Autonomous Excavation of Rocks Using a Gaussian Process Model and Unscented Kalman Filter

Autonomous Excavation of Rocks Using a Gaussian Process Model and Unscented Kalman Filter

Filippos E. Sotiropoulos and H. Harry Asada, *Member, IEEE*

***Abstract*—In large-scale open-pit mining and construction works, excavators must deal with large rocks mixed with gravel and granular soil. Capturing and moving large rocks with the bucket of an excavator requires a high level of skill that only experienced human operators possess. In an attempt to develop autonomous rock excavators, this paper presents a control method that predicts the rock movement in response to bucket operation and computes an optimal bucket movement to capture the rock. The process is highly nonlinear and stochastic. A Gaussian process model, which is nonlinear, non-parametric and stochastic, is used for describing rock behaviors interacting with the bucket and surrounding soil. Experimental data is used directly for identifying the model. An Unscented Kalman Filter (UKF) is then integrated with the Gaussian process model for predicting the rock movements and estimating properties of the rock. A feedback controller that optimizes a cost function is designed based on the rock motion prediction and implemented on a robotic excavator prototype. Experiments demonstrate encouraging results towards autonomous mining and rock excavation.**

***Index Terms*—Robot manipulation, mining robotics, robotics in construction, model learning for control, field robots, Gaussian process, UKF**

I. Introduction

AUTONOMOUS excavation promises to aid in meeting the ever-growing demand of mining and infrastructure industries. Excavators are among the most versatile machines in mining and construction environments [1], being used for a variety of earthmoving tasks. In particular, for large-scale open-pit mining and other construction works where a large ledge is blasted, excavators have to deal with large monolithic rocks [2] mixed with gravel and finer soil. Collecting these large rocks and transferring them to dump trucks and other locations requires a high level of skill; only experienced human operators can perform those challenging tasks.

In the literature of autonomous excavation a number of methods have been reported for improving productivity and energy efficiency as well as for controlling excavators under diverse soil conditions [2]–[10]. The primary challenge arises from the complexity of soil-bucket interactions which are difficult to model. Initial efforts used simple soil models [4], [5] to select digging trajectories. These methods rely on the Fundamental Equation of Earthmoving [11] and other simple models, which are limited in validity. Various feedback control methods have been developed to deal with uncertain soil-bucket interactions. Bernold [3] utilized impedance control to excavate fine soil and more recently Dobson et al. [2] proposed admittance control to deal with interaction forces while loading fragmented rock with a load hauler.

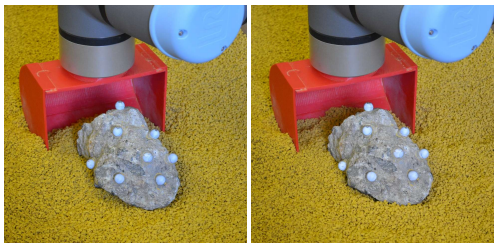
Collecting large monolithic rocks with an excavator is a new research challenge. Behaviors of large rocks are significantly different from those of homogeneous soil and gravel. The control objective, too, differs significantly. A discrete rock must be captured with a bucket, unlike the scooping of homogeneous soil, which is distributed and continuous. It should be noted that capturing a large rock resembles robotic manipulation where discrete rigid bodies are grasped and transferred. Inspired by the analogy between robot manipulation and rock excavation, we can gain insights into the handling of rocks. This opens up the possibility of applying the technologies developed for robotic manipulation to excavation.

Capturing a rock with a bucket can be classified as a non-prehensile manipulation task in the robotics literature [12]. Essentially, a bucket or a non-prehensile end-effector must make the rock move in a desired direction through interactions with the environment. This area of robotics research has seen significant progress in recent years. Improved physical models, which often harness data-driven methods make it possible to handle complex dynamics arising from friction [13]–[15]. Furthermore, other work has addressed dealing with environments with less structure due to cluttering [16], manipulating multiple rigid bodies interacting with each-other through modeling their interaction [17] and learning the control policy directly from video sensory data [18]. These methods, though, have in common that the manipulation involves direct contact of the end-effector with the manipulated objects and often leverages planar sliding to constrain the task. These assumptions do not apply to the current work where instead interaction between the bucket and rock can be transmitted through the surrounding soil. The behavior of soil is significantly complex and difficult to model. Furthermore, the motion of the rock will not be along a ground plane due to the uneven and changing nature of the soil supporting it.

II. Rock Excavation Problem

In this section the framework proposed to tackle the rock collection problem is outlined. Figure 1 shows a preliminary experiment of rock excavation. Case A is a successful trial where the rock was captured by the bucket as it moved forward. Case B, on the other hand, is an unsuccessful case. As the bucket moved forward in the same way as Case A, only soil was taken into the bucket, and the rock was pushed away. The rock behaviors are complex and confounding.

