Active Perception for Modelling Energy Consumption in Off-Road Navigation

Rafael Oliveira, Lionel Ott, Fabio Ramos

School of Information Technologies, The University of Sydney, Australia.

rdos6788@uni.sydney.edu.au, lionel.ott@sydney.edu.au, fabio.ramos@sydney.edu.au

Abstract

In this paper, we propose a Bayesian optimisation (BO) method to actively learn a model of a robot's power consumption and use it to find energy-efficient paths between two fixed locations over an uneven terrain. Most of the prior work in this area relies on models of the vehicle's dynamics or on accurate information about the terrain and its physical properties. In contrast, our method uses Gaussian process (GP) regression to predict the power consumption of the robot in configurations along candidate paths. We combine this model with a BO algorithm using an energy-based exploration-exploitation criterion to select paths between the given start and goal locations and to find energy-efficient routes between them. Experiments in simulation with synthetic data evaluate the convergence of the model and the performance of the algorithm in building it and finding energy-efficient paths. We also demonstrate the efficiency of our method against other non-linear optimisation algorithms in the simulation scenario. Finally, we demonstrate the method working on a physical robot in a real-world environment.

1 Introduction

Mobile robots can be applied to a variety of tasks requiring them to traverse rough, uneven terrains multiple times, such as cargo transportation in mining operations [Brown, 2012], planetary exploration [Carsten *et al.*, 2007], etc. In such tasks, robots are often required to operate autonomously under limited resources.

Especially in the case of off-road navigation, an important constraint that limits the capability of a robot to operate autonomously is energy consumption. In this case, most of the limited on-board power supply of the robot is spent by the traction system. Besides internal factors such as vehicle dynamics, terrain properties can directly influence power consumption. Most of these factors are generally hard to model: for example, variations in roughness [Molino *et al.*, 2007], terrain slope [Sun and Reif, 2005], and interactions between the wheels and the soil [Amar and Bidaud, 1995]. Altogether, these terrain-related complexities introduce noise and uncertainty in power-consumption models. A large part of the research into energy-efficient path planning, however, assumes that the model of the terrain and its associated energy costs are known.

In this paper, we present a method that learns a model to predict the energy cost of paths connecting two locations over a possibly-uneven terrain and uses this model to find energy-efficient paths between those locations. The only information required by our method is the robot's localisation and the power-consumption measurements collected by the robot as it moves. The model is built using an active perception approach in which the robot executes paths between the given locations that are selected according to their expected energy consumption and the uncertainty in this estimate. With such a model, energy-efficient paths can be obtained for use in future tasks. In this work, we make the following contributions:

1. a model to estimate the power consumption of a robot moving at constant linear speed given its location and heading direction based on *Gaussian process* [Rasmussen and Williams, 2006] regression;
2. a path-planning method that explores paths between two fixed locations building a model of the robot's power consumption over the terrain using *Bayesian optimisation* [Brochu *et al.*, 2010].

The remainder of this paper is organised as follows. In the next section, we review relevant prior work in this area. In Section 3, we specify and formulate the problem of modelling energy consumption, which is followed by a description of the proposed path-planning algorithm in Section 4. Then, in Section 5, we present experimental results in simulation and on a physical robot to evaluate our approach. Finally, in Section 6, we conclude and propose some directions for future work.

2 Related Work

In this section, we review some relevant work done in the area of energy-aware motion planning and Bayesian approaches applied to planning under uncertainty.

2.1 Vehicle-Dynamics Control

Heavy robots usually spend most of their energy supply on mobility. Therefore, some research effort has been put into modelling the power demands of traction systems to provide cost models for energy-aware path planners. In this sense, energy efficiency can be achieved by the vehicle's motion control strategies, such as avoiding high-cost movements like braking and accelerating [Mei *et al.*, 2006], turning [Morales *et al.*, 2006], or by optimising velocity profiles ([Kim and Kim, 2007] and [Tokekar *et al.*, 2011]). In general, these methods are dependent on parameters that require extensive evaluation and analysis of the robot in experiments and are highly vehicle dependent.

2.2 Elevation-based Strategies

Another strategy is to model power usage as a function of terrain properties, approaching it from a mechanical energy point of view. If the geometry of the terrain is known a priori, different techniques using grid-based elevation maps can be used to estimate energy consumption along a path. This approach in general infers the power costs associated with individual actions over the terrain as a function of local slope angles and friction coefficients ([Sun and Reif, 2005], [Liu *et al.*, 2008], [Anuntachai *et al.*, 2014], and [Ganganath *et al.*, 2014]). The energy expenditure of a path is then computed as the sum of the individual costs associated with each state transition on the path. However, these strategies require accurate information about the terrain type and its characteristics, besides the vast amounts of sensor data necessary to build the elevation models, which has its own challenges [Plagemann *et al.*, 2008].

2.3 Bayesian Approaches

Most of the approaches discussed so far depend on deterministic cost models that inform path planners when computing the costs of a path. Alternatively, Bayesian approaches can be effectively used to learn power-consumption models for energy-efficient path planning under uncertainty.

Plonski et al. [Plonski *et al.*, 2013] present a Bayesian approach to model energy. They use Gaussian process (GP) regression to build a solar-power map over an area, and then use this map to plan energy-optimal paths using empirical cost models to predict the energy spent on individual actions. Martin and Corke in [Martin and Corke, 2014] present an approach to map the robot's power usage over the terrain via GPs. At each iteration, their map is used to plan minimum-cost paths using D* and then updated with measurements collected along the way. A limitation of these models is that they do not consider uneven terrains, where the robot's heading affects the power consumption. Additionally, the uncertainty of the model's predictions is not taken into account during the planning phase.

