Optimization-Based Framework for Excavation Trajectory Generation

Yajue Yang, Pinxin Long, Xibin Song, Jia Pan, and Liangjun Zhang

***Abstract*—In this letter, we present a novel optimization-based framework for autonomous excavator trajectory generation under task-specific constraints. Traditional excavation trajectory generators over-simplify the geometric trajectory parameterization thereby limiting the space for optimization. To expand the search space, we formulate a generic task specification for excavation by constraining the instantaneous motion of the bucket and adding a target-oriented constraint to control the amount of excavated soil. The trajectory is represented with a waypoint interpolating spline. Time intervals between waypoints are relaxed as variables to facilitate generating the time-optimal trajectory in one stage. Experiments on a real robot platform demonstrate that our method is adaptive to different terrain shapes and outperforms other optimal path planners in terms of the minimum joint length and minimum travel time.**

***Index Terms*—Mining robotics, robotics in construction, trajectory optimization.**

I. Introduction

HEAVY machinery, such as hydraulic excavators, is widely used in construction, mining and many other scenarios. Nowadays, excavators are mainly controlled by humans operators. In order to address the labor shortage for skillful operators, save the increasing labor cost, and improve working conditions especially for hazardous environments, there exists a strong need for developing autonomous excavators and automating the overall excavation process.

This work focuses on generating an optimal trajectory for a single excavation, which is a fundamental task of the overall automation. A trajectory capable of digging the soil out must be constrained to follow some scooping-like pattern. Thus, what an excavation trajectory generator does is to formulate such constraints at first and then search for trajectories that satisfy the constraints. The constraint formulation is not trivial because, on one hand, there is no clear-cut mathematical description of the scooping-like pattern, but on the other hand, over-simplifying the description would restrict the room for optimization. In industry, it is important to optimize the excavation trajectory for various purposes. For example, while guaranteeing to excavate the soil sufficiently, it is desired to maximize the time efficiency (productivity) or minimize the energy consumption [1]. Inspired by this fact, we aim to find a constraint formulation allowing a large space for optimization.

Previous work on trajectory generation for excavation usually separates a bucket path into three phases: linear penetration, dragging and scooping, of which the first two are restricted to pure translation movement for simplicity [2]–[5], as depicted in Fig. 1(a). The three-phase (TP) pattern, according to the following analysis, only represents a subgroup of all possible excavation trajectories thereby heavily reducing the space for optimization. To generalize the characterization, we distill the essence of a successful excavation:

1) the bucket motion continuously causes failure of the soil in front of the bucket, *i.e.* the front soil failing to retain its original geometric shape and displacement [6];
2) the failed soil accumulates on or in front of the bucket bottom plate (the bucket shape is annotated in Fig. 1(b));
3) when the failed soil accumulates sufficiently, it will be rolled into the bucket and lifted out of the terrain.

Therefore, what matters for excavation is not the geometric shape of the bucket path as parameterized in the TP pattern, but the instantaneous motion that causes the desired physical effect of the soil.

For a typical bucket, two orthogonal instantaneous motions exist to cause the failure and accumulation of the front soil:

1) the bucket penetrates through the soil along the direction tangent to the bottom plate, which leads to a rupture of the soil contacting with the bottom plate, as depicted in Fig. 1(c);
2) the bucket separates and presses the front soil along the direction normal to the bottom plate, which leads to a rupture of a wedge area and the soil accumulation [7], as shown in Fig. 1(d).

Digging trajectories are varied because the bucket motion at each timestep during the entire excavation process could combine the instantaneous penetration and separation in many different ways. Fig. 1(e) gives an example of a valid excavation trajectory with different composite motions at each timestep. In contrast to characterizing the geometric bucket path at the macro level, we propose to regulate the trajectory more flexibly

Manuscript received October 16, 2020; accepted January 31, 2021. Date of publication February 9, 2021; date of current version February 25, 2021. This letter was recommended for publication by Associate Editor J. Kim and Editor Y. Choi upon evaluation of the reviewers' comments. This work was supported in part by Innovation and Technology Fund (ITF) ITS/457/17FP and in part by General Research Fund (GRF) 11207818, 11202119. *(Corresponding authors: Liangjun Zhang; Jia Pan.)*

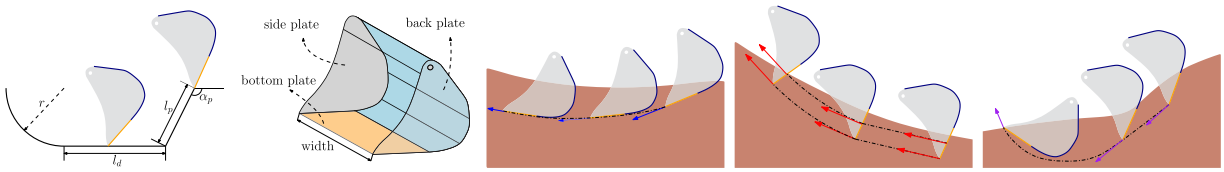
Yajue Yang is with the Department of Biomedical Engineering, City University of Hong Kong, Hong Kong, China. This work was done when she was an Intern at Baidu Research, Baidu Inc., Beijing, China (e-mail: yawnyoung@gmail.com).

Pinxin Long, Xibin Song, and Liangjun Zhang are with the Baidu Research, Baidu Inc., Beijing, China (e-mail: pinxinlong@gmail.com; song.sducg@gmail.com; liangjun.zhang@gmail.com).

Jia Pan is with the Department of Computer Science, the University of Hong Kong, Hong Kong (e-mail: panjia1983@gmail.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2021.3058071, provided by the authors.

Digital Object Identifier 10.1109/LRA.2021.3058071



(a) three-phase pattern (b) bucket shape (c) penetration mode (d) separation mode (e) composite mode

Fig. 1. (a) Three-phase pattern and its parameterization. (b) Typical bucket shape. (c) The bucket penetrates through the soil along the direction tangent to the bottom plate. The tangential velocity is denoted with the blue arrow. (d) The bucket breaks and moves the front soil by pure separation *i.e.* moving in the normal direction of the bottom plate. The normal velocity is denoted with the red arrow. (e) The bucket velocity at each timestep is composed of the tangent and normal velocity. The resultant velocity is denoted with the purple arrow.

by constraining micro instantaneous motions and their relationships, which enables optimization over a wider trajectory space.

