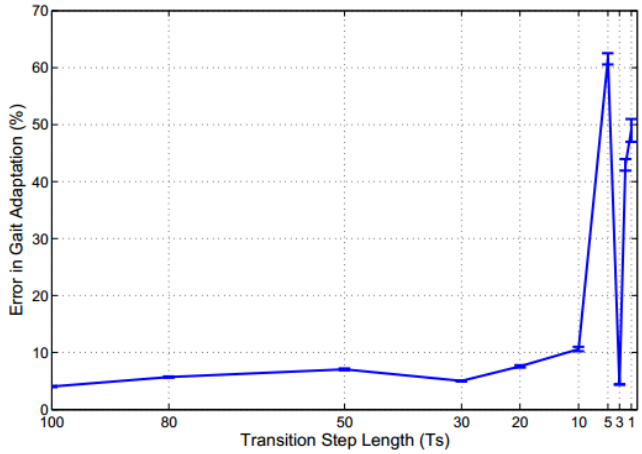


Fig. 9: Error in choosing right gait plotted against transition step length (T_s).

In Fig. 10, we illustrate the relationship between the terrain feature distribution (represented by the distance between the Gaussians) and the classification accuracy. Here, we observe that the classification error reaches minimum as the Gaussian distributions are moved apart. The results are in accordance with the expectation, farther the sample distributions, better is the performance of the linear separator.

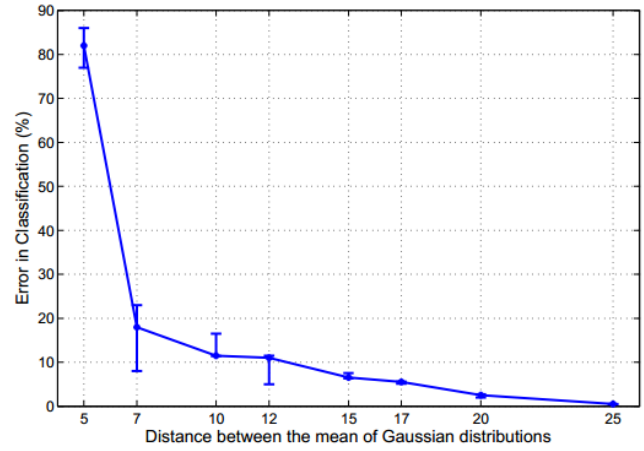


Fig. 10: Classification error plotted against distance between the means of Gaussian distributions.

C. Experiments with Real Robot Data

In this section we present the results produced with the real robot experiments conducted in field tests. We initiated walking sequences using a radio link to the robot, but in related work we have also examined gestural interfaces to in-field control of walking and swimming behaviors [15]. Our measurements of odometry and system dynamics were achieved using external observations of the robot as well as measurements from the on-board IMU although we plan to exploit in-board visual odometry in the future. We consider two different datasets generated by the real robot trials: 1) a dataset created by juxtaposing the data obtained by running the robot on two terrains (grass and dry sand) separately to simulate a terrain transition, 2) a dataset obtained by running the robot on a naturally occurring continuous path having a terrain transition from dry sand to grass as shown in Fig. 1.

1) Juxtaposed Robot Data: We collected the sensor samples as the robot walked over two different terrains - Dry Sand and Grass, and juxtapose these samples to form the first dataset. These sensor samples are split into data packets (each data packet containing the samples from one complete leg rotation), and then processed by PCA feature extractor to generate the dataset. Fig. 11 illustrates the distribution of data packets from two terrains, dry sand and grass.

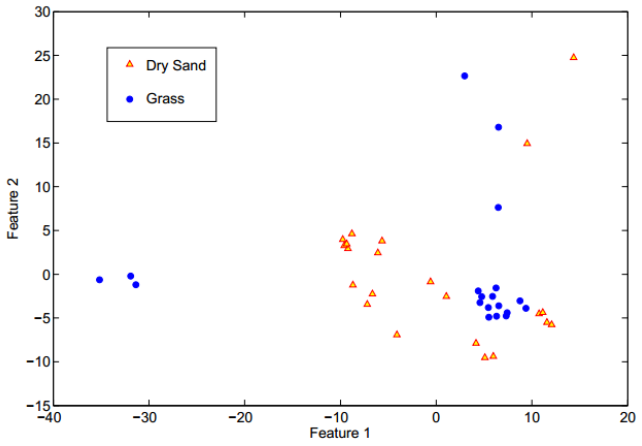


Fig. 11: Robot sensor dataset collected from two terrains : dry sand and grass.

Similar to previous experiments, we evaluate the algorithm with three test cases by varying the transition step length (T_s). In these experiments with robot sensor data, however, we use k-nearest neighbor classifier generated through cost minimization methodology. We use three neighbors ($k = 3$) for classification. Fig. 12, 13, and 14 present the classifier and gait adaptation algorithm results for a transition step length of 22, 3, and 2 respectively. As observed in previous experiments, results in Fig. 14 show that the performance of gait adaptation algorithm is poor when the terrain transition step length is lesser than the data packet length ($T_s < DP_s$).