Marchant and Ramos [Marchant and Ramos, 2014] present a Bayesian optimisation (BO) approach to plan continuous paths for robots to monitor spatial-temporal processes while exploring the environment. They use a GP to learn a model of the physical phenomena that the robot is monitoring, while BO is employed to select the sample-collecting path that yields the highest reward in each iteration. In this problem, the main goal is to build a model of the process being modelled throughout the whole area of the experiment, with less uncertainty in areas close to peaks of the phenomena. There is no goal location for the robot to reach. And the robot's heading does not affect the readings. However, a similar framework can be applied to the problem in this paper.

3 Problem Formulation

Our goal is to enable a robot to learn a model that can be used to find energy-efficient paths in an obstacle-free area over a terrain with smooth elevation changes. The paths are taken from a family of parametric curves $\mathcal{C}$ and constrained by a given fixed starting location $A = (x_0, y_0)$, with initial heading angle θ_0, and an also fixed goal location $B = (x_f, y_f)$, with free final orientation. Since we are searching for energy-efficient paths, the model should provide less uncertainty in the predictions for paths that will yield better energy savings. In addition, we assume that the only information we have access to are noisy measurements of the instantaneous power consumption w of the robot and its pose, which contains position and orientation. Therefore, we have no access to information about the terrain itself, such as slope, roughness, elevation maps, etc.

The total amount of energy E needed to execute a path $\mathcal{T}$ can be obtained by integrating the instantaneous power consumption w along the path over time, i.e.:

$$E[\mathcal{T}] = \int_{t_0}^{t_f} w(\mathbf{x}(t))dt. \tag{1}$$

where t_0 corresponds to the starting time, when the robot leaves A, t_f corresponds to the final time, when the robot reaches B, and $\mathbf{x} \in \mathcal{T}$ is a state vector comprising variables that affect the power demands of the traction system along the path. For our case, this state vector will simply be the robot's pose, $\mathbf{x} = (x, y, \theta)$, which is valid under a set of plausible assumptions to be stated later in the text.

In this setup, predicting the amount of energy required to execute a particular $\mathcal{T}$ is equivalent to providing a model to predict the expected instantaneous power consumption $w(\mathbf{x})$ at each state $\mathbf{x} \in \mathbb{R}^D$ that the robot will be along the path and then integrating over these estimates. Consequently, our task is to learn a function $f : \mathbb{R}^D \to \mathbb{R}$ mapping states to power consumption values. We assume this function is a sample from a Gaussian process and provide a Bayesian optimisation method to best select paths $\mathcal{T}$ from $\mathcal{C}$ so that our model is efficiently improved after executing each path from A to B.

3.1 Modelling Assumptions

In our model, we assume power consumption to be a function only of the robot's pose, which is provided by the localisation system. The pose alone does not provide information about other variables that can affect the power consumption of the robot. However, under a set of plausible assumptions, it can be sufficient for the present modelling task. Firstly, the variations in the power drawn by the traction system when the robot is in motion are usually much higher than the ones presented by the other on-board systems in any reasonably-sized robot. So we can assume that the power demanded by these other systems is constant. Secondly, we assume a static environment so that the terrain's characteristics do not change over time. Lastly, in our experiments the robot will travel at a constant linear speed $v > 0$ with an angular speed ω small enough to not significantly affect the power-consumption values. In addition, the GP inherently considers the noise of the underlying process, which allows it to adapt to the noise introduced by these limitations.

4 Methodology

In this section, we present our method to learn a model to predict energy consumption and find energy-efficient paths, which is comprised of two parts: a Gaussian process (GP) model and a Bayesian optimisation (BO) algorithm. After a quick overview of the proposed system, we provide a basic review on GPs and present how we apply them to estimate energy consumption over paths. Next, we introduce Bayesian optimisation and present our path-planning method, which selects paths to execute and updates the GP model.

4.1 System Overview

Figure 1 shows a simplified diagram of our system. The GP model is designed to learn the distribution of the expected power usage of the robot as it traverses the terrain. The model uses power-consumption measurements from traversed locations to infer the power necessary to traverse other locations, taking into account the heading direction as well. The BO algorithm queries the GP for individual estimates of power usage along a candidate path. By maximising a utility function, the algorithm selects the next path to follow. This path is given to the robot, which executes it, collecting power-consumption samples along the way. Finally, these samples are fed back as observations to the GP, updating the model. This cycle repeats in each iteration of the BO algorithm. As iterations are performed, the expected amount of energy required to execute a path, as predicted by the GP, converges to the measured amount of energy consumption.

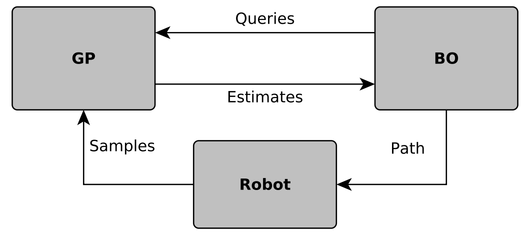


Figure 1: System diagram. The GP learns a model of the power consumption of the robot from the samples collected as it moves. The BO algorithm maximises a reward function over the GP estimates and selects the best candidate path. The robot executes the path, and the GP model is updated.

4.2 GP Model for Power Consumption

As a Bayesian non-parametric method, Gaussian process (GP) regression [Rasmussen and Williams, 2006] offers a powerful tool to model non-linear spatial distributions. For this reason, we utilise this method to learn a model that estimates the robot's instantaneous power-consumption $w(\mathbf{x})$ over the terrain.

