Efficient Reliability-Based Path Planning of Off-Road Autonomous Ground Vehicles Through the Coupling of Surrogate Modeling and RRT*

Jianhua Yin, *Member, IEEE*, Zhen Hu, Zissimos P. Mourelatos, David Gorsich, Amandeep Singh, and Seth Tau

***Abstract*—Reliability-based global path planning incorporates reliability constraints into path planning to ensure that off-road autonomous ground vehicles can operate reliably in uncertain off-road environments. Current two-stage reliability-based path planning methods involve separate stages for surrogate modeling of mobility prediction and global path planning, resulting in a large number of unnecessary mobility simulations that makes the approaches computationally expensive. To tackle this challenge, this work proposes a novel efficient reliability-based global path planning approach, named ER-RRT*, which couples adaptive surrogate modeling with the rapidly-exploring random tree star (RRT*) algorithm. Firstly, a surrogate model for vehicle mobility prediction is used to guide the exploration of random trees subject to a mobility reliability constraint. Subsequently, the exploration trees and reliability assessment are employed to inform mobility simulations for the surrogate model refinement. These steps are implemented iteratively and thereby drastically reducing the required mobility simulations for path planning through the integration of adaptive surrogate modeling with global path planning. With a focus on the uncertainty in the slope map and soil properties of deformable terrain, we demonstrate ER-RRT* using a case study and compare it with the current two-stage approach. The results show that ER-RRT* is much more efficient than the current method in both computational time and the required number of mobility simulations for surrogate model construction. In addition, the path identified by ER-RRT* exhibits a comparable cost in distance to its counterpart obtained using the two-stage method.**

***Index Terms*—Reliability, path planning, uncertainty, off-road, autonomous ground vehicle**

I. Introduction

Path planning is critical for autonomous ground vehicles and robotics [1]–[4]. It is used to find the shortest path from a starting point to the goal point meeting specific safety and energy requirements. Two types of path planning methods exist, namely online (local) path planning [5]–[7] and offline (global) path planning [8]–[10]. Online path planning updates the next position based on the perception of the surrounding environment using sensors (e.g. Lidar, Radar, camera, etc.) in a real-time manner [11]. Offline path planning determines the optimal path in the target area before a mission begins, based on prior environmental information.

One important application of path planning is in off-road autonomous ground vehicles (AGVs). Deploying AGVs in harsh and/or boring off-road working environments, such as battlefields [12], outer space [13]–[15], or the agricultural industry [16], could benefit from reducing fatalities and labor cost. Although current online and offline path planning methods have shown promising performance for robots and autonomous vehicles working on well-structured roads using proper path planning and control strategies [17]–[20], challenges still exist in their application to off-road AGVs. First, sensor-perception-based online path planning could fail to detect the obstacles in the working environment of off-road AGVs. For example, sensors may not able to identify terrain conditions, such as sand, mud pond, and wet soil [21]. Even if a camera can detect the terrain condition, it cannot tell whether the AGVs will lose mobility in the deformable terrain. Second, the working environment of off-road AGVs is highly uncertain, which leads to uncertainty in vehicle mobility [22]. For instance, the soil properties of a certain soil type in the off-road environment are uncertain in nature. Even for the same type of soil, soil properties such as the cohesive strength, and friction coefficient can vary across locations and are unknown. Since it is impossible to measure the soil properties at every location, these parameters have to be modeled as uncertain variables to describe the natural variability which leads to uncertainty of the off-road AGV mobility. As a result, the vehicle has a certain chance of losing mobility in a deformable terrain. The area where the vehicle loses mobility is defined as an obstacle that is probabilistic instead of deterministic due to uncertainty. Without taking the uncertainty of vehicle mobility into account in path planning, the resulting mission could have a high chance of failure.

A. Path planning based on modeling and simulation

Physics-based modeling and simulation (M&S) can potentially address the aforementioned challenges of path planning of AGVs in complex and uncertain off-road working

This research was supported by the Automotive Research Center (ARC) in accordance with Cooperative Agreement W56HZV-19-2-0001 U.S. Army DEVCOM Ground Vehicle Systems Center (GVSC), Warren, MI, and in part by the U.S. National Science Foundation under Grant CMMI-2301012. The support is gratefully acknowledged (Corresponding author: Zhen Hu.)

Jianhua Yin and Zhen Hu are with the Department of Industrial and Manufacturing Systems Engineering, University of Michigan-Dearborn, Dearborn, MI 48128, USA (E-mail: yin1991me@gmail.com, zhennhu@umich.edu).

Zissimos P. Mourelatos is with the Mechanical Engineering Department, Oakland University, Rochester, MI 48309, USA (E-mail: mourelat@oakland.edu).

David Gorsich, Amandeep Singh and Seth Tau are with the U.S. Army Combat Capabilities Development Command, Ground Vehicle Systems Center, Warren, MI 48397, USA (E-mail: {david.j.gorsich.civ, amandeep.singh2.civ, seth.a.tau.civ}@army.mil).

environments for several reasons [23]–[25]. First, physics-based high-fidelity M&S can predict various vehicle mobility quantities for given terrain conditions before operating the actual vehicle. In this way, whether the vehicle will lose mobility (e.g., get stuck) on a mission can be predicted using a physics-based method. Second, physics-based M&S allows for the consideration of various uncertainty sources in path planning to ensure the reliability of a planned mission. Since offline global path planning is less computationally demanding compared to online local path planning, it can be used with physics-based mobility prediction. To quantify the influence of various uncertainty in the working environment of off-road AGVs, the computationally expensive high-fidelity M&S must be executed numerous times (e.g., millions of times if the direct Monte Carlo simulation (MCS) method is adopted). The required computational effort could be prohibitive even for M&S-based off-line global path planning. How to accurately and efficiently predict vehicle mobility in the off-road environment and in the presence of uncertainty remains a long-standing research problem.