F. E. Sotiropoulos (fes@mit.edu) and H. H. Asada (asada@mit.edu) are with the Department of Mechanical Engineering, Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, USA. (The corresponding author is F. E. Sotiropoulos).



Case A Case B

Fig. 1. Initial experiments in excavation using heuristically generated trajectories. It was observed that generally similar conditions resulted in both success and failure for identical trajectories.

The following observations were made from our preliminary experiments:

- It is difficult to predict the rock movement relative to the bucket from a snap-shot image and data. The rock motion must be tracked.
- Physical models, such as terramechanics models, are difficult to use for rock movement prediction due to many unknown parameters and factors.
- The process is stochastic. Under the same conditions in a well-structured environment, the resultant rock motion may often differ, leading to success or failure in capturing a rock.

In an attempt to autonomously excavate rocks despite these challenges, here we present:

- A recursive filter that predicts the rock movement recursively from real-time measurement;
- A data-driven probabilistic model for designing the predictor; and
- A recursive optimal control law for controlling the bucket based on the prediction of the rock movement.

Figure 2 shows a schematic of the rock-bucket system considered in this paper. For simplicity, we consider a single, isolated rock, the size of which is approximately 30-80 % of the bucket size. It is smaller than a bucket, but is significantly larger than the grain of the surrounding soil. Furthermore, we consider the motion in only the x-z plane. This is consistent with the motion of the excavation machinery where the bucket moves along the direction perpendicular to the front edge except for a few special cases.

The task of collecting a large rock in the bucket must be defined in terms of quantifiable variables. For a rock to be successfully collected it must be brought to a position relative to the bucket where it is mostly encompassed by the bucket. This then allows a simple scooping motion to capture the rock so that it can be transported to another location to be dumped. Thus, the essence of the problem comes down to controlling the relative position between the bucket and rock. Given the restriction to the x-z plane the configuration of the rock relative to the bucket can be described by:

$$\mathbf{x}_r = \begin{bmatrix} x_{rb} & z_{rb} & \theta_r \end{bmatrix}^\top \tag{1}$$

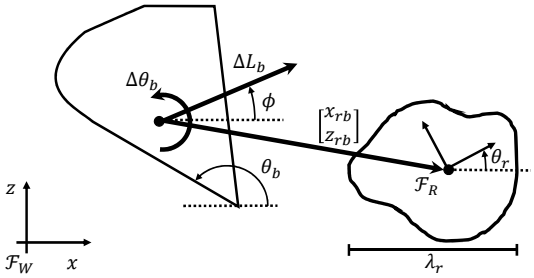
where the subscript rb indicates the relative position of rock and bucket in the world frame and θ_r is the orientation of the rock in the world frame. The rock orientation, θ_r, is also included since, although it is not explicitly controlled for, its evolution through time may help in performing the overall task by informing the rock property estimation as will be discussed in Section III. The system is controlled with the machine inputs:

$$\mathbf{u}_b = \begin{bmatrix} \Delta x_b & \Delta z_b & \Delta\theta_b \end{bmatrix}^\top \tag{2}$$

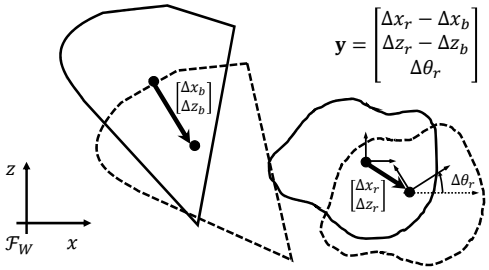
which describe incremental movements of the bucket in the world coordinate frame. Thus, the discrete state-transition equation is:

$$\mathbf{x}_{r,k} = f(\mathbf{x}_{r,k-1}, \mathbf{u}_{b,k-1}) + \varepsilon_{k-1} \tag{3}$$

which describes the transition between consecutive times which are Δt apart. ε_{k-1} is additive Gaussian noise. This is important since the task at hand is influenced by phenomena dominated by grain-to-grain mechanics as well as friction which on a macroscopic scale make the process inherently noisy. An analytic expression for the transition function f is difficult to determine given the complexity of the dynamics and thus a data-driven approach is taken as will be described in Section III.



(a)



(b)

Fig. 2. The input (a) and output (b) space of the data driven model.