Review on GP Regression

Unlike regular non-Bayesian regression methods, a GP places a joint Gaussian distribution as a prior over the space of functions that map inputs $\mathbf{x}$ to outputs z, where $z = f(\mathbf{x}) + \epsilon$ is a noisy observation of the true underlying function value $f(\mathbf{x})$, and $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$ is Gaussian-distributed noise with zero mean and standard deviation σ_n. A GP model can be completely specified by a mean function $m(\mathbf{x})$ and a positive semi-definite covariance function $k(\mathbf{x}, \mathbf{x}')$, such as the squared exponential

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')\mathbf{L}(\mathbf{x} - \mathbf{x}')^{\mathrm{T}}\right), \quad (2)$$

where σ_f^2 is called the signal variance, which measures how much the function is expected to deviate from the mean, and $\mathbf{L}$ is a D-by-D diagonal matrix with elements $\mathbf{L}_{ii} = 1/l_i^2$, which are length-scale coefficients that regulate the rate of change in covariance along each dimension of $\mathbf{x}$.

Given a set of N observations $\mathcal{O} = \{\mathbf{x}_i, z_i\}_{i=1}^N$, the GP predicts function values at a given query point $\mathbf{x}^*$ as $f(\mathbf{x}^*) \sim \mathcal{N}(\mu(\mathbf{x}^*), \sigma^2(\mathbf{x}^*))$, where:

$$\mu(\mathbf{x}^*) = m(\mathbf{x}^*) + \mathbf{k}(\mathbf{x}^*, \mathbf{X})\mathbf{K_X}^{-1}(\mathbf{z} - \mathbf{m}(\mathbf{X})) \quad (3)$$

with $\mathbf{X}$ as a N-by-D matrix formed by vertically stacking each $\mathbf{x}_i \in \mathcal{O}$, $\mathbf{K_X} = \mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I$, where each $\mathbf{K}(\mathbf{X}, \mathbf{X})_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, and $\mathbf{k}(\mathbf{x}^*, \mathbf{X})$ denotes a 1-by-N vector defined by $\mathbf{k}(\mathbf{x}^*, \mathbf{X}) = (k(\mathbf{x}^*, \mathbf{x}_1), \ldots, (k(\mathbf{x}^*, \mathbf{x}_N))$. The GP also determines the variance of its estimation as

$$\sigma^2(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}(\mathbf{x}^*, \mathbf{X})\mathbf{K_X}^{-1}\mathbf{k}(\mathbf{X}, \mathbf{x}^*) \quad (4)$$

which measures the uncertainty in the estimation.

Power-Consumption Modelling

To construct our GP model, we utilise a stationary mean function $m(\mathbf{x}) = m_s \; \forall \; \mathbf{x}$, where m_s is a constant hyper-parameter optimised during an initial training phase. We model the covariance between two input poses $\mathbf{x}$ and $\mathbf{x}'$ using a modified version of the squared exponential kernel in Equation 2:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}d^2(\mathbf{x}, \mathbf{x}')\right), \quad (5)$$

where $d(\mathbf{x}, \mathbf{x}')$ is a custom distance metric.

As our inputs contain angles, we can not directly apply Euclidean distances to compute the distance between two different poses. To address this, we compute differences in the angular component as follows:

$$|\theta - \theta'| = \cos^{-1}(\cos(\theta - \theta')) \in [0, \pi], \quad (6)$$

which corresponds to the shortest arc between the two orientation angles. With this, the distance between two points becomes:

$$d(\mathbf{x}, \mathbf{x}') = \sqrt{\frac{(x - x')^2}{l_x^2} + \frac{(y - y')^2}{l_y^2} + \frac{|\theta - \theta'|^2}{l_\theta^2}}, \quad (7)$$

where l_x, l_y, l_θ are positive length-scale hyper-parameters also learnt from a training set.

With this kernel, the covariance between function values at two different points $\mathbf{x}$ and $\mathbf{x}'$ will smoothly decay according to their distance. Consequently, this covariance function is suitable for modelling power-consumption as a Gaussian process on smooth natural terrains, which usually do not present abrupt transitions of surface characteristics.

All in all, our GP model is fully determined by the hyper-parameters $(m_s, \sigma_f, l_x, l_y, l_\theta, \sigma_n)$. In our implementation, these values are determined by maximising the log-marginal likelihood in a training phase on an initial set of observations. See Chapter 5 in [Rasmussen and Williams, 2006] for more details on hyper-parameters learning.

4.3 Energy Integration

With power-consumption estimates from the GP model, $w(\mathbf{x}) \sim \mathcal{N}(\mu_w(\mathbf{x}), \sigma_w^2(\mathbf{x}))$, for a finite set of poses $\{\mathbf{x}_i\}_{i=0}^n$ sampled along a candidate path $\mathcal{T}$, we can predict the total energy expenditure for that path, which we model as $E \sim \mathcal{N}(\mu_E, \sigma_E^2)$, as:

$$\mu_E[\mathcal{T}] = \sum_{i=0}^{n-1} \mu_w(\mathbf{x}_i)\Delta t_i, \quad (8)$$

$$\sigma_E^2[\mathcal{T}] = \sum_{i=0}^{n-1} \sigma_w^2(\mathbf{x}_i)\Delta t_i^2. \quad (9)$$

Since we have no access to information about the terrain elevation, we estimate the time interval between two consecutive poses on a path using a linear 2D motion model with constant speed v, as:

$$\Delta t_i \approx \frac{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}}{v}. \quad (10)$$

This approximation does not consider height differences, which could lead to errors in areas of high slope. However, for typical smooth terrains, this should not lead to significant errors in the estimation of E.

4.4 Bayesian Optimisation for Path Selection

We propose a Bayesian optimisation (BO) method to enable the robot to select paths trading off going from A to B as efficiently as possible with the largest amount of model improvement.