Many studies have been carried out focusing on improving the fidelity of the physics-based mobility model starting from the 1960s. The current mobility models for M&S can be roughly classified into two categories, namely semi-empirical models and computational simulation models. Semi-empirical models [26]–[28], for example the NATO Reference Mobility Model (NRMM) [26], are built from experience and experiments, which are very efficient to solve but may have large errors in mobility prediction [29]. Computational simulation models [30]–[32] have arisen and become popular in recent years with the advancement of computational mechanics, multi-physics analysis, and the improvement in computing power. These simulation models are usually partial differential equations solved by finite element method (FEM) or other numerical methods, which have high fidelity. For instance, vehicle mobility is predicted considering the coupled vehicle-tire-terrain interactions solved by FEM and discrete element method in Refs. [33], [34]. The U.S. Army and NATO are making an effort to develop the next-generation NRMM model, named NG-NRMM, by leveraging a high-performance computing platform, numerical algorithms for high-fidelity vehicle-terrain interactions, uncertainty quantification methods, and integration with GIS data and mapping software [30]. While the mobility simulation models are getting more and more sophisticated and accurate, the required computational power is also getting higher and higher.

B. Reliability-based path planning

Motivated by finding a reliable path under uncertain off-road environments and overcoming the associated computational challenges caused by high-fidelity mobility simulations, several approaches have been developed using surrogate models, which replace the original mobility simulation model with a cheap yet accurate machine learning model. For instance, Gonzalez et al. [25] combined Kriging surrogate model with MCS to quantify the uncertainty of vehicle mobility caused by uncertainty in terrain and soil parameters. A stochastic mobility map was constructed using dynamic Kriging to predict the reliability of an AGV in an area of interest [35]. Jiang et al. [21] proposed a two-stage reliability-based global path planning approach which considered uncertainty in a more realistic manner and took the space-dependency of the environment uncertainty into account [36]–[38]. Based on the two-stage framework, several reliability-based path planning approaches have been developed with more advanced reliability considerations [23], [24].

C. Motivation

Although the current surrogate modeling methods show promising potential in tackling the computational challenge in reliability-based path planning using physics-based M&S, they require a lot of computational effort to train an accurate surrogate model. Sometimes, the required computational cost is even prohibitive if reliability constraints are considered. This is mainly attributed to the following reasons. First, current methods find an accurate reliability map in the whole target area before path planning. This process explores all the locations on the map, in which some exploration might be unnecessary since these areas may not be useful at all for path planning. This will lead to huge computational cost waste, especially when the target map is large. Second, the mobility at the next step not only depends on the locations that the vehicle passed but also on the future locations for dynamic problems. There are too many combinations of the training paths which makes it computationally too expensive to train an accurate surrogate model that can fully represent the original physics-based high-fidelity M&S. There is an urgent need to develop a more efficient surrogate modeling method for reliability-based global path planning.

D. Contributions

This paper meets the above need by creating a novel reliability-based global path planning approach through the coupling of adaptive surrogate modeling and rapidly-exploring random tree star (RRT*) method. The proposed approach finds the shortest path while satisfying mobility reliability requirements by sequentially implementing surrogate model refinement using active learning, reliability map updating, path planning, and path validation. Different from the conventional two-stage approaches, the proposed method does not need to explore the whole map area which can reduce the computational cost of generating a reliability map. The mobility surrogate model and reliability map are updated recursively by validating the sub-area around the path identified in each iteration. The RRT* algorithm is coupled with the refinement of the mobility surrogate and reliability map (obstacle map) updating to more effectively utilize the computational resources. In addition, the existing random trees from previous RRT* are trimmed based on the updated obstacles in the map, so that the trees from the previous iteration can be reused and served as the initial random trees of current path planning instead of starting a new exploration in every iteration using RRT*. As shown in the numerical example provided in Section V, such a coupling between adaptive surrogate modeling and RRT* leads to a significant reduction in the computational cost for global path planning.

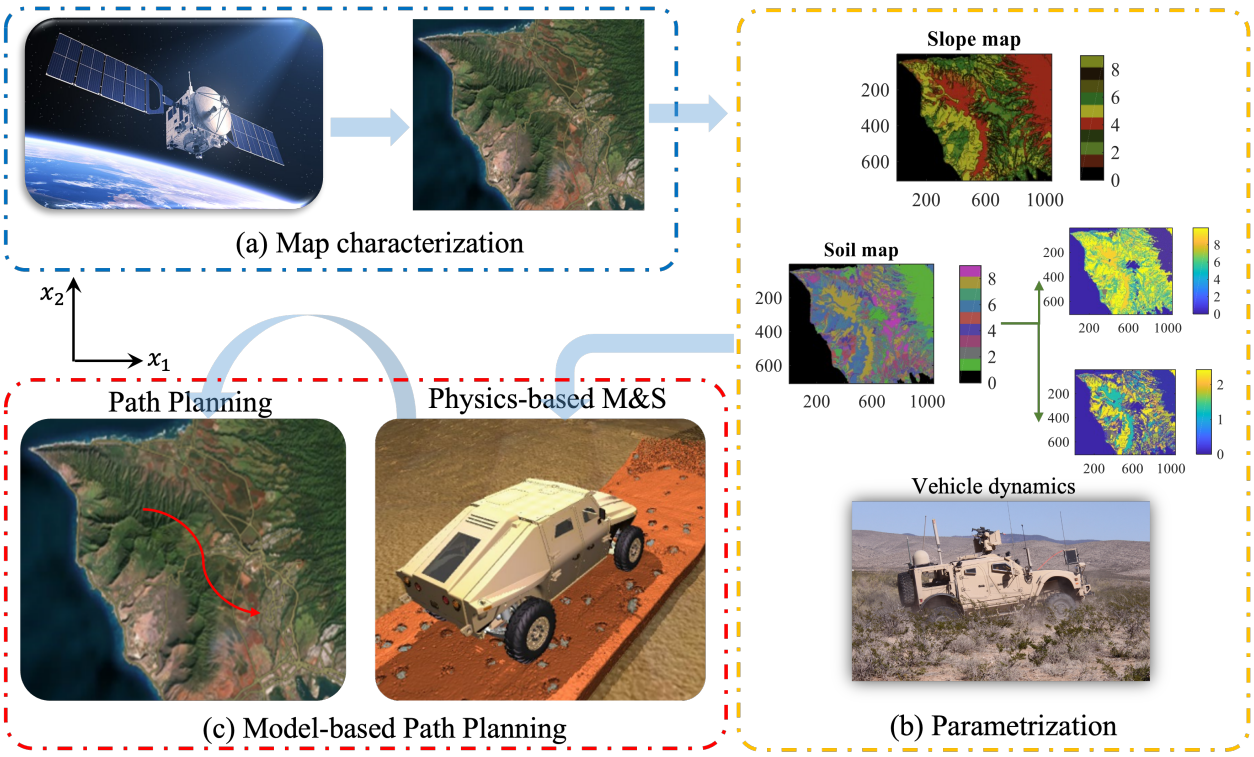

Fig. 1. An overview of the model-based approach to global path planning for off-road AGVs. (a) A geological map is characterized by satellites or geological survey. (b) Slope map, soil map, and vehicle dynamics are parameterized based on geological information including the terrain slope, soil properties, and vehicle-related parameters. (c) Path planning with physics-based modeling & simulation (M&S) is used to obtain a reliable path. Physics-based M&S is employed to describe the interactions between vehicles and terrain for the prediction of vehicle mobility at different locations.