To search for the optimal trajectory under such constraints of the end-effector (*i.e.* bucket), we formulate the excavation task as a trajectory optimization problem [8]. This approach enables us to handle the complex nonlinear constraints with advanced optimization techniques, such as sequential quadratic programming [9] and stochastic initial trajectory generation [10]. Furthermore, we can naturally consider the objective for maximizing the time efficiency and the constraint of the trajectory travel time for synchronizing with other equipment. Similar to [11], we introduce time intervals as variables into the optimization problem, which enables simultaneous generating a constraint satisfied path and minimizing the travel time. In contrast, the classical method of time-optimal trajectory generation is to separate this process into two steps, which is known as the two-stage approach [12]. It first produces a geometric path, and then generates the time-optimal trajectory that exactly follows the path under velocity and acceleration limits or constraints induced by dynamics singularities [13]–[15]. We refer our method as one-stage in the following text.

Main Contributions: In summary, this paper presents a novel optimization-based framework of excavation trajectory generation. Specifically, a continuous trajectory in joint configuration space is represented with discrete waypoints interpolated by a polynomial spline. Apart from the waypoints, time intervals are introduced as variables, which facilitates explicitly arranging the time profile of trajectories. A group of process-oriented geometric constraints is imposed to ensure that the bucket motion obeys the excavation principle. To collect a sufficient amount of soil, we add a target-oriented constraint, *i.e.* swept volume factor that indicates the estimated amount of the excavated soil. This constraint-based task specification guarantees the success of excavation and allows large room for trajectory optimization. The framework is suitable for various objective functions *e.g.* minimum time, minimum joint displacement and minimum torque, etc.

We implement the proposed framework on a UR5 manipulator arm to demonstrate that the generated trajectories are able to excavate sufficient mass of soil for terrains in different shapes. We first show that the minimal length of trajectories generated by our method are 60% shorter than the aforementioned TP methods. Then we compare our one-stage method with the two-stage method in time-optimal excavation trajectory generation. The experiment result shows that trajectories generated by the one-stage method cost 18% less time on average.

II. Related Work

Previous studies on the optimal excavation trajectory generation generally minimize the travel time or torque consumption [16]–[18]. The bucket-tool interaction is considered in two aspects: the resistance force from the soil and the displacement of the soil. While most methods adopt analytical models to estimate the resistance force [19], Singh examines learning methods for predicting the resistance force [20]. Besides, some work directly simulates the overall bucket-tool interaction behavior with numerical techniques *e.g.* the discrete element method and the continuum model [21], [22]. Although a lot of efforts have been made, the bucket-tool interaction modeling is still very challenging and remains an open question. Some work sidesteps this problem by developing a real-time controller adaptive to the feedback on the soil status [5], [23]. These controllers basically follow a reference trajectory (usually the TP pattern) and modulate the execution according to observation and prediction. In [23], the controller switches from the penetration to the dragging once it observes a large resistance force. In [5], whether to drag or curl in the next step is determined by predicting the bucket filling of the two motions. These methods do not consider the global optimization of the trajectory. Son *et al.* modulates the human expert emulating trajectory in real-time to cope with underground uncertainties [24].

As one of the optimal trajectory generators, our method is distinguished from others by the way constraining the trajectory for excavation. Zou *et al.* parameterize trajectories in the task space with the B-splines instead of the TP pattern [18]. Being faced with the potential problem of failing to solve inverse kinematics, they resort to only parameterizing the position bucket trajectory – relaxing the orientation constraint – and resolving the redundancy by introducing the self-motion parameter constraint. In contrast, our method directly optimizes the trajectory in the joint space and explicitly formulates constraints on 3D bucket motion including both the position and orientation. Although Kim *et al.* also parameterize trajectories in the joint space [17], they regard the digging process as a point-to-point motion without explicitly imposed path constraints. As a consequence, the generated trajectory is prone to undesired movement such as pressing the soil.

III. Method

In this section, we present our time variable optimization-based framework of excavation trajectory generation. We represent the trajectory with a waypoint-interpolating spline. To make

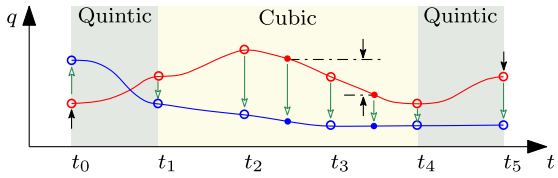


Fig. 2. Illustration of the interpolating spline and the path constraints enforcement. A spline curve interpolates all the waypoints marked by hollow circles. Dotted circles represent intermediate points on the spline. The background of each region where a quintic polynomial locates at is filled with a light green, while the cubic region is filled with a light yellow. As to the optimization process, we use the red and the blue to distinguish the spline at the first and the second iteration respectively. Constraints on the waypoints and intermediate points are imposed at the first iteration and denoted by black arrows, which push the waypoint at t_0 upwards and the waypoint at t_5 downwards. Besides, the distance between the two intermediate points between t_2 and t_4 are required to be diminished. The updates to positions of the 6 waypoints are denoted by the green arrows.

the trajectory capable of excavation in practice, we impose task-specific path constraints on the waypoints and the intermediate points sampled on the spline.

A. Problem Statement

A trajectory $\mathbf{q}(t) : \mathcal{R} \mapsto \mathcal{C} \subseteq \mathcal{R}^d$ is a continuous function mapping time t to a d-dimensional robot configuration state $\mathbf{q}$ within a constrained region $\mathcal{C}$. Trajectory optimization refers to minimizing some objective functionals under a set of constraints over the trajectory:

$$\text{Minimize}_{\mathbf{q}(t)} \quad \int_0^{t_f} L(\mathbf{q}(t))dt \tag{1}$$

$$\text{Subject to} \quad \mathbf{g}(\mathbf{q}(t)) \leq \mathbf{0} \tag{2}$$

$$\mathbf{h}(\mathbf{q}(t)) = \mathbf{0} \tag{3}$$

where t_f denotes the travel time. To make the problem tractable, $\mathbf{q}(t)$ is discretized into a finite sequence of waypoints: $\mathbf{Q} = \{\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_n\}$ with the time knots ranging from t_0 to t_n. With $t_0 = 0$, we introduce n time intervals as optimization variables $T = \{\Delta t_0, \Delta t_1, \ldots, \Delta t_{n-1}\}$ where $\Delta t_i = t_{i+1} - t_i$. Consequently, the original problem is converted to a nonlinear programming over a multidimensional vector which is comprised of $d \times (n+1)$ joint position variables and n time variables.