2) Robot Data with Natural Terrain Transition: This dataset consists of robot sensor data collected over a walk

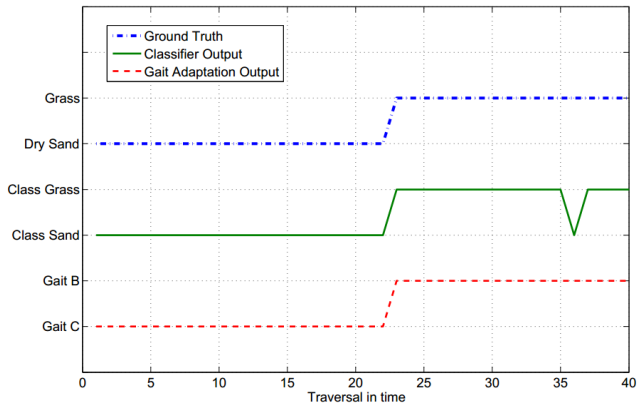


Fig. 12: Results of classification and gait adaptation plotted against time samples. Gait selection displays maximum success.

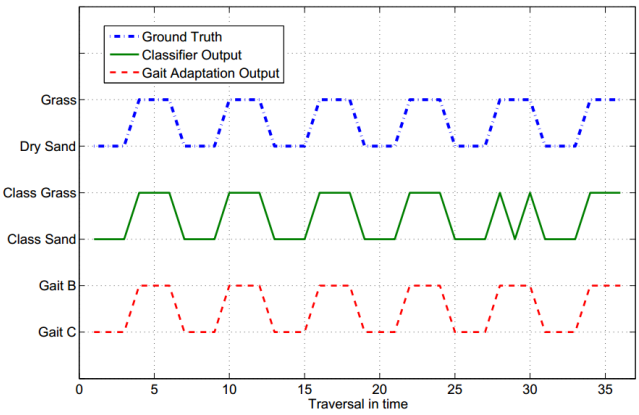


Fig. 13: The step length is changed to 3 (T_s=3) and this simulates a change in terrain for every 3 steps. Gait selection is successful.

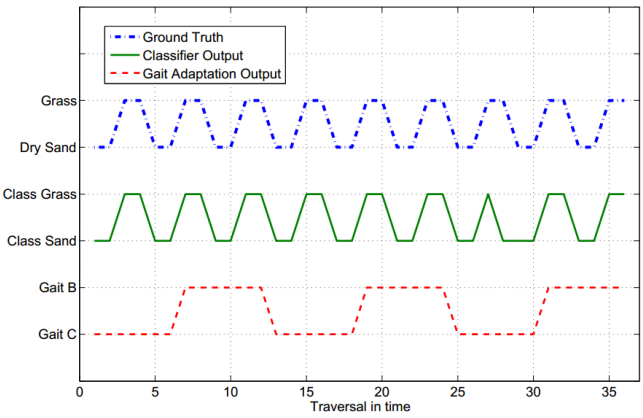


Fig. 14: The step length is changed to 2 (T_s=2) and this simulates a change in terrain for every 2 steps. Gait selection fails.

through the continuous path having a terrain transition between dry sand and grass. These experiments were conducted in a children's park and Fig. 1 demonstrates the experiment setup. As the length of walk considered for these experiments was limited to 5m, the dataset includes only 15 steps. The results in Fig. 15 illustrates a two-step delay in adapting to the right gait. This is because of the misclassification done by the classifier at the terrain-transition patch.

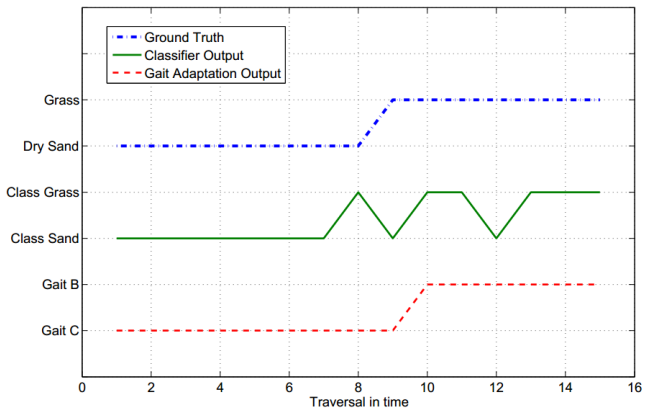


Fig. 15: Gait adaptation and classification results on the natural terrain with transition from dry sand to grass.