The contributions of this work are multi-fold and are summarized below.

- An efficient reliability-based global path planning is proposed via the coupling of adaptive surrogate modeling and global path planning.
- Accounting for reliability constraint in path planning in the proposed framework. The sequential updating of the reliability map in subareas of interest reduces the exploration effort and area without sacrificing accuracy.
- The efficiency of the overall algorithm is improved by reusing the trees of RRT* from previous iterations instead of starting a new exploration. This is especially attractive when the map is large.

The remainder of the paper is organized as follows. The work related to this study is reviewed in Section II. The general elements of reliability-based global path planning are provided in Sectiont III. In Section IV, the proposed coupled reliability-based path planning approach is presented. A case study is used to demonstrate the proposed approach in Section V. Last, concluding remarks are given in Section VI.

II. Preliminary

A. Model-based global path planning

Model-based global path planning finds the shortest path given the map information and a physical model (vehicle dynamics model). Fig. 1 presents an overview of model-based global path planning. Several general concepts and their symbols are defined here and are used throughout the paper to explain the proposed method.

The target map is called the configuration space and is denoted by Ω. Arbitrary coordinates in Ω are represented as $\mathbf{x} = (x_1, x_2)$, which are respectively the latitude and longitude positions in the configuration space. The space with obstacles that an AGV cannot pass in Ω is called an "obstacle space" denoted by Ω_{os}. The space that an AGV can maintain its mobility in Ω is called a "free space" and is denoted by Ω_{free}.

Based on the above definitions, a generalized optimization model for model-based path planning is given by

$$\begin{aligned} &\Gamma^* = \arg\min_{\Gamma\in\Omega}\{L(\Gamma)\} \\ &\text{s.t.} \\ &\mathbf{x}_0 \in \Gamma = \mathbf{x}_{\text{ini}};\ \mathbf{x}_{\text{end}} \in \Gamma = \mathbf{x}_{\text{goal}} \\ &\mathbf{x}_i \in \Omega_{\text{free}},\ \forall \mathbf{x}_i \in \Gamma \\ &\mathbf{x}_{\text{goal}} = \{\mathbf{x} \in \Omega_{\text{free}} \,|\, \|\mathbf{x} - \mathbf{x}_{\text{goal}}\| \le \varepsilon\}, \end{aligned} \tag{1}$$

where Γ^* is the optimal path, $L(\Gamma)$ is the cost of a path which is in the configuration space ($\Gamma \in \Omega$), the starting point $\mathbf{x}_0$ of a path Γ is the initial point $\mathbf{x}_{\text{ini}}$ of the mission, and the end point $\mathbf{x}_{\text{end}}$ of Γ is the goal $\mathbf{x}_{\text{goal}}$ of the mission. Besides, all coordinates $\mathbf{x}_i$ on Γ ($\forall \mathbf{x}_i \in \Gamma$) should be in the free space Ω_{free}. It is assumed that the goal $\mathbf{x}_{\text{goal}}$ is found when the distance between a location $\mathbf{x}$ in Ω_{free} and $\mathbf{x}_{\text{goal}}$ is less than a threshold distance ε.

The free space and obstacle space depend on the mobility of vehicles. If a vehicle loses mobility at specific locations,

the locations belong to obstacle space. On the other hand, the location belongs to free space if the vehicle can maintain mobility at the location. Due to the presence of deformable terrain, the free and obstacle spaces are not explicitly known and need to be predicted using a physics-based mobility simulation model. Next, we discuss how to classify a location as an obstacle or free-space based on physics-based analysis.

B. Physics-based mobility simulation model

The physics-based vehicle mobility prediction model used in this work is a high-fidelity multi-body dynamics simulation model coupled with a deformable terrain mechanics model, called PyChrono [39], [40]. Fig. 2 shows an example of a moving AGV in the simulation of PyChrono.

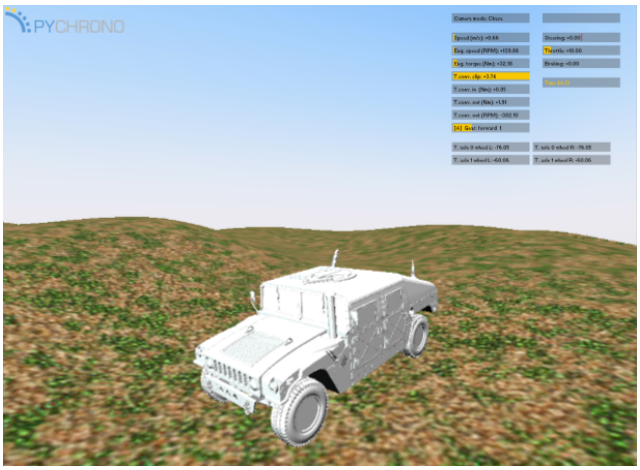

Fig. 2. An example of PyChrono simulation. The simulation includes vehicle dynamics and vehicle-terrain interaction.

The mobility model is mathematically represented by multiple coupled partial differential equations (PDEs). Since solving the PDEs is not a contribution of this work, we present a general form of mobility model instead of providing the PDEs. The generalized mobility model is given by

$$Y(\mathbf{x}) = M(\mathbf{C}, \mathbf{S}(\mathbf{x})), \mathbf{x} \in \Omega, \tag{2}$$

where $Y(\mathbf{x})$ represents the vehicle mobility (maximum attainable speed in this work) at location $\mathbf{x}$ in the map Ω; $\mathbf{C}$ is a vector of vehicle-related parameters; and $\mathbf{S}(\mathbf{x})$ is a vector of terrain-related parameters including terrain slope, soil bulk density, soil friction coefficient, etc. (see Sec. V for details) at location $\mathbf{x}$.