We use a second-order differentiable polynomial spline $\mathbf{s}(t)$ interpolated between the discrete waypoints to represent an excavation trajectory. Although a cubic spline is sufficient for the requirement of smoothness, we adopt a mixed-order spline: the two segments at the ends are interpolated with quintic polynomials while the remaining waypoints are interpolated with a cubic spline, as shown in Fig. 2. In particular, the spline $\mathbf{s}(t)$ is formulated as follows:

- for $i = 0$ and $i = n-1$,

$$\mathbf{s}(t) = \sum_{j=0}^{j=5} \mathbf{a}_{ij}(t - t_i)^j, \quad \text{if } t_i \leq t < t_{i+1} \tag{4}$$

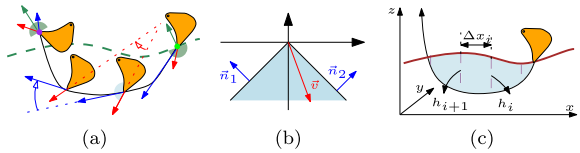


Fig. 3. (a) Overview of the trajectory constraints: The black curve denotes the bucket path, starting from the green point to the purple point which are anchor points. Blue arrows denote $\vec{t}$, and red arrows represent $\vec{h}$. The normal vector of terrain surface, depicted with the green arrow, confines $\vec{t}$ and $\vec{h}$ at both anchor points in the area denoted with the green region. As the hollow arrows describe, $\vec{t}$ and $\vec{h}$ are constrained to monotonously rotate clockwise along the path. The light blue region denotes the halfspace separated by $\vec{h}$ where $\vec{t}$ is limited to lie on. (b) Direction constraint: $\vec{v}$ is restricted to be located in the blue triangle region. $\vec{n}$s denote normal vectors to the hyperplanes. (c) Swept volume estimation: The blue region denotes the swept area for an excavation trajectory.

- for $0 < i < n-1$,

$$\mathbf{s}(t) = \sum_{j=0}^{j=3} \mathbf{a}_{ij}(t - t_i)^j, \quad \text{if } t_i \leq t < t_{i+1}. \tag{5}$$

To find $\mathbf{s}(t)$ interpolating $\mathbf{Q}$ involves solving a linear equation system in terms of $\mathbf{a}_{ij}$s. Given a $\mathbf{Q}$ and corresponding T, the coefficients $\mathbf{a}_{ij}$s of the cubic part (in Eq. (5)) can be uniquely determined using the method proposed in [25]. Subsequently, $\mathbf{a}_{ij}$s of the quintic segments (in Eq. (4)) are solved with certain constraints including the zero-speed constraint at each end of $\mathbf{s}(t)$ and the continuity requirement at each cubic-quintic intersection. In this way, every $\mathbf{a}_{ij}$ is actually a one-to-one function of $\mathbf{Q}$ and T, and so is every point on $\mathbf{s}(t)$ because $\mathbf{s}(t)$ is only characterized with $\mathbf{a}_{ij}$s. Thus, constraints on spline points are functions of $\mathbf{Q}$ and T. Fig. 2 shows how a path constraint is enforced in the optimization.

The integral objective function in Eq. 1 can be transformed into sum of terms accordingly. For example, the squared length of a trajectory can be represented as:

$$f(\mathbf{Q}, T) = \sum_{i=0}^{m} \|\mathbf{s}(t_{i+1}) - \mathbf{s}(t_i)\|^2 \tag{6}$$

where m denotes the number of samples. In the similar way, the time-related objective and constraint can also be formulated as:

$$\text{Minimize}_T \sum_{i=0}^{n-1} \Delta t_i \quad \text{and} \quad \sum_{i=0}^{n-1} \Delta t_i = t_f \tag{7}$$

where t_f denotes the desired finish time.

B. Excavation Task Specific Constraints

As mentioned in Section I, the key to successful excavation is sequentially combining instantaneous motions that cause failure and accumulation of the front soil with suitable geometric guidance and regulation. We propose below a list of rules of thumb on the excavation process, which constrains the instantaneous motions. For convenience, we refer to the direction of the bottom plate as $\vec{h}$ and the bucket tip translation direction as $\vec{t}$ in the rest of the text. As illustrated the Fig. 3(a), *Rules* for excavation include:

1) At both ends of the trajectory, the bucket tip position is constrained to touch the terrain surface;

2) At the first point, $\vec{\boldsymbol{t}}$ should be opposite to the normal vector of the terrain surface to enter into the terrain;
3) At the last point, $\vec{\boldsymbol{t}}$ should be along the normal vector of the terrain to leave;
4) Along the entire path, $\vec{\boldsymbol{t}}$ rotates monotonically;
5) Along the entire path, $\vec{\boldsymbol{t}}$ points to the left half side of $\vec{\boldsymbol{h}}$;
6) At the first point, $\vec{\boldsymbol{h}}$ should be opposite to the normal vector of the terrain surface to break into the terrain;
7) At the last point, $\vec{\boldsymbol{h}}$ should be upward so that the collected soil does not spill out;
8) Along the entire path, $\vec{\boldsymbol{h}}$ must either stay the same or rotate clockwise, otherwise the bucket back will press the soil;

Rule 1 to *Rule* 4 serve to constrain the bucket path to be under the terrain surface and in a roughly concave shape. *Rule* 5 serves to regulate instantaneous motions beneath the terrain to cause soil failure. *Rule* 6 to *Rule* 8 serve to constrain the bucket to rotate gradually to roll the soil into the bucket. Note that *Rules* are sufficiently relaxed in the sense that they do not rigidly restrict the motion pattern as previous work does, thereby allowing more diverse trajectories and leaving a larger room for optimization.

In summary, *Rules* mainly stipulate how the bucket move to break and collect the soil by imposing constraints on bucket poses and motions along the path. However, they do not specify the amount of the soil to be excavated. Therefore, we introduce the swept volume constraint to estimate the soil to be excavated with the volume of the soil above the bucket path. Apart from these excavation specific constraints, common robotic constraints such as joint velocity and acceleration limits are imposed in this problem. In the following text, we first explain constraints on the pose and motion of the bucket according to *Rules*. Then we describe the swept volume constraint.