III. Rock Motion Prediction Model

The discrete transition in (3) can also be described in terms of the change in the state variables:

$$\mathbf{x}_{r,k} = \mathbf{x}_{r,k-1} + \begin{bmatrix} \Delta x_{rb,k-1} \\ \Delta z_{rb,k-1} \\ \Delta\theta_{r,k-1} \end{bmatrix} + \varepsilon_k \tag{4}$$

The changes to the rock position and orientation are the random variables to be predicted:

$$\mathbf{y} = \begin{bmatrix} \Delta x_{rb} & \Delta z_{rb} & \Delta\theta_r \end{bmatrix}^\top \tag{5}$$

In the subsequent section we will obtain the probabilistic model for predicting the output $\mathbf{y}$.

A. Data-Driven Rock Motion Model

Gaussian Process (GP) regression [19] is used to generate the predictive model for the three outputs. The three outputs are treated independently so that a separate GP is used to model each output y_i.

The inputs to the GP model must inform how the output y_i is created. The current state of the rock-bucket system as well as the motion of the bucket will result in variation in the outputs. Therefore, we consider the following input variables, collectively denoted as vector ξ:

$$\xi = [\, x_{rb} \quad z_{rb} \quad \theta_r \quad \theta_b \quad \lambda_r \quad \phi \quad \Delta L_b \quad \Delta\theta_b \,]^\top \tag{6}$$

which is composed of $\mathbf{x}_r$ describing the configuration of the rock, $\mathbf{u} = [\phi \quad \Delta L_b \quad \Delta\theta_b\,]^\top$ which is a reformulated robot action where:

$$\phi = \arctan(\frac{\Delta z_b}{\Delta x_b}) \quad \text{and} \quad \Delta L_b = \sqrt{\Delta x_b^2 + \Delta z_b^2} \tag{7}$$

are the bucket movement direction (ϕ) and absolute change in bucket position (ΔL_b), respectively. The properties of the rock are approximated here simply by a characteristic length scale λ_r describing the longest dimension of the rock. Finally, θ_b is the bucket angle, which can have a significant effect on how the tool interacts with the rock and soil.

The training data-set for each output is composed of N_D instances of ξ and y_i: $\mathcal{D}_i = \langle \Xi, \mathbf{y}_i \rangle$ where $\Xi = [\xi_1, \xi_2, \ldots, \xi_{N_D}]$ and $\mathbf{y}_i = [y_{i,1}, y_{i,2}, \ldots, y_{i,N_D}]$. Then for a new input ξ_* the GP defines a Gaussian distribution over the predicted output y_{i*} [19]:

$$\mathbb{E}[y_{i*}] = \mathbf{k}_{i*}^\top [K_i + \sigma_{n,i}^2 I]^{-1} \mathbf{y}_i \tag{8}$$

$$\mathrm{Var}[y_{i*}] = k_{i**} - \mathbf{k}_{i*}^\top [K_i + \sigma_{n,i}^2 I]^{-1} \mathbf{k}_{i*} \tag{9}$$

where k_i represents the kernel function used for the GP for predicting output y_i. Subsequently, $\mathbf{k}_{i*}$ is the vector evaluating the kernel between the test input and the training inputs: $\mathbf{k}_{i*}[j] = k_i(\xi_*, \Xi[j])$, K_i is a square matrix evaluating the kernel between all training inputs: $K_i[j,l] = k_i(\Xi[j], \Xi[l])$ and finally k_{i**} is the kernel evaluated between the test input and itself: $k_{i**} = k(\xi_*, \xi_*)$. The kernel embodies the level of similarity between different inputs. $\sigma_{n,i}^2$ represents the variance of the process noise. A commonly used kernel is the squared exponential kernel and in this work we use the Automatic Relevance Determination (ARD) [19] variant which allows for similarity along each input dimension to have its own characteristic length scale. The kernel for each of the three independent models is:

$$k_i(\xi, \xi') = \sigma_{f,i}^2 \, \exp\left(-\frac{1}{2} (\xi - \xi')^\top W_i^{-1} (\xi - \xi') \right) \tag{10}$$

where $\sigma_{f,i}^2$ is the output variance and W_i, which is a diagonal matrix, defines the length scales along the input dimensions. The set of hyperparameters describing each model: $\boldsymbol{\gamma}_i = [\sigma_{n,i}^2 \quad \sigma_{f,i}^2 \quad W_i]$ are then optimized by minimizing the negative log likelihood for the training data:

$$\boldsymbol{\gamma}_i^* = \arg\min_{\boldsymbol{\gamma}_i} [-\log(p(\mathbf{y}_i | \Xi, \boldsymbol{\gamma}_i))] \tag{11}$$