Based on the physics-base mobility simulation, the free space Ω_{free} and obstacle space Ω_{os} given in Eq. (1) are defined as

$$\mathbf{x} \in \begin{cases} \Omega_{\text{free}}, & \text{if } Y(\mathbf{x}) \geq y_e, \\ \Omega_{\text{os}}, & \text{if } Y(\mathbf{x}) < y_e \end{cases}, \forall \mathbf{x} \in \mathbf{\Omega}. \tag{3}$$

where y_e is the threshold to determine whether the vehicle loses mobility. As mentioned above, the mobility considered in this paper is the maximum attainable speed. If the maximum attainable speed is less than y_e, the vehicle will get stuck in the deformable terrain (i.e., loss of mobility).

*C. RRT**

RRT* is a sampling-based path planning algorithm using a space-filling random tree technique. The tree grows starting from the initial point $\mathbf{x}_{\text{ini}}$ until a path is found toward the goal $\mathbf{x}_{\text{goal}}$. A random tree growing is a process of generating a number of random samples in Ω. The random samples are accepted if $\mathbf{x}_{\text{rand}} \in \Omega_{\text{free}}$ and rejected if $\mathbf{x}_{\text{rand}} \in \Omega_{\text{os}}$. This process is also called collision check. The nearest node $\mathbf{x}_{\text{nst}}$ to the accepted random node $\mathbf{x}_{\text{rand}}$ is identified. If there is no obstacle between $\mathbf{x}_{\text{nst}}$ and $\mathbf{x}_{\text{rand}}$, then $\mathbf{x}_{\text{rand}}$ is accessible to $\mathbf{x}_{\text{nst}}$. The random node $\mathbf{x}_{\text{rand}}$ is then inserted into the tree. If $\mathbf{x}_{\text{rand}}$ cannot access $\mathbf{x}_{\text{nst}}$, a steering function is used to generate a new node $\mathbf{x}_{\text{new}}$ which is added to the tree by connecting it to $\mathbf{x}_{\text{nst}}$ after $\mathbf{x}_{\text{new}}$ and $\mathbf{x}_{\text{nst}}$ pass the collision check.

Next, the most important feature of RRT* called rewiring is used to modify the tree. The rewiring process consists of four major steps as shown in Fig. 3. The numbers on the edges used to connect two nodes are the distances between the paired nodes. The four major steps are summarized as follows.

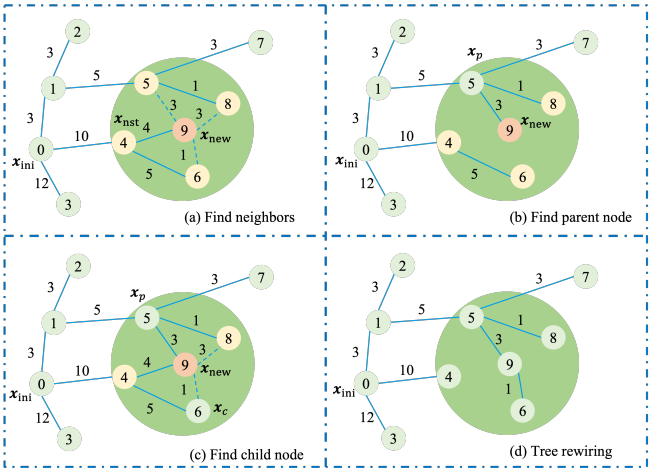

Fig. 3. Illustration of the rewiring process of RRT*.

Step 1: finding the the neighbor nodes of $\mathbf{x}_{\text{new}}$ within a sphere whose radius is defined by

$$r = \gamma \left(\frac{\log n_{\text{iter}}}{n_{\text{iter}}} \right)^{\frac{1}{\dim}}, \tag{4}$$

where γ is a constant coefficient, n_{iter} is the iteration number of random sampling, and dim is the dimension of the configuration space. Four neighbors are identified in the defined area of Eq. (4) as shown in Fig. 3(a).

Step 2: identifying the parent node from the four neighbor nodes. From $\mathbf{x}_{\text{ini}}$ to $\mathbf{x}_{\text{new}}$, if the cost (distance) of a path that passes one of the neighbors is the lowest, the neighbor node is the parent node. According to Fig. 3(b), node 5 is the parent node $\mathbf{x}_p$ with the lowest cost of 11.

Step 3: finding the child node $\mathbf{x}_c$ from the other three neighbor nodes. The path from $\mathbf{x}_{\text{ini}}$ to $\mathbf{x}_c$ passing $\mathbf{x}_p$ and $\mathbf{x}_{\text{new}}$, the lowest cost is 12 when node 6 is the child node $\mathbf{x}_c$. Hence, we have the child node (node 6) as illustrated in Fig. 3(c).

Step 4: rewiring the edges by removing the edges with higher cost ($9 \rightarrow 4$ and $4 \rightarrow 6$) and adding new edges ($5 \rightarrow 9$ and $9 \rightarrow 6$). The rewired tree is shown in Fig. 3(d).

III. Reliability-Based Path Planning

A. Problem formulation

As discussed previously, uncertainty prevails in AGV and its working environment. The uncertainty sources lead to uncertainty in vehicle mobility given Eq. (2). As a result, the obstacle space and free space defined in Eq. (3) are also uncertain. If a deterministic path planning approach is used to find the optimal path, the mission has a certain chance to fail. The reliability-based path planning, therefore, is introduced to tackle this issue. Reliability-based path planning finds the shortest path while satisfying certain reliability requirements, named state mobility reliability (SMR) [21]. The mobility reliability at a certain location $\mathbf{x}$ in the configuration space is defined by

$$\begin{aligned} R(\mathbf{x}) &= \Pr\{Y(\mathbf{x}) = M(\mathbf{C}, \mathbf{S}(\mathbf{x})) > y_e\} \\ &= \iint\limits_{Y(\mathbf{x})>y_e} f_{\mathbf{C},\mathbf{S}}(\mathbf{c}, \mathbf{s}(\mathbf{x}))d\mathbf{c}d\mathbf{s}, \end{aligned} \tag{5}$$