1) Constraints on Bucket Pose and Motion: A bucket pose is obtained through the forward kinematics (FK) mapping: Pose $= (\boldsymbol{p}, \vec{\boldsymbol{h}}) = \text{FK}(\boldsymbol{q})$, where $\boldsymbol{p}$ denotes the position of the bucket tip. $\vec{\boldsymbol{t}}$ between two poses is then obtained by subtraction between tip positions.

All constraints on directions in this work are formulated in terms of the halfspace into which a hyperplane passing through the origin divides the whole space:

$$\text{halfspace} = \{\boldsymbol{x} \mid \boldsymbol{x} \cdot \vec{\boldsymbol{n}} < 0\} \tag{8}$$

where $\vec{\boldsymbol{n}}$ denotes the normal vector of the hyperplane. The region that a direction is confined to can be a half-space or a intersection of several half-spaces. Take the case in Fig. 3(b) as an example, the region is formulated as

$$\text{for } i = 1 \text{ and } 2, \quad \vec{\boldsymbol{v}} \cdot \vec{\boldsymbol{n}}_i < 0 \tag{9}$$

Rule 2, 3, 6 and 7 can be formulated in a similar way to Eq. 9 with $\vec{\boldsymbol{n}}$ determined by the terrain surface.

As to path constraints prescribing bucket motions and their relationship, we add constraints to m points uniformly sampled on the path. Constraints involving monotonous direction changes can be formulated as:

$$\text{for } 0 < i < m-1, \quad \vec{\boldsymbol{v}}_{i+1} \cdot \vec{\boldsymbol{n}}_{\vec{\boldsymbol{v}}_i} < 0 \tag{10}$$

where $\vec{\boldsymbol{n}}_{\vec{\boldsymbol{v}}_i}$ denotes the normal vector corresponding to $\vec{\boldsymbol{v}}_i$. $\vec{\boldsymbol{v}}$ can be replaced by $\vec{\boldsymbol{t}}$ for *Rule* 4 and $\vec{\boldsymbol{h}}$ for *Rule* 8. Similarly, *Rule* 5 concerning the relationship between translation direction and heading direction is formulated as:

$$\text{for } 0 < i < m-1, \quad \vec{\boldsymbol{t}}_i \cdot \vec{\boldsymbol{n}}_{\vec{\boldsymbol{h}}_i} < 0 \tag{11}$$

2) Swept Volume Constraint: Although the amount of the soil collected needs to be accurately computed through the computationally intensive simulation of the bucket-soil interaction, we argue that estimating it with the swept volume is admissible provided heavy machinery operations usually do not require high accuracy. Swept volume is defined as the soil volume above the bucket path and calculated through integrating swept areas on the excavation x-z plane along the y axis, as illustrated in Fig. 3(c). The swept area A_{swept} on the x-z plane is numerically computed with the trapezoidal rule. The swept volume V_{swept} is then computed by sampling l x-z slices from one end point on the bottom plate edge to the other:

$$\text{A}_{\text{swept}} = \sum_{i=0}^{m-1} \frac{(h_i + h_{i+1})\Delta x_i}{2}, \quad \text{V}_{\text{swept}} = \sum_{j=0}^{l-1} \text{A}^j_{\text{swept}} \Delta y \tag{12}$$

where we sample m points on the bucket path; h_i denotes the difference of heights of terrain and the bucket tip at point i; x_i is the horizontal distance between point i and point $i+1$; $\text{A}^j_{\text{swept}}$ is swept area on j-th slice; Δy is the slice distance.

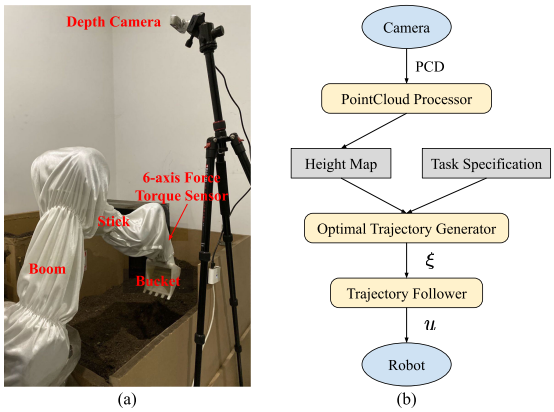
We regulate the trajectory to roughly excavate the desired amount of soil through assigning a constraint on the normalized bucket fill factor K_{bf}, *i.e.* the ratio of swept volume to the real bucket volume:

$$\underline{\text{K}}_{bf} \leq \text{K}_{bf} = \frac{\text{V}_{\text{swept}}}{\text{V}_{\text{bucket}}} \leq \overline{\text{K}}_{bf} \tag{13}$$

where V_{bucket} is computed by multiplying the area of the bucket side plate and the width, as shown in Fig. 1(b). We adopt an inequality constraint since it is not necessary to stipulate a fixed swept volume. We could control the volume of the excavated soil through assigning a desired K_{bf} or its range. Ideally, to achieve the goal of full-bucket filling, *i.e.* to excavate the soil with exact volume of the bucket, we can set $\underline{\text{K}}_{bf} = \overline{\text{K}}_{bf} = 1$. However, since the estimation is not accurate, we assign K_{bf} according to the observed real collected amount of the soil. For example, if we observe the bucket is not full with $\text{K}_{bf} = 1$, we will increase the desired K_{bf}.

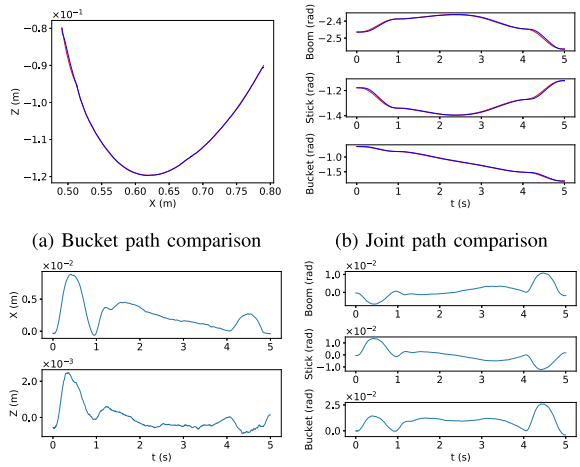
C. Summation

In this section, we summarize the excavation specific objectives and constraints within the general frame of trajectory optimization. In particular, general constraints in Eq. (2) and (3) are in the form of the excavation related constraints, *i.e.* Eq. (9)–(11) and Eq. (13). The general objective function Eq. (1) can be specified according to the task requirement. For tasks requiring high productivity, we can use the minimum travel time objective in Eq. (7). In the case of loose soils where energy consumption might be ignored due to slight resistance forces, we could consider the pure kinematic criteria such as the minimum squared length in Eq. (6). In the case of hard soils where resistance forces are predominant, large joint torque might be required for execution, which could inflict severe damage on joints. In the similar way proposed in [17], [18], we can incorporate the minimum squared torque objective and the torque limitations



(a) (b)

Fig. 4. (a) Experiment platform. (b) Flow chart of a single cycle of excavations. ξ denotes the optimal trajectory. u represents the joint motion command.