Conjugate gradient descent is used to perform the optimization. For the purpose of brevity the independent models for each output will be combined such that the predicted output mean and covariance are:

$$\begin{aligned} \mathbb{E}[\mathbf{y}_*] &= \begin{bmatrix} \mathbb{E}[\Delta x_{rb*}] \\ \mathbb{E}[\Delta z_{rb*}] \\ \mathbb{E}[\Delta \theta_{r*}] \end{bmatrix} \\ &= \mathrm{GP}_\mu(\xi_* | \mathcal{D}) \end{aligned} \tag{12}$$

$$\begin{aligned} \mathrm{CoVar}[\mathbf{y}_*] &= \begin{bmatrix} \mathrm{Var}[\Delta x_{rb*}] & 0 & 0 \\ 0 & \mathrm{Var}[\Delta z_{rb*}] & 0 \\ 0 & 0 & \mathrm{Var}[\Delta \theta_{r*}] \end{bmatrix} \\ &= \mathrm{GP}_\Sigma(\xi_* | \mathcal{D}) \end{aligned} \tag{13}$$

Given the uncorrelated output assumption, $\mathrm{GP}_\Sigma(\xi | \mathcal{D})$ is diagonal.

B. Rock Length Estimation

Rocks are partially buried in soil and may be partially occluded. During a rock excavation operation the characteristic rock length cannot be measured directly. It is therefore estimated online recursively using a Bayesian filtering approach. More specifically, a Gaussian Process Unscented Kalman Filter (GP-UKF) [20] is applied. The use of an UKF is motivated by the nonlinear nature of the transition model. The estimation is enabled by augmenting the state space $\mathbf{x}_r$ to include the rock length:

$$\mathbf{x} = [\, x_{rb} \quad z_{rb} \quad \theta_r \quad \theta_b \quad \lambda_r \,]^\top \tag{14}$$

The UKF yields an estimate of the states ($\hat{\mathbf{x}}_k$) and covariance (Σ_k) of that estimate at each time step, k, based on previous estimates ($\hat{\mathbf{x}}_{k-1}$, Σ_{k-1}) given models for the transition ($\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + w_k$) and observation ($\mathbf{z}_k = h(\mathbf{x}_k) + v_k$), as well as matrices describing the process (Q_k) and measurement (R_k) noise covariance.

$$\hat{\mathbf{x}}_k,\, \Sigma_k \leftarrow \mathrm{UKF}(\hat{\mathbf{x}}_{k-1}, \Sigma_{k-1} \mid f,\, h,\, Q_k,\, R_k) \tag{15}$$

The mean (noiseless) state propagation is determined from the output mean of the GPs for the states associated with the rock configuration, an integration of the input $\Delta\theta_b$, and no change to the rock length:

$$\begin{aligned} \mathbf{x}_k &= f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \\ &= \mathbf{x}_{k-1} + \begin{bmatrix} \mathrm{GP}_\mu(\xi_{k-1} | \mathcal{D}) \\ \Delta\theta_b \\ 0 \end{bmatrix} \end{aligned} \tag{16}$$

As described in [20] the output covariance estimate provided by GP_Σ is used as the process noise for the UKF:

$$Q_k = \begin{bmatrix} \mathrm{GP}_\Sigma(\xi_{k-1} | \mathcal{D}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0}^\top & \sigma_{\dot{\theta}}^2 & 0 \\ \mathbf{0}^\top & 0 & \sigma_{\dot{\lambda}}^2 \end{bmatrix} \tag{17}$$

where σ_θ^2 is the process noise associated with the control of the bucket angle, σ_λ^2 is the process noise for the transition of the rock length estimate (which is set to zero). All other terms in the square Q_k matrix are zero. Most generally the observation model can also be nonlinear, and modelled as a GP, but presently given the experimental implementation (further described in Section V) a linear measurement model is used:

$$\mathbf{z}_k = h(\mathbf{x}_k) = \mathbf{x}_{r,k} \tag{18}$$

The measurement noise R_k is determined from the sensor specification. Now the estimate of the rock state (including the estimated characteristic length) can be used as input to the prediction model:

$$\hat{\xi} = [\,\hat{x}_{rb} \quad \hat{z}_{rb} \quad \hat{\theta}_r \quad \hat{\theta}_b \quad \hat{\lambda}_r \quad \phi \quad \Delta L_b \quad \Delta\theta_b\,]^\top \tag{19}$$

which can be used for predicting future motion based on the trained model and current observations.

IV. Rock Collection Control

As alluded to previously in Section I, a strategy for successfully collecting a rock in the bucket is to have the bucket and rock move so that their relative distance may get smaller, resulting in the bucket to surround the rock. The rock motion model is used to achieve this active control, while the initial positioning and final scooping actions are performed with given reference trajectories. As can be seen in Fig. 3, firstly the bucket is moved to a location just behind the rock. From there the bucket is controlled, using the predictive model, to minimize the relative distance between the rock and bucket. Finally, when the rock is considered "in" the bucket (as judged by a threshold distance, $r_S < r_R$ in Fig. 3c) a scripted scooping action is taken.