where $f_{\mathbf{C},\mathbf{S}}(\mathbf{c}, \mathbf{s}(\mathbf{x}))$ is the joint probability density function (PDF) in terms of $\mathbf{C}$ and $\mathbf{S}(\mathbf{x})$. Based on the definition of reliability, the reliability-based free space and obstacle space are given by

$$\mathbf{x} \in \begin{cases} \Omega_{\text{free}}, & \text{if } R(\mathbf{x}) \geq R_{\text{lim}}, \\ \Omega_{\text{os}}, & \text{if } R(\mathbf{x}) < R_{\text{lim}} \end{cases}, \tag{6}$$

in which R_{lim} is the reliability requirement. Accordingly, we can rewrite Eq. (1) in a probabilistic manner resulting in the reliability-based path planning model given by

$$\begin{aligned} &\Gamma^* = \arg\min_{\Gamma\in\Omega}\{L(\Gamma)\} \\ &\text{s.t.} \\ &\mathbf{x}_0 \in \Gamma = \mathbf{x}_{\text{ini}};\ \mathbf{x}_{\text{end}} \in \Gamma = \mathbf{x}_{\text{goal}} \\ &\mathbf{x}_i \in \Omega_{\text{free}},\ \forall \mathbf{x}_i \in \Gamma \\ &\Omega_{\text{free}} = \{\mathbf{x} | R(\mathbf{x}) > R_{\text{lim}},\ \forall \mathbf{x} \in \Omega\} \\ &\mathbf{x}_{\text{goal}} = \{\mathbf{x} \in \Omega_{\text{free}} |\ \|\mathbf{x} - \mathbf{x}_{\text{goal}}\| \leq \varepsilon\}. \end{aligned} \tag{7}$$

However, evaluating Eq. (5) is computationally prohibitive, if the high-fidelity mobility simulation model is used directly. This is attributed to the fact that each reliability assessment requires thousands of runs of the mobility prediction model for any given location in the configuration space.

B. Surrogate Modeling

To address the computational challenge in reliability-based path planning, a computationally cheap yet accurate surrogate model is usually constructed to replace the original high-fidelity mobility simulation model using machine learning (ML) methods as

$$\hat{Y}(\mathbf{x}) = G_M(\mathbf{C}, \mathbf{S}(\mathbf{x})), \tag{8}$$

where $\hat{Y}(\mathbf{x})$ denotes the prediction of a ML method, and $G_M(\cdot)$ denotes the ML-based surrogate model.

Since the ML model is cheap to evaluate, the reliability given in Eq. (5) can be approximated using Monte Carlo Simulation (MCS) as follows [41]

$$R(\mathbf{x}) \approx \frac{\sum_{i=1}^{N_{\text{mcs}}} \mathbb{1}(\hat{Y}(\mathbf{x}))}{N_{\text{mcs}}}, \tag{9}$$

where N_{mcs} is the number of MCS samples and $\mathbb{1}(\hat{Y}(\mathbf{x}))$ is an indicator function given by

$$\mathbb{1}(\hat{Y}(\mathbf{x})) = \begin{cases} 1, & \hat{Y}(\mathbf{x}) \geq y_e \\ 0, & \hat{Y}(\mathbf{x}) < y_e \end{cases}. \tag{10}$$

Then, the free space and obstacle space can be approximated using Eqs. (6) and (9). The accuracy of the surrogate model is critical for reliability-based path planning, and is significantly affected by the data used for the training. How to effectively construct an accurate surrogate model with a limited number of training data from high-fidelity simulations is an active research topic, in which active learning is one of the popular ways to refine the surrogate model and to improve the prediction accuracy of a surrogate model [42], [43].

C. The existing methods and challenges

As shown in Fig. 4, the existing method employs a two-stage strategy for reliability-based path planning. The first stage is to train a very accurate surrogate model based on high-fidelity simulations and model refinement using active learning. After that, in the second stage, the optimal path is identified using RRT* based on reliability analysis using the trained surrogate model by accounting for uncertainty sources in the mobility model. Even though it is much more efficient than the original high-fidelity simulation model, such a two-stage approach could still be computationally expensive for two main reasons:

- The computational cost for training a very accurate surrogate model could be very high, when the map is large or there are a large number of input variables [44].
- The reliability map does not need to be accurate everywhere since it is highly possible that many areas of the map will not be important at all for path planning. Improving the accuracy of the reliability map in those areas could be a waste of computational resources.

Motivated by improving the efficiency of model-based path planning, we propose an efficient approach through the coupling of adaptive surrogate modeling and path planning as illustrated in Fig. 4. The main difference between the proposed method and the existing method is that the adaptive surrogate modeling and path planning are performed in a sequential and mutually informed manner instead of in two stages. Next, we introduce the proposed method in detail.

IV. Coupling of Adaptive Surrogate Modeling with RRT*

Since the proposed method couples adaptive surrogate modeling with RRT* for efficient reliability-based path planning,