(a) Bucket path comparison (b) Joint path comparison

(c) Bucket path error (d) Joint path error

Fig. 5. Path tracking performance. The planned path is depicted with red, and the executed path is depicted with blue.

within the optimization framework. Specifically, given a resistance force $\boldsymbol{f}$ applied at the bucket tip, the torque $\boldsymbol{\tau}$ is computed with the inverse dynamics: $\boldsymbol{\tau} = \mathrm{ID}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}, \boldsymbol{f})$. Consequently, the torque-related objective and constraint are formulated as:

$$\underset{\mathbf{Q},T}{\text{Minimize}} \sum_{i=0}^{n-1} \|\boldsymbol{\tau}_i\|^2 \quad \text{and} \quad \boldsymbol{\tau}_i \leq \bar{\boldsymbol{\tau}},\ i = 0, \ldots, n-1 \tag{14}$$

where $\bar{\boldsymbol{\tau}}$ is upper limits of the torque.

IV. Experiments

We conduct experiments on a real robot platform, where the excavator is substituted by a Universal Robot UR5 with a small bucket, as shown in Fig. 4 a. The last two joints are locked to imitate the 4 DoF structure of typical excavators. A 6-axis force-torque sensor connects the end-effector and the bucket, which is used to measure the mass of excavated soil.

The flow chart of a single cycle of excavations is depicted in Fig. 4 b. At the beginning of each cycle, a depth camera (a Intel RealSense D435 is used in this work) captures a pointcloud of the terrain which is further processed and converted into a height map. Taking the height map and a task specification as inputs, the optimal trajectory generator plans a trajectory to be tracked by the trajectory follower. At each time step, the trajectory follower sends a motion command to the robot to be executed. A task specification can be assigned manually or by a higher level planner, which in general includes the start position of the digging, the desired K_{bf} and the objective.

The optimization problem comprises an objective and multiple constraint functions, whose values are at different scales. We introduce a weight associated to each function to scale them to a uniform range. On this basis, the weights are slightly tuned to adjust relative importance of the functions. In our experiments, we set weights of constraints larger than that of the objective to primarily guarantee a successful excavation. We utilize the advanced optimizer with sequential quadratic programming (SQP) developed in [9], which tackles nonlinearities of the problem by iteratively and locally approximating the original problem with a quadratic objective and a set of linear constraints. We implement the proposed algorithm on a desktop with a 12-Core Genuine Intel i9-8950 CPU and 32 GB memory. A planning takes 0.1 s on average.

The trajectory follower continuously sends target joint positions to be reached with a position-based controller embedded in the robot. With sampling rate of the streaming (500 Hz) higher than the control frequency (125 Hz), the controller can smoothly track the trajectory by considering target joint positions in a certain look-ahead time. Fig. 5 shows path tracking results of a digging in a dry soil. The mean errors of joint positions are 2.7e-3rad, 4e-3rad, 9.7e-3rad for boom, stick and bucket respectively. The mean bucket position errors are 2.5 mm and 0.5 mm for x and z.

A. Swept Volume Estimation Validation

The primary target of an excavation task is to dig enough soil out. Therefore, it is crucial to demonstrate that the amount of the soil actually excavated can be controlled by setting desired K_{bf}. In this experiment, we generate trajectories with different desired K_{bf} to manifest a positive correlation between K_{bf} and the resulting amount of soil actually excavated. We also show such relation is valid in a wide variety of excavation conditions distinguished by three other factors: the soil property, the terrain shape and the bucket velocity.

We conduct experiments on two soils with different humidity labelled as the dry and the moist in the following text. For each type of soil, we compare results of excavations in 4 conditions distinguished by terrain shapes (flat vs. slope) and bucket velocity (slow trajectories with travel time of 10 s vs. fast ones with 5 s). For each condition, we conduct a group of experiments with K_{bf} starting from 1 and increasing 1 each time until the bucket is fully filled. The dry soil filling the entire bucket weighs 1.68 N while the moist soil weighs 1.9 N. We repeat 20 trials for each experiment to learn the stochastic distribution of the mass of excavated soil in terms of K_{bf}.

Images of the excavated soil with different K_{bf} in Dry-Flat-Slow experiments are given in Fig. 6. Fig. 7 compares the normalized weight of the excavated soil in all tested conditions. The normalized weight is computed as the ratio of real weight to

(a) $K_{bf} = 1$ (b) $K_{bf} = 2$ (c) $K_{bf} = 3$ (d) $K_{bf} = 4$

Fig. 6. The excavated soil in Dry-Flat-Slow experiments.

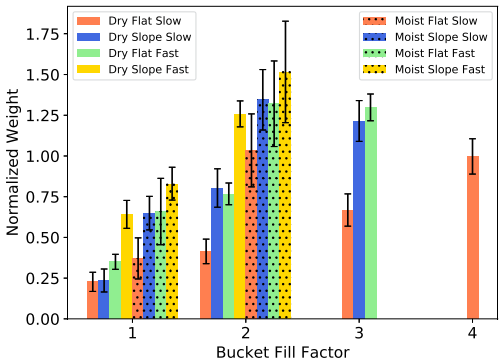
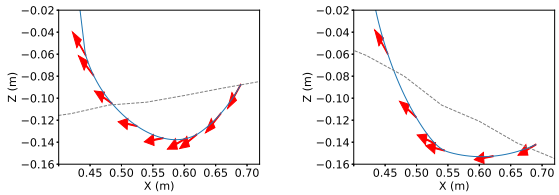
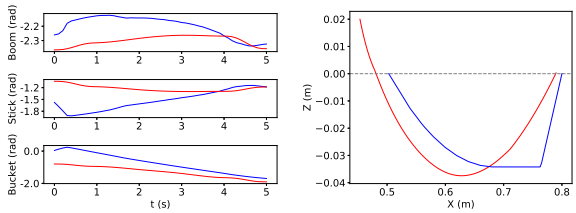


Fig. 7. Statistics on normalized weight of the excavated soil in different conditions. The error bars represent average normalized weights and standard deviations.