During the middle stage the control is based on selecting the control input at each time step such that the predicted relative distance between rock and bucket reference points is minimized at the next time step. In other words the inputs are chosen such that:

$$\mathbf{u}_k = \arg\min_{\mathbf{u}_k} J(\mathbf{u}_k|\mathbf{x}_k) \tag{20}$$

where the cost J is defined as:

$$J = \mathbb{E}[(x_{rb}+\Delta x_{rb})^2 + (z_{rb}+\Delta z_{rb})^2] \tag{21}$$

$$= \mathrm{Var}[\Delta x_{rb}] + \mathrm{Var}[\Delta z_{rb}] + (x_{rb}+\mathbb{E}[\Delta x_{rb}])^2 + (z_{rb}+\mathbb{E}[\Delta z_{rb}])^2 \tag{22}$$

This optimization is performed at each time-step.

V. Experiment

In this section the procedure for implementing and testing the methods described in previous sections is presented. The experimental implementation has two key aspects. The first is the procedure for generating the training data with which to train the GP. Then, using the simple controller specified in Section IV the feasibility for use in rock excavation tasks is evaluated.

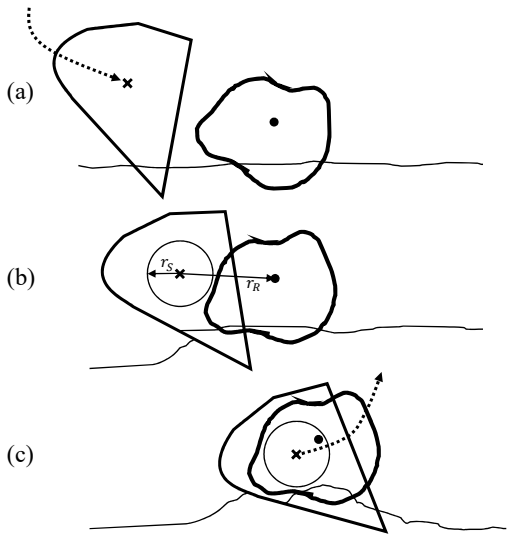


Fig. 3. The complete rock collection is performed in 3 stages. Firstly (a) the bucket is moved to a reasonable position behind the rock. Then the bucket motion is controlled using the data-driven model (b) so as to minimize the distance r_R between rock and bucket. Finally (c) when r_R falls below a threshold r_S a scooping action is performed.

A. Experimental Setup

The experimental setup shown in Fig. 4 was developed to test the proposed methods. The robot used is a Universal Robots UR10e, which has been fitted with a 3D printed bucket end-effector. The rock motion is measured with an OptiTrack Flex3 motion capture system which has been calibrated with respect to the robot. Furthermore, the granular "soil" material surrounding the rock is EPDM rubber granules approximately 3mm in size.

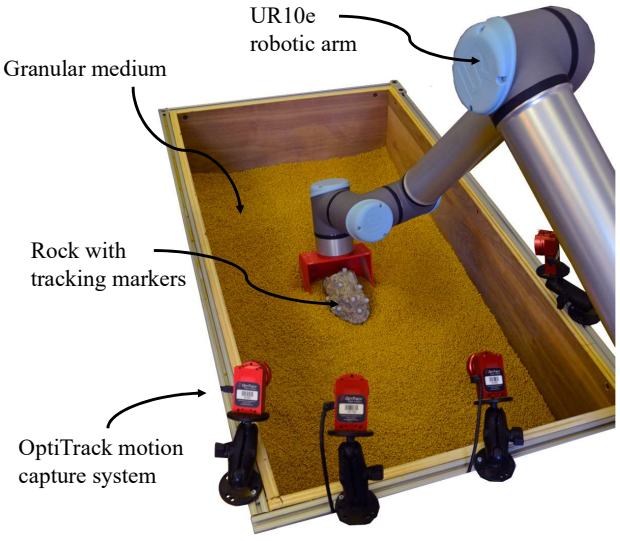


Fig. 4. Experimental setup used for data acquisition and experimentation. A digging pit environment is created where a robotic arm fitted with a bucket can excavate rocks surrounded by granular material. An OptiTrack motion capture system tracks the motion of the rock.

B. Training Data

Training data is collected by performing sequences of incremental random bucket movements that cause interactions with a rock and the surrounding soil, and measuring the resultant rock displacements. Firstly, the bucket is moved so that the bucket center is aligned with the center of the rock in the y-direction and offset by a small random amount in the x-direction (as in Fig. 3a). The bucket is then moved repeatedly where each component of $\mathbf{u}_b$ is selected randomly from a uniform distribution with reasonable limits. The bucket motions, $\mathbf{u}_b$, are executed at a piece-wise constant velocity over the time-step $\Delta t = 1$sec. 20 consecutive increments are performed before the rock and bucket are reset[1]. Two rocks of different sizes were used in training.