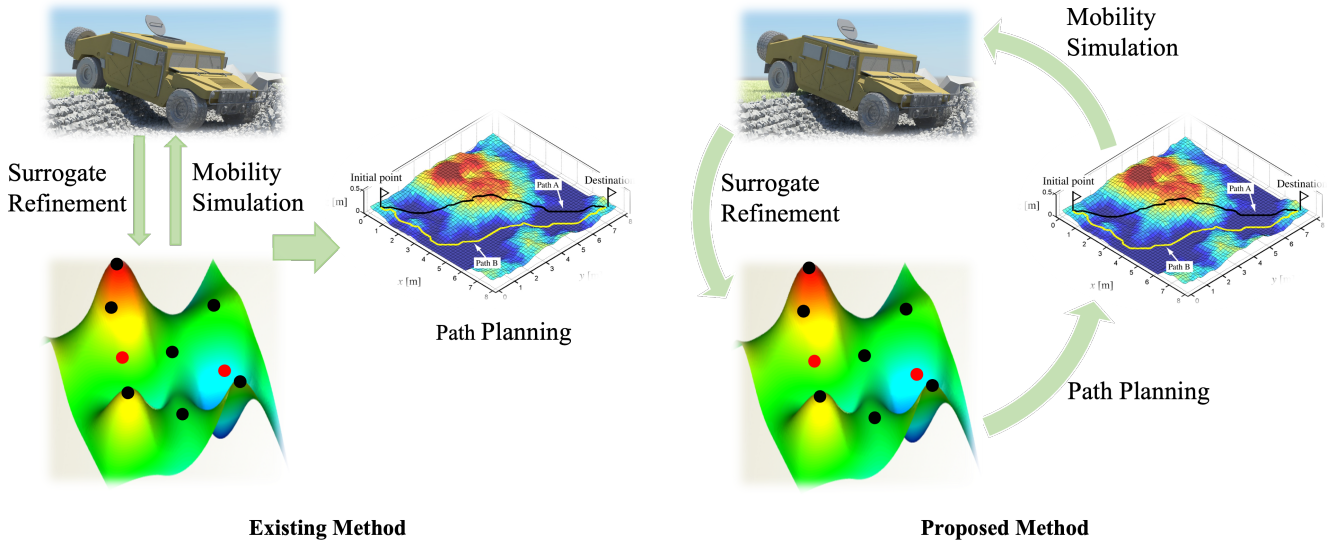

Fig. 4. Comparison of the existing two-stage method and the proposed method. **The existing method** uses a two-stage strategy which first trains an accurate surrogate model based on which an optimal path is determined by RRT*. **Proposed method** couples surrogate refinement and path planning iteratively to find the optimal path.

we named it efficient reliability-based RRT* (ER-RRT*). As mentioned above, different from the conventional two-stage approach, ER-RRT* couples the surrogate model refinement with path planning. The surrogate model refinement is used to improve the accuracy of the reliability map for reliability-based path planning. The path planning algorithm RRT* in turn informs the refinement of the surrogate model. A unique feature of the proposed method is that the refinement area of the reliability map is only in the vicinity of the path identified by RRT* instead of the whole map. This leads to a drastic reduction in the computational effort for surrogate modeling for reliabiltiy-based path planning. Moreover, to reduce the overhead time, we propose a trim tree algorithm to use the tree generated by RRT* from previous iterations instead of starting a new search at each iteration.

Next, we introduce different elements of the ER-RRT* algorithm, which include uncertainty modeling, pseudo path planning based on surrogate modeling, surrogate model refinement informed by path planning, and tree updating.

A. Uncertainty Modeling

As discussed previously, uncertainty prevails in the working environment of AVGs and vehicles themselves. In this work, we only consider the uncertainty from the working environment (slope and soil) of the AGV. The uncertainty is space-dependent and simulated by random fields [36], [45], [46]. The truncated Karhunen-Loeve (K-L) expansion is employed to generate realizations of the random fields and is given by [47]

$$s(\mathbf{x}) = \mu(\mathbf{x}) + \sigma(\mathbf{x}) \sum_{i=1}^{T_n} \sqrt{\lambda_i} \varphi_i(\mathbf{x}) \xi_i, \tag{11}$$

where $s(\mathbf{x})$ is a realization of a random field $S(\mathbf{x})$, $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$ are respectively the spatial-dependent mean and standard deviation functions of the random field, $\mathbf{x}$ is the spatial location, T_n is the number of truncation terms, λ_i and $\varphi_i(\mathbf{x})$ are the eigenvalue and eigenvector from eigen-analysis of a correlation matrix, respectively, and ξ_i is the i^{th} independent standard normal random variable.

The correlation matrix is obtained using the correlations of a set of mesh nodes based on a correlation function which is given by

$$k_{12} = \exp\left\{ - \left\| (\mathbf{x}_1 - \mathbf{x}_2) \oslash \boldsymbol{\theta}_{\mathbf{x}} \right\|_2 \right\}, \tag{12}$$

where k_{12} is the correlation of two points, $\mathbf{x}_1$ and $\mathbf{x}_2$ are two arbitrary points of the mesh nodes, and $\boldsymbol{\theta}_{\mathbf{x}}$ is a vector of correlation lengths, $\oslash$ represents the Hadamard (element-wise) division, and $\|\cdot\|_2$ represents the ℓ^2-norm operation. If we use a uniform mesh ($N_m \times N_m$) to mesh the configuration space, the size of the correlation matrix is $N_m^2 \times N_m^2$.

B. Surrogate Modeling and Pseudo Path Planning

As discussed previously, the high-fidelity simulation by PyChrono is time-consuming for obtaining the reliability map since numerous runs of the simulations are needed. Therefore, we construct a surrogate model using Gaussian Process Regression (GPR) to replace the original PyChrono model.

First, we use Latin Hypercube Sampling (LHS) to generate a group of training data for GPR training according to the range of the uncertain parameters. The training samples of the inputs are denoted by $[\mathbf{S}] = \{\mathbf{s}^{(i)}\}_{i=1}^{N}$ whose corresponding labels (i.e., maximum attainable speed given in Eq. (2)) are obtained by evaluating the PyChrono model and given by $[\mathbf{Y}] = \{y^{(i)}\}_{i=1}^{N}$. We define the training data as $\mathcal{D} = \{\mathbf{s}^{(i)}, y^{(i)}\}_{i=1}^{N}$.

The GPR model is obtained using the training points and is denoted by

$$\hat{y} = G(\mathbf{s}) | \mathcal{D} = \mathbf{f}(\mathbf{s})^{\mathrm{T}} \boldsymbol{\beta} + \zeta, \tag{13}$$

where $\hat{y}$ is the predicted response by the GPR model, $G(\mathbf{s})|\mathcal{D}$ means the GPR model in terms of the terrain-related parameters $\mathbf{s}$ given the training data $\mathcal{D}$, $\mathbf{f}(\cdot)$ is a vector of basis functions, $\boldsymbol{\beta}$ is a vector of the coefficients of the basis functions, and ζ is a noise term following a Gaussian process with the mean of 0 and the variance of σ^2, namely, $\zeta \sim N\left(0, \sigma^2\right)$. To train the GPR model, we maximize the likelihood of the prior distribution and posterior distribution given the training data $\mathcal{D}$. The prior distribution of the training points is estimated by

$$[\mathbf{Y}] \sim N\left(\mathbf{0}, K([\mathbf{S}],[\mathbf{S}]) + \sigma^2 \mathbf{I}_n\right), \tag{14}$$