(a) Flat terrain (b) Slope terrain

Fig. 8. Typical bucket paths generated for flat and slope terrain. Gray dashed lines represent the terrain surfaces. Blue curves are paths of the bucket tip. Red arrows denote $\vec{h}$s.



(a) Joint Trajectory (b) Bucket Path

Fig. 9. Minimal squared length trajectories starting position $x = 0.8$ m, $z = 0$ m and $K_{bf} = 3$. Red trajectories are generated by our method; Blue trajectories are generated by the TP method.

the full-bucket weight. For all conditions, the normalized weight apparently grows as K_{bf} increases, which suggests that K_{bf} has a significant positive correlation with the weight of excavated soil. For all conditions, the bucket is not fully filled with $K_{bf} = 1$, which is as expected. This is because some soil is pushed away from the bucket during a practical excavation, while the offline swept volume estimation reduces this complex dynamic process into a static situation. Thus, the estimation is prone to over-estimating the amount of excavated soil. This problem can be solved by increasing K_{bf}. However, the exact relation between K_{bf} and the weight varies in different conditions, because other factors also greatly impact the dynamic process of bucket filling. Therefore, the rates of increase of the soil weights differ between different conditions, and so do final K_{bf}s for full-bucket filling.

Here we analyze the results in terms of the other three factors:

1) *Soil Property:* The bucket always digs more moist soil out when other factors are the same. There are two experimental phenomena that indicates the reason: Firstly, compared to the dry soil, it is harder to push the moist soil away due to the increased compactness induced by the moisture; secondly, extra moist soil is rolled into the bucket due to the strong adhesion. As a result, the bucket could be fully filled with the moist soil by assigning $K_{bf} = 2$ in all tested conditions.
2) *Bucket Velocity:* Fast trajectories excavate more soil than the slow ones. It is possibly because the bucket moves so fast that fewer soil escapes from the bucket.
3) *Terrain Shape:* The bucket could load more soil of the slope terrain than the flat terrain with the same K_{bf}. Fig. 8 depicts the tested terrain shapes and the different trajectories planned for them. As the slope situation (Fig. 8(b)) shows, the bucket firstly penetrates into the terrain for a while and then directly lift the above soil, which hardly pushes the front soil.

B. Larger Room for Optimization

This experiment is to demonstrate that the proposed task specification allows larger room for optimization compared to the three-phase (TP) specification. In other words, our constraint-based task specification enables us to find better trajectories than those found by optimal path searching methods using the TP specification. In theory, trajectories with the TP pattern are special cases among all trajectories satisfying the proposed constraints. Therefore, our specification fundamentally has larger feasible trajectory space to search. In practice, this experiment shows that, with the advanced SQP optimizer, our optimization processes indeed converge to a better solution, even though the problem to be solved is more complex. We compare the squared length of joint trajectory obtained by our method and that obtained by exhaustive searching parameters of the TP specification as depicted in Fig. 1(a). As for other criteria, such as minimal torque, it is arguable that the advantage of large search space is still held.

We compare minimal squared lengths of trajectories (computed as in Eq. (6)) starting from 4 different positions with 3 different K_{bf}. Furthermore, on the dry soil, we execute fast trajectories starting at $z = -0.1m$ and compare the weights of excavated soil. Table I lists experiment settings and results.

It is demonstrated that our method is able to find a shorter trajectory for all conditions. The fundamental reason is that the linear translation in TP actions requires simultaneous movements of multiple joints in the excavator, which is determined by the mechanical structure of the excavator. However, large multi-joint movements are usually unnecessary to dig soil out. This can be verified by Fig. 9(a): from 1 s to 4 s, the trajectory of the TP method experiences a large three-joint movement, while with our method the boom and stick only move subtly. Corresponding bucket paths are depicted in Fig. 9(b).

TABLE I
COMPARISON OF MINIMAL SQUARED LENGTHS (rad^2) AND WEIGHTS (N) OF THE EXCAVATED SOIL OF JOINT TRAJECTORY USING THE TP METHOD AND OUR METHOD. RESULTS OF OUR METHOD ARE HIGHLIGHTED IN BOLD TYPE

K_{bf}	pos: $x = 0.6\,\mathrm{m},\ z = 0\,\mathrm{m}$	pos: $x = 0.8\,\mathrm{m},\ z = 0\,\mathrm{m}$	pos: $x = 0.6\,\mathrm{m},\ z = -0.1\,\mathrm{m}$		pos: $x = 0.8\,\mathrm{m},\ z = -0.1\,\mathrm{m}$	
	Min. Sq. Length	Mean Soil Weight	Min. Sq. Length	Mean Soil Weight	Min. Sq. Length	Mean Soil Weight
1	0.0595, **0.0312**	0.1287, **0.0453**	0.0615, **0.0332**	0.36, **0.59**	0.1271, **0.0458**	0.34, **0.61**
2	0.0764, **0.0329**	0.1101, **0.0433**	0.0793, **0.0315**	0.58, **1.06**	0.1100, **0.0378**	0.87, **1.18**
3	0.0835, **0.0339**	0.1388, **0.0356**	0.0909, **0.0339**	1.13, **1.44**	0.1288, **0.0365**	1.22, **1.85**

TABLE II
COMPARISON OF TIME OPTIMAL TRAJECTORIES GENERATED WITH THE ONE-STAGE AND TWO-STAGE METHOD. MIN IS THE MINIMAL TIME OUT OF ALL GENERATED TRAJECTORIES; MEAN IS THE AVERAGE TIME; STD. DEV IS THE STANDARD DEVIATION. RESULTS OF OUR METHOD ARE HIGHLIGHTED IN BOLD TYPE