VI. CONCLUSIONS AND FUTURE WORK

In this work, we presented an approach to enhance the walking experience of an autonomous system by learning its environment through proprioceptive sensing and the moderation of its actions based on the environment. We experimentally observed a clear connection between the environment, the robot's performance, and the gait, and demonstrated that this connection can be exploited to enhance performance relative to prior approaches which did not adapt dynamically. We also provided statistically significant results to prove our claim that energy efficient walking can be achieved by adapting the gait of a legged robot to suit terrain properties. The results also validated the effectiveness of the algorithm proposed for autonomous gait adaptation based on the terrain feedback.

Our methodologies in this work have some limitations which we plan to address in near future. We limit our study of gait adaptation to only one of the gait parameters, which is cycle-frequency. Other gait parameters, the stance angle and the stance period, also have an impact on the performance of the robot and considering additional gait parameters and performance factors should lead to further performance enhancement, although the basic principle we have illustrated would still stand.

The algorithm we propose requires a predetermined set of terrain classes to successfully train the classifiers. There is a limitation with respect to the kinds of terrains that the algorithm is capable of adapting to. Also, our algorithm does not explore the entire space of gait parameter as we only use a set of fixed gaits. It would be more elegant to adapt to any new untrained terrain by optimizing over gait parameter space, however, the complexity in the gait space made us choose a set of predetermined gaits for our initial attempt with gait adaptation.

We plan to conduct further experiments with the robot running on a more diverse sampling of rough terrains and adapting its behavior accordingly. This would provide more insights on better strategies for terrain class decisions and value of gait switch. In future work, we also plan to enhance the gait adaptation algorithm to optimize the gait parameters of the robot, on the run.

REFERENCES

[1] J. K. Karpick, J. G. Cham, J. E. Clark, and M. R. Cutkosky, "Stride period adaptation for a biomimetic running hexapod," in *Robotics Research*. Springer, 2003, pp. 133–145.

[2] U. Saranli, M. Buehler, and D. E. Koditschek, "Rhex: A simple and highly mobile hexapod robot," *The International Journal of Robotics Research*, vol. 20, no. 7, pp. 616–631, 2001.

[3] D. Wettergreen, H. Thomas, and C. Thorpe, "Planning strategies for the ambler walking robot," in *IEEE International Conference on Systems Engineering, 1990.* IEEE, 1990, pp. 198–203.

[4] S. Manjanna, G. Dudek, and P. Giguere, "Using gait change for terrain sensing by robots," in *CRV '13: Proceedings of the 2013 International Conference on Computer and Robot Vision*. IEEE Computer Society, May 2013, pp. 16–22.

[5] A. E. Schultz, J. H. Solomon, M. A. Peshkin, and M. J. Hartmann, "Multifunctional whisker arrays for distance detection, terrain mapping, and object feature extraction," in *ICRA 2005. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005.* IEEE, 2005, pp. 2588–2593.

[6] P. Giguere and G. Dudek, "A simple tactile probe for surface identification by mobile robots," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 534–544, 2011.

[7] P. Giguere, "Unsupervised learning for mobile robot terrain classification," Ph.D. dissertation, McGill University, 2010.

[8] J. Shill, E. Collins, E. Coyle, and C. Jonathan, "Terrain classification on a one-legged hopping robot using high-resolution pressure images," in *Proceedings of 2014 IEEE International Conference on Robotics and Automation, 2014. ICRA 2014.* IEEE, 2014.

[9] E. Krotkov and R. Simmons, "Perception, planning, and control for autonomous walking with the ambler planetary rover," *The International journal of robotics research*, vol. 15, no. 2, pp. 155–180, 1996.

[10] D. Sadhukhan, "Autonomous ground vehicle terrain classification using internal sensors," 2004.

[11] P. Giguere and G. Dudek, "Clustering sensor data for terrain identification using a windowless algorithm," *Robotics: science and systems IV*, p. 25, 2009.

[12] J. D. Weingarten, G. A. Lopes, M. Buehler, R. E. Groff, and D. E. Koditschek, "Automated gait adaptation for legged robots," in *Proceedings of International Conference onRobotics and Automation, 2004, ICRA'04. 2004 IEEE*, vol. 3. IEEE, 2004, pp. 2153–2158.

[13] G. Dudek, P. Giguere, C. Prahacs, S. Saunderson, J. Sattar, L.-A. Torres-Mendez, M. Jenkin, A. German, A. Hogue, A. Ripsman, *et al.*, "Aqua: An amphibious autonomous robot," *Computer*, vol. 40, no. 1, pp. 46–53, 2007.

[14] G. C. Haynes and A. A. Rizzi, "Gaits and gait transitions for legged robots," in *Proceedings of 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 1117–1122.

[15] G. Dudek, J. Sattar, and A. Xu, "A visual language for robot control and programming: A human-interface study," in *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, 2007, pp. 2507–2513.