where $K([\mathbf{S}],[\mathbf{S}])$ is a $N \times N$ symmetric covariance matrix; $\mathbf{I}_n$ is an N-dimensional unit matrix. The Matern kernel is employed to obtain the covariance matrix [48]. Given a set of new samples $[\mathbf{S}^*]$, the joint prior distribution of the training points and new points is given by

$$\begin{bmatrix} [\mathbf{Y}] \\ [\mathbf{Y}^*] \end{bmatrix} \sim N\left(\mathbf{0}, \begin{bmatrix} K_{\mathbf{SS}} + \sigma^2 \mathbf{I}_n & K_{\mathbf{SS}^*}^{\mathrm{T}} \\ K_{\mathbf{SS}^*} & K_{\mathbf{S}^*\mathbf{S}^*} \end{bmatrix}\right), \tag{15}$$

where $[\mathbf{Y}^*]$ is the prediction of GPR by evaluating $[\mathbf{S}^*]$, $K_{\mathbf{SS}}$ is the abbreviation of $K([\mathbf{S}],[\mathbf{S}])$, $K_{\mathbf{SS}^*} = K_{\mathbf{S}^*\mathbf{S}}^{\mathrm{T}}$ is the covariance matrix between training points $[\mathbf{S}]$ and the new points $[\mathbf{S}^*]$, and $K_{\mathbf{S}^*\mathbf{S}^*}$ is the covariance matrix of the new points. After the training of the GPR model, the posterior distribution of an arbitrary point is given by

$$\hat{y} = G(\mathbf{s})|\mathcal{D} \sim N\left(\mu_y(\mathbf{s})|\mathcal{D}, \sigma_y^2(\mathbf{s})|\mathcal{D}\right) \tag{16}$$

where $\mu_y(\mathbf{s})$ and $\sigma_y^2(\mathbf{s})$ are respectively the mean value and variance of the prediction $\hat{y}$ of the GPR model.

As discussed in Sec. IV-A, the random field is used to describe the uncertainty of slope and soil parameters $\mathbf{s} = [s_1, s_2, ..., s_m]^{\mathrm{T}}$. The uncertain parameters can be rewriten as $\mathbf{s}(\mathbf{x}) = [s_1(\mathbf{x}), s_2(\mathbf{x}), ..., s_m(\mathbf{x})]^{\mathrm{T}}$ since they are space-dependent variables. Hence, we have maps of $\hat{y}(\mathbf{x})$ by mapping the spatial-dependent input $\mathbf{s}(\mathbf{x})$ using Eq. (13), where $\hat{y}(\mathbf{x}) \sim N\left(\mu_y(\mathbf{s}(\mathbf{x})), \sigma_y^2(\mathbf{s}(\mathbf{x}))\right)$ The size of $\hat{y}(\mathbf{x})$ is the same as the mesh size in Sec. IV-A. Fig. 5 illustrates the spatial-wise mapping between soil properties and the predicted vehicle mobility.

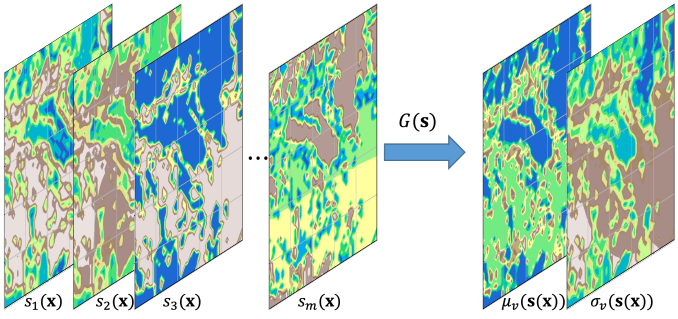

Fig. 5. Illustration of spatial-wise mapping between soil property random realizations and the predicted vehicle mobility maps

As mentioned previously, the AGV mobility of interest in this paper is the speed-made-good [21]. Specifically, a speed threshold y_e is employed to determine whether the AGV loses mobility. We assume that the AGV loses mobility if the speed at a certain location is less than the threshold.

However, the predictions by the GPR surrogate model may not be accurate in some places due to the limited number of training data. This inaccurate prediction can be informed by the surrogate model uncertainty (i.e., epistemic uncertainty). Problems could be introduced if we directly use the above criterion to determine if the AGV fails or not. Hence, we define a pseudo-obstacle event and a pseudo-free event below. The pseudo-obstacle event ($\hat{\mathcal{O}}$) for any given slope and soil parameters $\mathbf{s}$ and after considering surrogate model uncertainty is defined as follows.

$$\hat{\mathcal{O}}(\mathbf{s}) = \left\{ \underbrace{\mu_y(\mathbf{s}) < y_e}_{\text{Obstacle condition}} \cap \underbrace{\Pr\{\hat{y}(\mathbf{s}) \geq y_e | \mu_y(\mathbf{s}) < y_e\} \leq \chi}_{\text{Accurately classified}} \right\}, \tag{17}$$

where χ is the classification accuracy requirement. Recall that $\hat{y}(\mathbf{s})$ from GPR is a random variable due to model uncertainty, $\Pr\{\hat{y}(\mathbf{s}) \geq y_e | \mu_y(\mathbf{s}) < y_e\}$ means the probability that $\hat{y}(\mathbf{s}) \geq y_e$ given the mean value $\mu_y(\mathbf{s})$ of $\hat{y}(\mathbf{s})$ is less than y_e. It means that only if we have very high confidence that the obstacle is accurately classified, we considered it as an "obstacle".