C. Robot Control

The control algorithm is run at the same frequency as the training data increments: 1Hz. The new commands are calculated as well as rock size estimation is performed at this frequency. The low level position tracking controller for the robot arm uses a high stiffness impedance controller to deal with soil-bucket interactions.

VI. Results and Discussion

In this section we evaluate the model and its effectiveness for moving the rock into the bucket so that it can be collected successfully.

A. Model Evaluation

We evaluate the performance of the models in terms of two metrics commonly used to evaluate probabilistic models. These are the normalized means squared error (NMSE) and the normalized log probability density (NLPD) [13], [21]:

$$NMSE = \frac{\sum_{j=1}^{n_*}(y_{*j} - \hat{y}_{*j})^2}{\sum_{j=1}^{n_*}(y_{*j} - \bar{y})^2} \tag{23}$$

$$NLPD = -\frac{1}{n_*}\sum_{j=1}^{n_*} \log p(y_{*j}|\mathcal{D}) \tag{24}$$

Here y_{*j} is the j-th sample in a testing data-set, $\hat{y}_{*j}$ is the corresponding mean prediction from the model, n_* are the number of test samples and $\bar{y}$ is the mean training observation used to normalize the NMSE. The NMSE evaluates the accuracy of the mean prediction and is often useful in evaluating the utility of the model where the prediction is directly used for control. In contrast, NLPD evaluates the probability that the test observations are to come from the model output distribution. In other words it penalizes predictions with variance that is too high or low. This is important in the estimation framework where the prediction variance is used in the UKF update.

Fig. 5 shows how the NLPD and NMSE vary as a function of training data-set size. As is to be expected both metrics decrease with increasing training data-set size, however the performance appears to approach saturation as one goes above 500 training points. This is equivalent to only 25 test scoops.

It is also worth noting the considerably worse performance for the prediction of the change in rock orientation ($\Delta\theta_r$) compared to the two translation predictions. This could be due to the fact that current inputs (ξ), do not inform the rotational dynamics adequately. For example, the distribution of soil providing support in front and behind the rock may heavily influence the direction and extent in which it tips, as might the location of its center of mass.

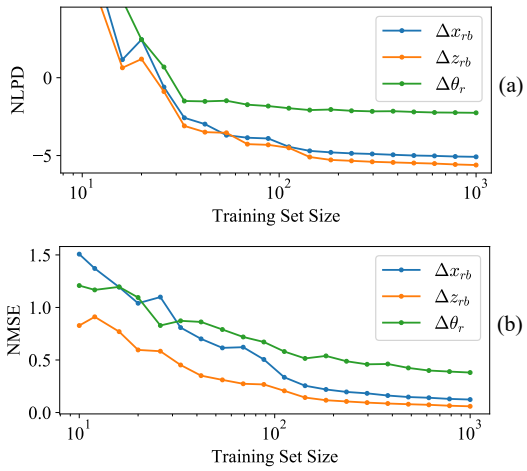


Fig. 5. NLPD (a) and NMSE (b) with varying training set size. Each point on the graphs represents the mean of 25 evaluations of the metric with random test sets of 150 samples and a randomly chosen training set of the remaining available data points.

An implicit assumption made in including the rock length in the prediction model is that the rock size has a measurable effect on the motion. This hypothesis is tested by evaluating the NLPD and NMSE while setting the rock length to the ground truth value vs. then using the mean rock length of all the training data. As can be seen in Table I the NLPD and NMSE is lower for all outputs when the true rock length is included.

TABLE I
Effect of Rock Length Accuracy on Model Prediction

	NLPD			NMSE		
	Δx_{rb}	Δz_{rb}	$\Delta\theta_r$	Δx_{rb}	Δz_{rb}	$\Delta\theta_r$
True Rock Length	-5.0	-5.6	-2.3	0.12	0.06	0.37
Mean Rock Length	-4.7	-4.8	-2.2	0.19	0.16	0.40

B. Rock Length Estimation

The estimation is tested on the rocks used for training. Each trial run consists of moving forward with random trajectories similarly to the data collection procedure. The mean and standard deviation for 30 tests on each rock are shown in Fig. 6. The dashed lines indicate ground truth lengths. As can be seen in Fig. 6 the mean rock estimate converges to a value in the neighborhood of the ground truth value.

[1]A video file includes example of training data cycle and other experimental results.