The BO Strategy

BO [Brochu *et al.*, 2010] allows computing the global optimum of unknown functions that are expensive to evaluate. The BO strategy applies a Bayesian model, which is typically a GP, as a prior to internally approximate the objective function. At each iteration, BO selects the point to perform the next evaluation of the objective based on maximising an acquisition function over the prior. This acquisition function is usually much simpler to evaluate than the objective and provides a utility value that enables the algorithm to perform a guided search for the global optimum. After evaluating the objective, the prior is updated with the new observation, and the algorithm proceeds to the next iteration, keeping track of the current optimum. As iterations proceed, the prior approximation converges to the objective function around the observed regions. Therefore, BO can be used not only to optimise, but also to build a model of the objective function.

Acquisition Function

In this paper, we use BO to optimise over a model mapping paths to energy-consumption estimates. A function $R[\mathcal{T}]$ works as an acquisition function for BO and computes the reward for executing a candidate path $\mathcal{T}$ based on its expected energy consumption, modelled as $E \sim \mathcal{N}(\mu_E, \sigma_E^2)$, and the uncertainty about it. In our case, we use an upper-confidence bound (UCB) [Brochu *et al.*, 2010] over the negative of the energy estimate, i.e.:

$$\begin{aligned} R[\mathcal{T}] &= -\mu_E[\mathcal{T}] + \lambda\sigma_E[\mathcal{T}] \\ &= -\sum_{i=0}^{n-1} \mu_w(\mathbf{x}_i)\Delta t_i + \lambda \left(\sum_{i=0}^{n-1} \sigma_w^2(\mathbf{x}_i)\Delta t_i^2 \right)^{1/2}, \end{aligned} \tag{11}$$

where λ is a positive scalar which controls the exploration-exploitation trade-off. Higher values of λ increase exploration giving high reward for paths whose energy estimate has a higher uncertainty, since observations along them might provide useful information to improve the accuracy of the GP estimates. On the other hand, lower values of λ favour exploitation, with the function preferring paths which the model is certain that will have low energy cost.

Path Parameterisation

We model paths using parametric curves $\mathcal{C}(u, \boldsymbol{\beta})$, which map $\mathbb{R}$ to $\mathbb{R}^D$ with $u \in [0, 1]$ and coefficients $\boldsymbol{\beta}$. In our case, we utilise 2D cubic splines, which are simple to work with and flexible enough to demonstrate the capabilities of our algorithm.

Poses along a path are computed using the following equations:

$$x = \mathcal{C}_x(u, \boldsymbol{\beta}) = a_x u^3 + b_x u^2 + c_x u + d_x \tag{12}$$

$$y = \mathcal{C}_y(u, \boldsymbol{\beta}) = a_y u^3 + b_y u^2 + c_y u + d_y \tag{13}$$

$$\tan\theta = \frac{\partial y}{\partial x} = \frac{\partial y / \partial u}{\partial x / \partial u} = \frac{3a_x u^2 + 2b_x u + c_x}{3a_y u^2 + 2b_y u + c_y} \tag{14}$$

where $\boldsymbol{\beta} = \{a_x, b_x, c_x, d_x, a_y, b_y, c_y, d_y\}$. However, since the starting pose at $u = 0$ and the final position of the robot at $u = 1$ are fixed, only a_x, a_y, c_x remain as free spline parameters and thus $\boldsymbol{\beta}$ is reduced to $\boldsymbol{\beta} = \{a_x, a_y, c_x\}$.

The Algorithm

With the above path parameterisation, we are ready to formulate the BO algorithm, which is outlined in Algorithm 1.

In each iteration, the algorithm starts in line 2 by choosing the set of spline parameters $\boldsymbol{\beta}$ that maximise the acquisition function $R[\mathcal{C}(u, \boldsymbol{\beta})]$ over the GP power-consumption prior model. Then, the robot executes the path in line 3, collecting a set of observations $\mathcal{O} = \{(\mathbf{x}_i, z_i)\}_{i=1}^n$, and updates the GP appending the new samples to its observations set in line 4.

Algorithm 1: BO Path Planner

Input:
- $\mathcal{S}$: spline coefficients sampling domain
- N: number of iterations
- (x_0, y_0, θ_0): starting pose
- (x_f, y_f): goal location

```
1 for i = 1..N do
2     β* ← argmax_{β∈S} R[C(u, β)]
3     O ← Sample along C(u, β*)
4     Update the GP with each observation in O
```

---

5 Experiments

In order to evaluate our approach, we first performed a set of experiments to test the accuracy of the GP model estimations and its integration with our BO path-planning algorithm using synthetic data in simulation. After that, we performed tests with a physical robot in an outdoor environment.

To avoid infeasible paths, we added constraints to the path selection in our experiments. The constraint functions check if the candidate path stays within the area specified for the experiment and if the path's maximum curvature would exceed a specific limit. These constraints are checked for each set of spline parameters evaluated during the acquisition function optimisation in Algorithm 1 using algorithms that support non-linear inequality constraints.

5.1 Simulation

We first present the simulation model we used to compute energy-costs that provide ground-truth values for a robot's power consumption before we present our simulation-based results.

Simulation Setup

In simulation, we model terrains as continuous height maps generated by means of Perlin noise [Perlin, 1985]. Then we use the following equation to compute synthetic ground-truth values for the robot's power consumption:

$$w(\mathbf{x}) = \begin{cases} (F_c + Mg\sin\phi(\mathbf{x}))v, & \phi(\mathbf{x}) \geq 0, \\ F_c(1 + \sin\phi(\mathbf{x}))v, & \phi(\mathbf{x}) < 0 \end{cases} \tag{15}$$

where M corresponds to the robot's mass, g is the gravitational acceleration magnitude, $\phi(\mathbf{x})$ is the local terrain slope angle, and F_c is a constant factor to simulate the effect of forces independent of the robot's pose on the terrain. Overall, this formulation simulates large efforts when the robot is moving uphill, having to overcome its own weight, and lower, but non-negative, efforts when it is moving downhill. We have also added

Gaussian noise with standard deviation σ_n to the power-consumption values provided as observations to the GP. Table 1 presents the settings used for the tests.