K_{bf}	pos: $x = 0.6\,\mathrm{m},\ z = 0\,\mathrm{m}$			pos: $x = 0.8\,\mathrm{m},\ z = 0\,\mathrm{m}$			pos: $x = 0.6\,\mathrm{m},\ z = -0.1\,\mathrm{m}$			pos: $x = 0.8\,\mathrm{m},\ z = -0.1\,\mathrm{m}$		
	Min (s)	Mean (s)	Std. Dev (s)	Min (s)	Mean (s)	Std. Dev (s)	Min (s)	Mean (s)	Std. Dev (s)	Min (s)	Mean (s)	Std. Dev (s)
1	6.19, **4.50**	6.83, **5.78**	0.48, **0.55**	6.07, **5.04**	8.80, **6.34**	2.17, **0.75**	5.68, **5.11**	7.46, **6.25**	1.04, **0.52**	5.89, **4.69**	8.25, **6.19**	1.39, **0.91**
2	6.08, **5.35**	7.53, **6.44**	1.51, **0.77**	6.22, **5.84**	8.71, **6.79**	1.60, **0.54**	5.97, **4.89**	7.64, **6.27**	1.19, **0.91**	6.00, **5.58**	8.22, **6.81**	1.49, **0.67**
3	6.05, **5.58**	7.41, **6.91**	0.94, **0.52**	6.91, **5.31**	8.93, **6.83**	1.53, **0.76**	6.43, **5.18**	7.82, **6.72**	1.10, **0.69**	5.92, **4.51**	8.47, **6.64**	1.53, **0.83**

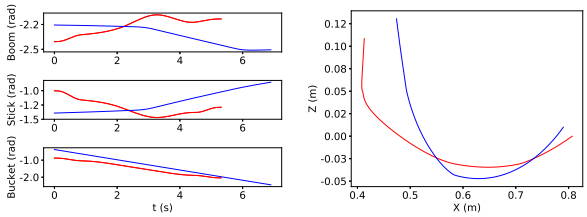
Our method generally excavates more soil than the TP method with the same desired K_{bf}. According to experiment phenomena, it is possibly because that, the distinct phase separation in the TP method physically results in discontinuity of the soil accumulation in front of the bucket (by dragging) and the soil collection into the bucket (by curling), and some accumulated soil is lost in the next curling phase. On the contrary, our method simultaneously accumulates and collects the soil which might reduce the soil loss.

C. Time Optimal Trajectory Generation

This experiment compares minimal travel time of our one-stage method and the two-stage method. For the one-stage method, we directly use the minimal travel time as the objective. As to the two-stage method, we first compute a path satisfying task constraints without objective function. Then we scale time intervals for this path using a time optimal trajectory generation technique proposed in [13]. As in IV-B, we conduct 12 groups of experiments with 4 different starting positions and 3 different K_{bf}. For all conditions, we try multiple initial trajectories with all initial time intervals of 2s to search the minimal time trajectory among them.

Results are shown in Table II. For all conditions, the minimal time of the one-stage method is shorter than that of the two-stage method by 16%. The improvement of the one-stage method in average time is more significant, which is 18% shorter than that of the two-stage method. Besides, the standard deviation of the one-stage timing cost is smaller than that of the two-stage method by 42%. However, we admit that the introduction of time variables makes the optimization problem more difficult due to more variables and nonlinearity. Thus, the rate of successful optimization of the one-stage method is less than that of the two-stage method by 30%. This gap could be hopefully narrowed by using a more powerful optimizer.

Fig. 10 depicts trajectories generated by the two methods and Fig. 11 reports joint lengths of trajectories with the minimum travel time of the 12 groups of experiments. The results show that the one-stage trajectories spend less time due to shorter lengths of bucket joint trajectories in all cases. Although lengths of one-stage boom and stick trajectories might longer than two-stage trajectories, lengths of bucket trajectories with one higher order of magnitude dominate the travel time.



(a) Joint Trajectory (b) Bucket Path

Fig. 10. Comparison of time optimal trajectories with starting position of $x = 0.8$ m, $z = 0$ m and $K_{bf} = 3$. Red trajectories are generated by our one-stage method; Blue trajectories are generated by the two-stage method.

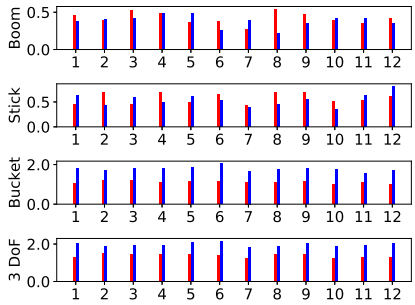


Fig. 11. Comparison of joint lengths of time optimal trajectories. Red bars are lengths of the one-stage trajectories; Blue bars are lengths of the two-stage trajectories. The unit of length is radian.

D. Productivity Test

This experiment is to demonstrate that our method can generate trajectories with high productivity and adaptability to terrains in different shapes. The task requires the excavator to efficiently remove the soil in a region until the surface of this local terrain is lower than a target height. We consider a line region that $x \in [0.4\,\mathrm{m}, 0.8\,\mathrm{m}]$ and $y = 0$ m. The initial height of the local terrain is around -0.1 m, and the target height is -0.12 m. As defined in [1], productivity refers to the volume of excavated soil per operation time. To achieve high productivity, we generate time-optimal trajectories under a swept volume constraint requiring that sufficient soil would be dug out for every excavation. As the excavation progresses, the shape of the terrain largely changes. To guarantee full-bucket excavation no matter what the terrain shape is, we conservatively assign

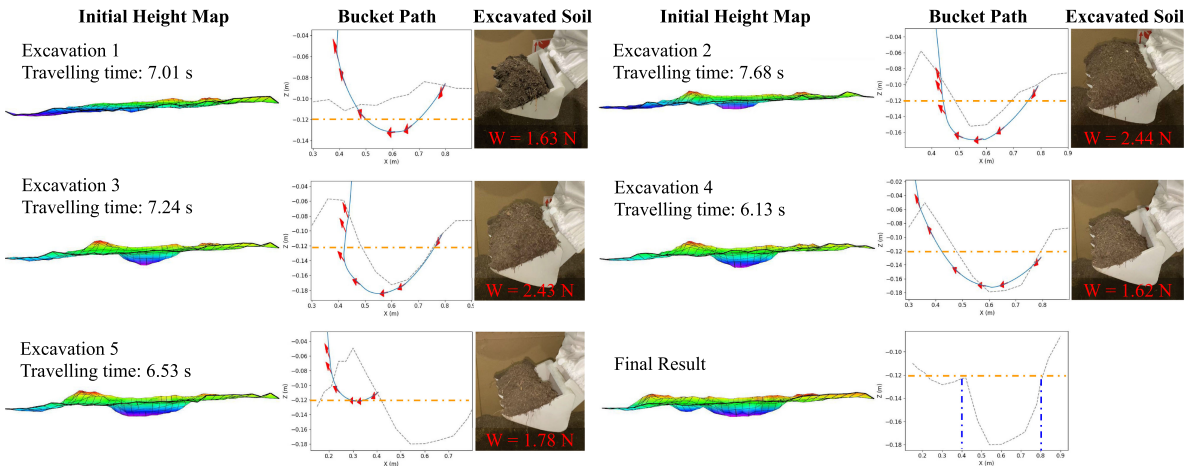


Fig. 12. Five continuous excavations in the productivity test. Blue dashed lines show the region that we consider. Orange dashed lines are the target height. W on each excavated soil image refers to the soil weight. Best viewed on screen.