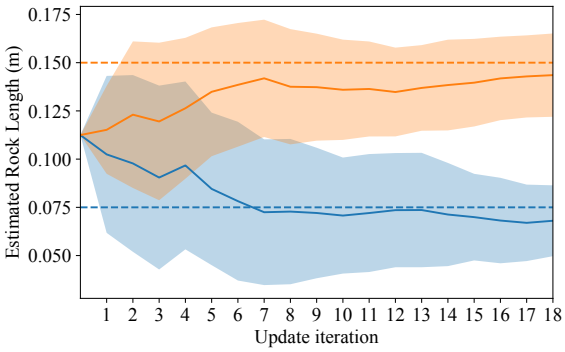


Fig. 6. Mean and standard deviation of the characteristic rock length estimate determined online. 30 trials for two different rocks are tested. The dashed line indicates the ground truth rock length.

C. Rock Collection Control

Here results as they pertain to evaluating the model for moving the rock into the bucket are presented. Firstly, in Fig. 7 the optimal bucket direction and magnitude of translation (in terms of the optimization in Section IV) are shown for combinations of x_{rb} and z_{rb}. The other variables in ξ are fixed at their respective mean values for the training data. It can be seen that for the majority of the space the optimum ΔL_b is in fact the maximum value of 0.02m. This makes sense since as long as there is an available direction of movement which is not increasing the overall distance between bucket and rock, then maximizing the bucket motion would likely result in the greatest decrease in relative distance.

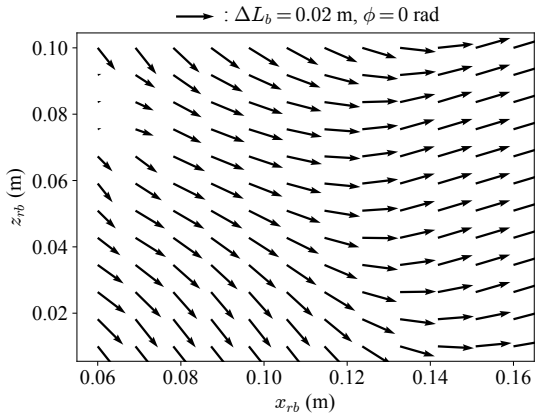


Fig. 7. The optimal ϕ and ΔL_b for varying relative rock-bucket position.

The effectiveness of the proposed method is tested by repeatedly attempting to excavate several rocks. The collection success rate for various conditions is shown in Table II.

TABLE II
SUCCESS RATE OF ROCK CAPTURE (N=20).

	Success Rate
Training rocks and **flat** soil	1.00
Other rocks and **flat** soil	0.90
Training rocks and **disturbed** soil	0.85

For the conditions which were used to train the model the controller achieves a perfect success rate for the test performed. It is when either the rock types or the surrounding soil are varied to situations not seen in training that performance deteriorates. When the distribution of soil is varied the controller still achieves positive results in the large majority of cases. The controller was able to compensate for the rock motion by choosing new optimal directions to move in at every time step. In some extreme instances, for example where there was a large void right behind the rock or the material distribution results in large lateral motion, the algorithm does not react sufficiently and the rock falls out of the bucket.

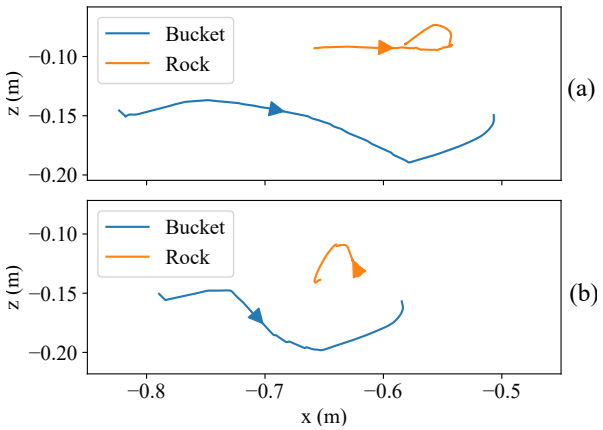


Fig. 8. Sample successful bucket and rock trajectory for both a large, 22cm (a) and small, 5cm (b) rock.

Fig. 8 shows two samples of successful rock collection trajectories. As can be seen, the directions the bucket and rock move for two rocks of different sizes differ substantially. Nonetheless, the rocks are gathered successfully in both cases due to the adaptation of the controller.

VII. Conclusion and Further Work

This paper presents the integration of a GP model with a UKF for the automated excavation of rocks. The GP model captures nonlinear and stochastic properties of the interaction between bucket, rock and soil, and allows an optimal control law to then guide the rock towards the inside of the bucket. We show that properties of the manipulated rock can be estimated online, by observing its displacement in response to the motion of the bucket.

In the authors' opinion this work opens multiple avenues for further development. One major aspect that was neglected is the effect of soil distribution when making predictions of the rock movement. For example, an angled soil surface can provide additional resistance which can push the rock into the bucket. The method presented in this paper could be expanded to consider a description of the distribution of surrounding soil and how this varies and can be leveraged for the task at hand.