F_c	M	g	v	σ_n
200 N	100 kg	10 m/s^2	1 m/s	20 W

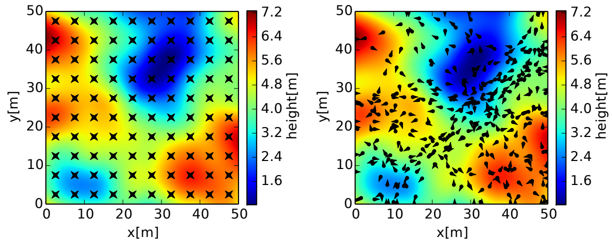
Table 1: Simulation settings

GP Model Accuracy

In a first test, we evaluated the quality of the GP power-consumption model predictions using two kinds of training sets. Figure 2a shows the first kind, with a training set containing sample poses uniformly spaced 5 metres apart on an (x, y)-grid pattern and using four values for orientation $\theta \in \{-135^o, -45^o, 45^o, 135^o\}$, yielding a total of 400 samples in the 50-by-50 metres terrain. Figure 2b shows the second kind of training set, containing random-walk samples with consecutive poses separated by 5 metres and with turns sampled from a uniform distribution limited between -30^o and 30^o. This set also contains the same amount of poses as the previous one, 400 in total. In both cases, we queried the GP with a test-set at a higher spatial resolution, with points spaced by 1 m over a grid, and at the same angular resolution of the first case, but with an offset, such that $\theta^* \in \{-90^o, 0^o, 90^o, 180^o\}$ for the queries. In total, the query set contains 10,000 query poses. We evaluated the accuracy in the predictions by computing the relative error between the predicted and the ground-truth values for each query pose. The root mean square (RMS) value of the relative error for the case in Figure 2a was 14.5%, and for the case in Figure 2b, it was 17.2%. In both cases, we utilised the same values for the GP hyper-parameters. Since the difference in the error levels between the ideal lawn-mower path of Figure 2a and the completely random path in Figure 2b is relatively small (2.7%) we are confident in the generalisation capabilities of the proposed GP power-consumption model, and can proceed with its integration into the BO framework.

Impact of λ on the Path Planner

We evaluated the performance of our BO path planner using different starting poses and end positions over multiple simulated terrains. We also tested the effect that different values of λ (see Equation 11), which controls the exploration-exploitation trade-off, have in the path planning. In Figure 3, we present a few test cases with the robot travelling from $A = (40, 40)$ to $B = (10, 10)$ and initial orientation $\theta_0 = -135^o$, i.e., the robot starts at the top-right corner and moves towards the bottom-left one. In this case, the shortest path is a straight line from A to B. However, this path goes through a valley, requiring a high power draw from the motors in the uphill area close to the goal. Consequently, in this scenario,



(a) Uniformly-spaced poses (b) Random path

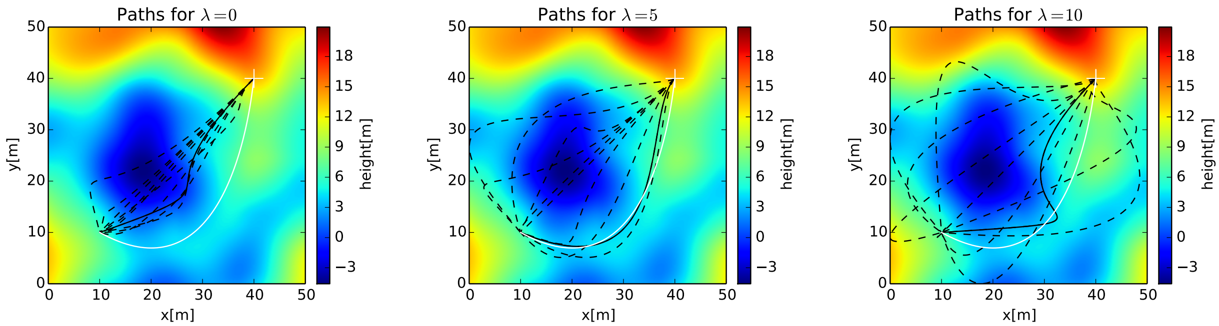
Figure 2: Example sets of sample poses used in tests with the GP model: on the left, a uniformly-spaced set, corresponding to a sweep pattern over the simulated terrain in 4 orientations; on the right, a random path where each pose is separated from the consecutive one by a fixed-length segment and a random turn angle.

the lowest-energy paths should not be taking a straight line from A to B, but going around the border of the valley to reach B spending less energy. The BO method should then be able to select paths to collect samples for the GP around that area, yielding better estimates for low energy-consumption paths. For comparisons, we computed the optimal spline path using simulated annealing on the noise-free cost function.

In all the test cases in Figure 3, we ran the algorithm for 10 iterations. For lower values of λ, the planner performs more exploitation than exploration. Consequently, the paths are usually very concentrated around a local low-energy region. These paths can also vary a lot among different runs of the test, depending on which low-energy path the algorithm finds first. However, for high values of λ the algorithm explores using longer paths, which can accumulate higher variance in the energy predictions. After 10 iterations, the estimates and the noise-free values for the lowest-energy paths found by the algorithm in each setting in Figure 3 are shown in Table 2. Among these test cases, the setting with $\lambda = 5$ found the best path in terms of energy consumption. We also observe that, due to noise and significant differences in height along the paths, the estimated energy value is slightly lower than its true value. However, for the case with $\lambda = 5$, the algorithm was still able to find a path close to the true optimum in terms of energy requirements, which requires 12396 J.