$K_{bf} = 3$ according to the experiment results of the dry soil in Section IV-A. Fig. 12 illustrates the five continuous excavations to fulfill the task.

V. Conclusion

We develop an optimization-based framework of excavation trajectory generation, which allows optimizing various criteria and guarantees full-bucket filling for different terrains shapes. The trajectory is specified with a set of flexible constraints which regulate bucket instantaneous motions. Thus, the optimization is performed over a broader trajectory space compared to the previous work and can converge to better solutions. Besides, this framework facilitates generating time-optimal trajectories in one stage with time variables. The outperformance of our framework is demonstrated in experiments on a real robot platform.

There remains a lot of future work related to the optimal excavation trajectory generation. Our framework currently estimates the amount of excavated soil with static swept volume, but it can extend to other accurate bucket-soil interaction models. We will conduct experimental research on dynamically optimal trajectory generation within this framework in terms of torque and energy.

References

[1] F. Ng, J. A. Harding, and J. Glass, "An eco-approach to optimise efficiency and productivity of a hydraulic excavator," *J. Cleaner Prod.*, vol. 112, pp. 3966–3976, 2016.
[2] S. Sing, "Synthesis of tactical plans for robotic excavation," Ph.D. dissertation, Carnegie Mellon University, 1995.
[3] Y. Yang, L. Zhang, X. Cheng, J. Pan, and R. Yang, "Compact reachability map for excavator motion planning," in *Proc. Int. Conf. Intell. Robots Syst.*, 2019, pp. 2308–2313.
[4] F. E. Sotiropoulos and H. H. Asada, "A model-free extremum-seeking approach to autonomous excavator control based on output power maximization," *Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1005–1012, 2019.
[5] R. J. Sandzimier and H. H. Asada, "A data-driven approach to prediction and optimal bucket-filling control for autonomous excavators," *RA-L*, vol. 5, no. 2, pp. 2682–2689, 2020.
[6] E. McKyes, *Soil Cutting and Tillage*. New York, NY, USA: Elsevier, 1985.
[7] B. Park, "Development of a virtual reality excavator simulator: A mathematical model of excavator digging and a calculation methodology," Ph.D. dissertation, Virginia Tech, 2002.
[8] M. Zucker *et al.*, "Chomp: Covariant hamiltonian optimization for motion planning," *Int. J. Robot. Res.*, vol. 32, no. 9-10, pp. 1164–1193, 2013.
[9] J. Schulman *et al.*, "Motion planning with sequential convex optimization and convex collision checking," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1251–1270, 2014.
[10] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, "STOMP: Stochastic trajectory optimization for motion planning," in *Proc. Int. Conf. Robot. Automat.*, 2011, pp. 4569–4574.
[11] C. Rösmann, A. Makarow, and T. Bertram, "Time-optimal control with direct collocation and variable discretization," 2020, *arXiv:2005.12136*.
[12] J. E. Bobrow, "Optimal robot plant planning using the minimum-time criterion," *IEEE J. Robot. Autom.*, vol. 4, no. 4, pp. 443–450, Aug. 1988.
[13] T. Kunz and M. Stilman, "Time-optimal trajectory generation for path following with bounded acceleration and velocity," *Robot.: Sci. Syst.*, pp. 1–8, 2012.
[14] Q.-C. Pham, "A general, fast, and robust implementation of the time-optimal path parameterization algorithm," *Trans. Robot.*, vol. 30, no. 6, pp. 1533–1540, 2014.
[15] K. Hauser, "Fast interpolation and time-optimization with contact," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1231–1250, 2014.
[16] S. Yoo, C. Park, S. You, and B. Lim, "A dynamics-based optimal trajectory generation for controlling an automated excavator," *Proc. Inst. Mech. Eng.*, Part C: J. Mech. Eng. Sci., vol. 224, no. 10, pp. 2109–2119, 2010.
[17] Y. B. Kim, J. Ha, H. Kang, P. Y. Kim, J. Park, and F. Park, "Dynamically optimal trajectories for earthmoving excavators," *Automat. Construction*, vol. 35, pp. 568–578, 2013.
[18] Z. Zou, J. Chen, and X. Pang, "Task space-based dynamic trajectory planning for digging process of a hydraulic excavator with the integration of soil-bucket interaction," *Proc. Inst. Mech. Eng., Part K: J. Multi-body Dyn.*, vol. 233, no. 3, pp. 598–616, 2019.
[19] S. Blouin, A. Hemami, and M. Lipsett, "Review of resistive force models for earthmoving processes," *J. Aerosp. Eng.*, vol. 14, no. 3, pp. 102–111, 2001.
[20] S. Singh, "Learning to predict resistive forces during robotic excavation," in *Proc. Int. Conf. Robot. Automat.*, 1995, vol. 2, pp. 2102–2107.
[21] G. Hess *et al.*, "Simulation of the dynamic interaction between bulk material and heavy equipment: Calibration and validation," in *Proc. Int. Conf. Bulk Mater. Storage, Handling Transp.*, 2016, Art. no. 427.
[22] C. Coetzee, A. Basson, and P. Vermeer, "Discrete and continuum modelling of excavator bucket filling," *J. Terramechanics*, vol. 44, no. 2, pp. 177–186, 2007.
[23] D. Jud, G. Hottiger, P. Leemann, and M. Hutter, "Planning and control for autonomous excavation," *Robot. Autom. Lett.*, vol. 2, no. 4, pp. 2151–2158, Oct. 2017.
[24] B. Son, C. Kim, C. Kim, and D. Lee, "Expert-emulating excavation trajectory planning for autonomous robotic industrial excavator," in *Int. Conf. Intell. Robots System*, 2020.
[25] J. Z. Kolter and A. Y. Ng, "Task-space trajectories via cubic spline optimization," in *Proc. Int. Conf. Robot. Automat.*, 2009, pp. 1675–1682.