In this work we targeted the identification of one property of the rock, namely its length, however an exploration into

what are the most informative parameters to identify for the purpose of improving collection performance, could be fruitful. For example, as opposed to simply identifying the rock length an estimate of the location of the center of mass of the rock may reveal more about its motion.

REFERENCES

[1] S. Dadhich, U. Bodin, and U. Andersson, "Key challenges in automation of earth-moving machines," *Autom. Constr.*, vol. 68, pp. 212 – 222, 2016.

[2] A. A. Dobson, J. A. Marshall, and J. Larsson, "Admittance control for robotic loading: Design and experiments with a 1-tonne loader and a 14-tonne load-haul-dump machine," *J. Field Rob.*, vol. 34, no. 1, pp. 123–150, 2017.

[3] L. E. Bernold, "Motion and path control for robotic excavation," *J. Aerosp. Eng.*, vol. 6, no. 1, 1993.

[4] S. Singh, "Synthesis of tactical plans for robotic excavation," Ph.D. dissertation, Robotics Institute, Carnegie Mellon Univ., Pittsburgh, June 1995.

[5] O. Luengo, S. Singh, and H. Cannon, "Modeling and identification of soil-tool interaction in automated excavation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 3, Oct 1998, pp. 1900–1906.

[6] H. Cannon and S. Singh, *Experimental Robotics VI*. London: Springer, 2000, ch. Models for automated earthmoving.

[7] D. A. Bradley and D. W. Seward, "The development, control and operation of an autonomous robotic excavator," *J. Intell. Rob. Syst.*, vol. 21, no. 1, pp. 73–97, January 1998.

[8] G. J. Maeda, I. R. Manchester, and D. C. Rye, "Combined ILC and disturbance observer for the rejection of near-repetitive disturbances, with application to excavation," *IEEE Trans. Contr. Syst. Technol.*, vol. 23, no. 5, pp. 1754–1769, Sept 2015.

[9] D. Jud, G. Hottiger, P. Leemann, and M. Hutter, "Planning and control for autonomous excavation," *IEEE Rob. Autom. Lett.*, vol. 2, no. 4, pp. 2151–2158, Oct 2017.

[10] F. E. Sotiropoulos and H. H. Asada, "A model-free extremum-seeking approach to autonomous excavator control based on output power maximization," *IEEE Rob. Autom. Lett.*, vol. 4, no. 2, pp. 1005–1012, April 2019.

[11] A. R. Reece, "Paper 2: The fundamental equation of earth-moving mechanics," *Proc. Inst. Mech. Eng.*, vol. 179, no. 6, pp. 16–22, 1964.

[12] M. T. Mason, "Progress in nonprehensile manipulation," *Int. J. Rob. Res.*, vol. 18, no. 11, pp. 1129–1141, November 1999.

[13] M. Bauza and A. Rodriguez, "A probabilistic data-driven model for planar pushing," in *Proceedings of the International Conference on Robotics and Automation*, 2017.

[14] M. Bauza, F. Hogan, and A. Rodriguez, "A data-efficient approach to precise and controlled pushing," in *Proceedings of the Conference on Robot Learning*, 2018.

[15] A. Byravan and D. Fox, "Se3-nets: Learning rigid body motion using deep neural networks," in *Proceedings of the International Conference on Robotics and Automation*, May 2017, pp. 173–180.

[16] M. R. Dogar and S. S. Srinivasa, "A planning framework for non-prehensile manipulation under clutter and uncertainty," *Autonomous Robots*, vol. 33, no. 3, pp. 217–236, Oct 2012.

[17] A. Ajay, M. Bauza, J. Wu, N. Fazeli, J. B. Tenenbaum, A. Rodriguez, and L. P. Kaelbling, "Combining physical simulators and object-based networks for control," in *Proceedings of the International Conference on Robotics and Automation*, May 2019, pp. 3217–3223.

[18] C. Finn, I. Goodfellow, and S. Levine, "Unsupervised learning for physical interaction through video prediction," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16, USA, 2016, pp. 64–72.

[19] C. E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge: the MIT Press, 2006.

[20] J. Ko and D. Fox, "GP-BayesFilters: Bayesian filtering using gaussian process prediction and observation models," *Autonomous Robots*, vol. 27, no. 1, pp. 75–90, Jul 2009.

[21] J. Quinonero Candela, C. Rasmussen, F. Sinz, O. Bousquet, and B. Schölkopf, "Evaluating predictive uncertainty challenge," in *Machine Learning Challenges: Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, Max-Planck-Gesellschaft. Berlin, Germany: Springer, Apr. 2006, pp. 1–27.