λ	0	5	10
$E[\mathcal{T}^*]$	14038 J	12663 J	16052.8 J
$\mu_E[\mathcal{T}^*]$	13437 J	12313 J	14584 J
$\sigma_E[\mathcal{T}^*]$	41 J	56 J	78 J

Table 2: True energy and its estimates for the lowest-energy paths found by the BO path planner in the test cases in Figure 3



(a) $\lambda = 0$ (b) $\lambda = 5$ (c) $\lambda = 10$

Figure 3: Height map of the simulated terrain with executed paths for different reward function λ values (see Equation 11). Each candidate path is plotted with a dashed line, with the lowest-energy path found by the BO algorithm represented by a solid black line in each case. The optimum path, found by simulated annealing over the noise-free cost function, is also plotted in a solid white line. The robot's start location $A = (40, 40)$ is marked with a white cross (+) symbol. In all the cases, the robot's initial heading is $\theta = -135^o$, facing the diagonal downwards. The goal point is also fixed, located at $B = (10, 10)$.

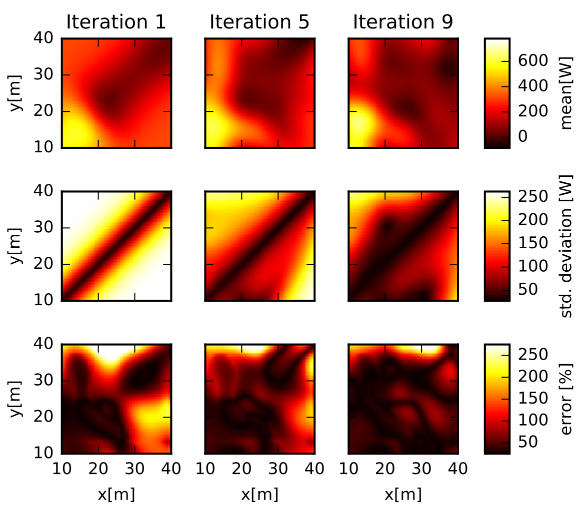


Figure 4: Evolution of the GP estimates for the robot's power consumption on poses with $\theta = -135^o$, main direction of movement, after iterations 1, 5 and 9 for the test case in Figure 3b. The upper plots show the estimated mean, $\mu_w(\mathbf{x})$. The middle plots show the standard deviation in the estimate, $\sigma_w(\mathbf{x})$. The lower plots show the relative error between the mean and the computed ground-truth costs, i.e. $|1 - \mu_w(\mathbf{x})/w(\mathbf{x})|$.

Though choosing a proper value for λ can be a trial and error process, in practice, a few points can be considered. For example, setting $\lambda = 2$ allows the algorithm to consider that a candidate path $\mathcal{T}$ could require an amount of energy in between $\mu_E[\mathcal{T}] \pm 2\sigma_E[\mathcal{T}]$ to be executed. This would correspond to a 95% confidence interval, if the GP model's hyper-parameters are properly set and the terrain slopes are not too steep to the approximation in the energy integration (Section 4.3). On the other hand, if these assumptions are not valid, the true energy value can be quite far from the estimate, then a higher λ could allow the algorithm to perform better.

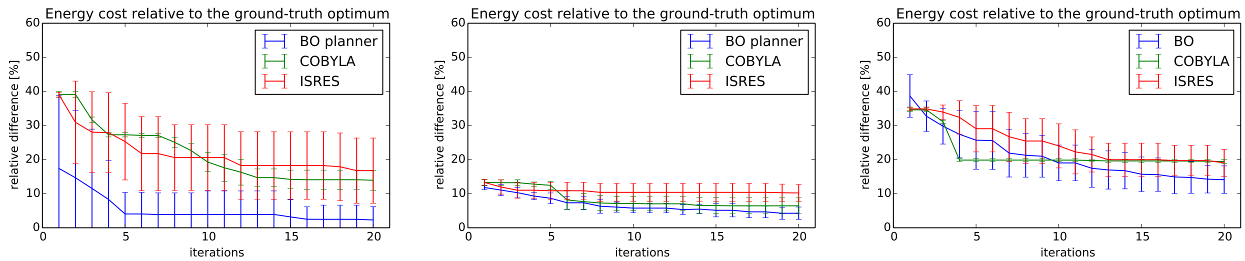
Figure 4 presents how the GP estimates evolve with the number of iterations in the paths for the above test case with $\lambda = 5$ (Figure 3b). As it can be seen in the error plots, the mean gradually converges to the ground-truth distribution in the area where the terrain is being explored. At the same time, the uncertainty about the estimations also reduces in the explored area. These results and the differences in paths seen for each setting in Figure 3 demonstrate that our BO algorithm is effectively exploring the terrain to learn a power-consumption model and select paths based on their reward, which involves both cost and information gathered.

Comparisons

Here we compare the performance of our BO path planner at the task of finding energy-efficient spline paths against the following non-linear optimisation algorithms:

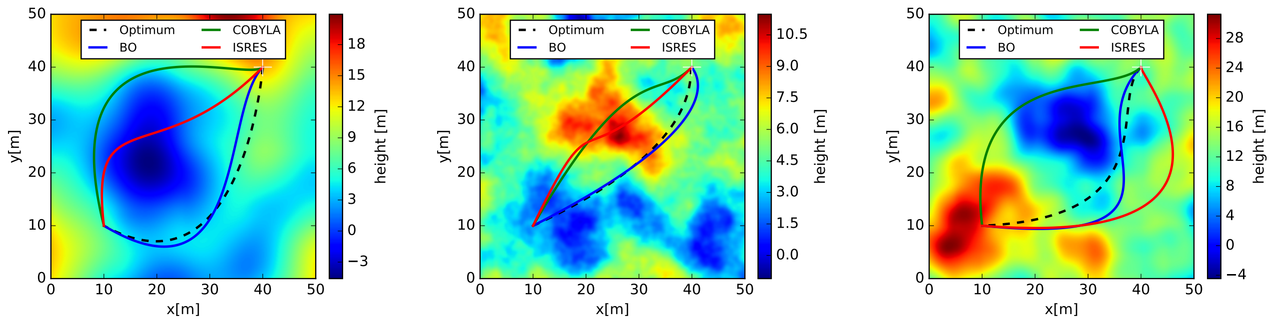
- *Improved Stochastic Ranking Evolution Strategy*(ISRES) [Runarsson and Yao, 2005]: a global optimisation algorithm; and
- *Constrained Optimization by Linear Approximations*(COBYLA) [Powell, 1998]: a local optimisation algorithm.

We used publicly-available implementations of these algorithms, provided by the package NLopt [Johnson, 2014]. They directly support non-linear inequality constraints, which were used to constrain the curvature of



(a) Performance on Terrain 1 (b) Performance on Terrain 2 (c) Performance on Terrain 3

Figure 5: Averaged relative energy cost between the current lowest-energy path found by each algorithm and the ground-truth optimum at each iteration. The error bars correspond to one standard deviation, computed over the re-runs of each algorithm. With only 5 iterations, the best paths found by the proposed BO planner are within 10% of the optimum. For these comparisons, we set $\lambda = 7$ (see Equation 11).



(a) Terrain 1 (b) Terrain 2 (c) Terrain 3

Figure 6: Examples of the lowest-energy paths found by each algorithm on the different terrains.

the paths and to limit them to the experimental area. We have set them to optimise spline parameters based only on the total energy spent for executing each path. Since both ISRES and COBYLA would have no prior information about the objective, their initial solution for the set of spline parameters was set to $\boldsymbol{\beta}_0 = \{0, 0, 0\}$, which corresponds to a straight line from start to goal. We have also used simulated annealing in these tests to find the true optimum spline path directly over the noise-free cost function for comparisons.

We evaluated the performance of each method on different terrains where the optimal spline path between the start and goal locations was non-trivial, i.e., not close to a straight line. This way, we could verify if the non-linear optimisation methods under comparison (COBYLA and ISRES) would be able to find paths close to the global optimum without the help of a GP model. Each method was configured to run for 20 iterations. The tests were repeated 10 times to ensure the consistency of the results. Figure 5 presents the performance of each algorithm in finding low-energy paths on 3 different terrain models. On average, the pure energy-based optimisation with COBYLA and ISRES seems to get stuck at a local minimum and, on average, is not able to find a better path within 20 iterations. On the other hand, since the BO planner builds a model of the terrain as it explores paths, it was able to find paths closer to the true optimum. Figure 6 presents some of the best paths found by each algorithm on each of those terrains.

5.2 Experiments with a Physical Robot

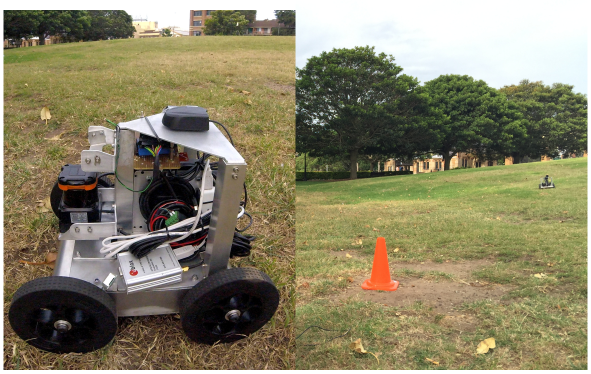
In this section we present results obtained in experiments with a physical robot. We utilised a small four-wheeled skid-steer robot, shown in Figure 7a, as a test platform. The robot is equipped with an on-board computer running ROS[1]. To measure power consumption, the robot carries sensors measuring battery voltage and total current drain. For localisation, wheel odometry, IMU and GPS data are fused using an extended Kalman filter (EKF) providing full 6D pose estimates.

The experiments were performed on an open grass field in an urban park, shown in Figure 8. For the start poses, we positioned the robot so that it was facing the north-east direction from within the area in the lower-left corner of the image. We set the goal location in

[1] Robot Operating System: `http://www.ros.org`

the upper-right corner of the image. This region had a small hill, partly seen in Figure 7b, between the robot's starting and end positions, so that paths going over the hill and paths going around it require different amounts of energy expenditure. We ran 12 iterations of our BO path-planning algorithm with the robot moving at a constant 0.7 m/s linear speed and at a maximum of 0.25 rad/s angular speed. For this experiment, we set $\lambda = 5$.

Figure 9 shows the predictions of our model for the energy required by each path chosen by BO immediately before executing the path and the actual measured energy consumption for the executed path. This measurement was made by directly integrating the instantaneous power-consumption measurements from the sensors over time. The plot shows that the predictions with the GP model gradually approximate the actual energy-consumption values, indicating that the model is improving over time, as seen in simulation. In addition, despite not much improvement due to the relatively simple terrain, towards the end, the paths' energy consumption are lower, demonstrating that the algorithm is able to find energy-efficient paths in the real-world scenario. This shows that our proposed method can adapt to multiple sources of noise and uncertainty in the physical world.



(a) Robot (b) Experiments area

Figure 7: Robot used and partial view of the area where the field experiments were performed

6 Conclusion

In this paper, we presented a Bayesian optimisation method to actively learn a model capable of predicting energy costs for paths between a start and a goal location in off-road terrains. The method does not require information about the terrain's characteristics, requiring only access to the robot's localisation and power-consumption measurements as it executes a path. As the model is built, the method is able to find energy-efficient paths, using a utility function designed to balance the trade off between exploration and exploitation.

Figure 8: Aerial view of the experiments area overlaid with the paths executed by the robot running the BO planner.

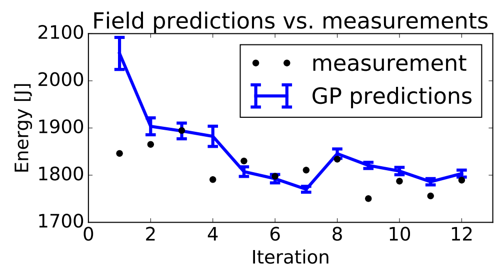


Figure 9: Predictions and measurements for the energy required to execute each path attempted by the algorithm in the field experiments.

In simulation, results demonstrated the performance of the method in building an accurate model of the robot's power-consumption over the terrain. Then, in comparisons with other non-linear optimisation algorithms, results showcase its capability of finding low-energy paths close to the global optimum within the family of parametric curves employed to represent paths.

A path-planning experiment with a physical robot demonstrated the applicability of our method to real-world environments, where a lot of the terrain and the vehicle's effects on power consumption are hard to model. Having a framework that does not depend on predefined models of these processes offers a great advantage in designing energy-efficient navigation systems. Therefore, we believe that the contributions in this paper can provide effective adaptable methods for a variety of outdoor applications where energy consumption is a major concern.

As future work, we intend to adapt this framework to include other variables not considered by the proposed power-consumption model, such as velocities. Other avenues include more complex path parameterisations and the integration of other constraints, such as obstacle avoidance, into the planning algorithm.

Acknowledgements

Author Rafael Oliveira was supported by CAPES (*Coordenação de Aperfeiçoamento de Pessoal de Nível Superior*), Brazil, scholarship BEX 13224/13-1.

References

[Amar and Bidaud, 1995] F. B. Amar and P. Bidaud. Dynamic Analysis of Off-Road Vehicles. In *International Symposium on Experimental Robotics (ISER)*, 1995.

[Anuntachai *et al.*, 2014] A. Anuntachai, O. Wongwirat, and A. Thammano. An application of ant algorithm for searching energy-efficient route a mobile robot takes using energy as a weighting factor. *Artificial Life and Robotics*, 2014.

[Brochu *et al.*, 2010] E. Brochu, V. M. Cora, and N. de Freitas. A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. Technical report, University of British Columbia, 2010.

[Brown, 2012] C. Brown. Autonomous vehicle technology in mining. *Engineering and Mining Journal*, 213(1):30–32, 01 2012.

[Carsten *et al.*, 2007] J. Carsten, A. Rankin, D. Ferguson, and A. Stentz. Global path planning on board the mars exploration rovers. In *2007 IEEE Aerospace Conference*, pages 1–11, March 2007.

[Ganganath *et al.*, 2014] N. Ganganath, C. Cheng, and C. K. Tse. Finding energy-efficient paths on uneven terrains. In *Congress on Mecatronics*, 2014.

[Johnson, 2014] S. G. Johnson. The nlopt nonlinear-optimization package. `http://ab-initio.mit.edu/nlopt`, 2014. Accessed: 2016-08-16.

[Kim and Kim, 2007] C. H. Kim and B. K. Kim. Minimum-energy translational trajectory generation for differential-driven wheeled mobile Robots. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 2007.

[Liu *et al.*, 2008] L. Liu, T. G. Crowe, and J. N. Bakambu. Efficient Exploration Algorithms for Rough Terrain Modeling Using Triangular Mesh Maps. In *IEEE International Conference on Robotics, Automation and Mechatronics*, 2008.

[Marchant and Ramos, 2014] R. Marchant and F. Ramos. Bayesian Optimisation for Informative Continuous Path Planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[Martin and Corke, 2014] S. Martin and P. Corke. Long-term exploration & tours for energy constrained robots with online proprioceptive traversability estimation. *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[Mei *et al.*, 2006] Y. Mei, Y. Lu, G. Lee, and C. Hu. Energy-efficient mobile robot exploration. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2006.

[Molino *et al.*, 2007] V. Molino, R. Madhavan, E. Messina, A. Downs, S. Balakirsky, and A. Jacoff. Traversability metrics for rough terrain applied to repeatable test methods. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2007.

[Morales *et al.*, 2006] J. Morales, J. Martinez, A. Mandow, A. Garcia-cerezo, J. Gomez-gabriel, and S. Pedraza. Power Analysis for a Skid-Steered Tracked Mobile Robot. In *IEEE International Conference on Mechatronics*, 2006.

[Perlin, 1985] K. Perlin. An image synthesizer. In *Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1985.

[Plagemann *et al.*, 2008] C. Plagemann, S. Mischke, S. Prentice, K. Kersting, N. Roy, and W. Burgard. Learning predictive terrain models for legged robot locomotion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008.

[Plonski *et al.*, 2013] P. Plonski, P. Tokekar, and V. Isler. Energy-efficient path planning for solar-powered mobile robots. *Journal of Field Robotics*, 2013.

[Powell, 1998] M. J. D. Powell. Direct search algorithms for optimization calculations. *Acta Numerica*, 7:287–336, 1998.

[Rasmussen and Williams, 2006] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning.* The MIT Press, Cambridge, MA, 2006.

[Runarsson and Yao, 2005] T. P. Runarsson and X. Yao. Search biases in constrained evolutionary optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(2):233–243, 2005.

[Sun and Reif, 2005] Z. Sun and J.H. Reif. On finding energy-minimizing paths on terrains. *IEEE Transactions on Robotics*, February 2005.

[Tokekar *et al.*, 2011] P. Tokekar, N. Karnad, and V. Isler. Energy-optimal velocity profiles for car-like robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.