The corresponding pseudo free event ($\hat{\mathcal{F}}$) is given by

$$\begin{aligned} \hat{\mathcal{F}}(\mathbf{s}) &= \bar{\hat{\mathcal{O}}}(\mathbf{s}) \\ &= \{\mu_y(\mathbf{s}) \geq y_e \ \cup \ \Pr\{\hat{y}(\mathbf{s}) \geq y_e | \mu_y(\mathbf{s}) < y_e\} > \chi\}. \end{aligned} \tag{18}$$

As mentioned above, for given $\mathbf{s}$, the GPR prediction follows a Gaussian distribution and thus we have

$$\Pr\{\hat{y}(\mathbf{s}) \geq y_e | \mu_y(\mathbf{s}) < y_e\} = 1 - \Phi\left(\frac{y_e - \mu_y(\mathbf{s})}{\sigma_y(\mathbf{s})}\right) \leq \chi. \tag{19}$$

The above equation is equivalent to

$$U(\mathbf{s}) = \left|\frac{\mu_y(\mathbf{s}) - y_e}{\sigma_y(\mathbf{s})}\right| \geq \left|\Phi^{-1}(\chi)\right|. \tag{20}$$

In this paper, we use $\chi = 0.05$ (i.e., $\left|\Phi^{-1}(\chi)\right| = 2$). We can then rewrite Eqs. (17) and (18) as

$$\hat{\mathcal{O}}(\mathbf{s}) = \left\{ \underbrace{\mu_y(\mathbf{s}) < y_e}_{\text{Obstacle condition}} \cap \underbrace{U(\mathbf{s}) \geq 2}_{\text{Accurately classified}} \right\}, \tag{21}$$

and

$$\hat{\mathcal{F}}(\mathbf{s}) = \{\mu_y(\mathbf{s}) \geq y_e \ \cup \ U(\mathbf{s}) < 2\}. \tag{22}$$

We define an indicator function for the pseudo-free event for given $\mathbf{s}$ as

$$\mathbb{1}_{\mathcal{F}}(\mathbf{s}) = \begin{cases} 1, & \hat{\mathcal{F}}(\mathbf{s}) \text{ is true,} \\ 0, & \hat{\mathcal{O}}(\mathbf{s}) \text{ is true,} \end{cases}. \tag{23}$$

For a given spatial location $\mathbf{x}$, due to uncertainty in slope and soil parameters, the probability that this location is pseudo-free (i.e., pseudo reliability) can be computed as

$$\begin{aligned}(R_F(\mathbf{x})|\mathcal{D}) &= \int \mathbb{1}_{\mathcal{F}}(\mathbf{s}) f_{\mathbf{S}(\mathbf{x})}(\mathbf{s}) d\mathbf{s} \\ &\approx \frac{1}{N_{\text{mcs}}} \sum_{i=1}^{N_{\text{mcs}}} \mathbb{1}_{\mathcal{F}}(\mathbf{s}^{(i)}),\end{aligned} \tag{24}$$

where N_{mcs} is the number of MCS samples used to approximate the integration.

Based on the above equations, we then define the pseudo-free and pseudo-obstacle spaces on the map as

$$\begin{aligned}(\hat{\Omega}_{\text{free}}|\mathcal{D}) &= \{\mathbf{x}|(R_F(\mathbf{x})|\mathcal{D}) \geq R_{\text{lim}}\}, \\ (\hat{\Omega}_{\text{os}}|\mathcal{D}) &= \{\mathbf{x}|(R_F(\mathbf{x})|\mathcal{D}) < R_{\text{lim}}\}.\end{aligned} \tag{25}$$

Since $(R_F(\mathbf{x})|\mathcal{D}) \geq R(\mathbf{x})$, we have

$$\begin{aligned}&\mathbf{x} \in (\hat{\Omega}_{\text{free}}|D),\ \forall \mathbf{x} \in \Omega_{\text{free}} \\ &\because \text{If } \mathbf{x} \in \Omega_{\text{free}}, R(\mathbf{x}) \geq R_{\text{lim}}, \\ &\quad (R_F(\mathbf{x})|D) \geq R(\mathbf{x}) \geq R_{\text{lim}}, \\ &\quad \text{Thus}, \mathbf{x} \in (\hat{\Omega}_{\text{free}}|D).\end{aligned} \tag{26}$$

We, therefore, have the following relationship between the pseudo spaces and the true spaces

$$\Omega_{\text{free}} \subseteq \hat{\Omega}_{\text{free}} \text{ and } \hat{\Omega}_{\text{os}} \subseteq \Omega_{\text{os}}. \tag{27}$$

Fig. 6 shows a pseudo reliability map obtained using Eq. (24) by substituting the coordinates of the configuration space.

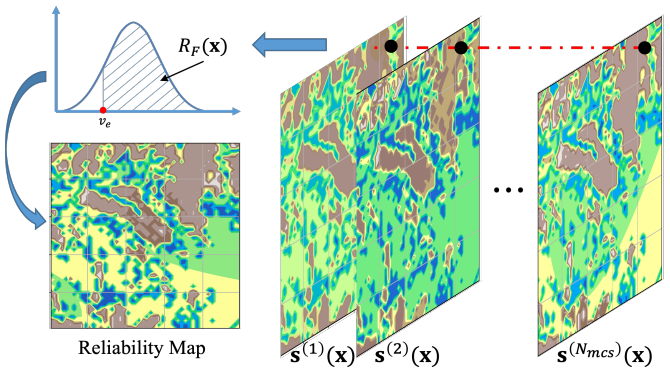

Fig. 6. Reliability map generation by MCS

Once the pseudo spaces are available, the optimal path $(\Gamma^*|\mathcal{D})$ can be identified using the RRT* algorithm in Sec. II-C. The detailed procedure of RRT* is given in Algorithm 1. The optimization model of path planning at this stage is given by

$$\begin{aligned}&(\Gamma^*|\mathcal{D}) = \arg\min_{\Gamma \in \Omega}\{L(\Gamma|\mathcal{D})\} \\ &\text{s.t.} \\ &\mathbf{x}_0 \in \Gamma = \mathbf{x}_{\text{ini}};\ \mathbf{x}_{\text{end}} \in \Gamma = \mathbf{x}_{\text{goal}} \\ &\mathbf{x}_i \in \hat{\Omega}_{\text{free}},\ \forall \mathbf{x}_i \in \Gamma \\ &\hat{\Omega}_{\text{free}} = \{\mathbf{x}|R_F(\mathbf{x})|\mathcal{D} > R_{\text{lim}},\ \forall \mathbf{x} \in \Omega\} \\ &\mathbf{x}_{\text{goal}} = \{\mathbf{x} \in \hat{\Omega}_{\text{free}}|\ \|\mathbf{x} - \mathbf{x}_{\text{goal}}\| \leq \varepsilon\}.\end{aligned} \tag{28}$$

Algorithm 1: RRT*

Data: Ω, $\hat{\Omega}_{\text{os}}$, $\mathbf{x}_{\text{ini}}$, $\mathbf{x}_{\text{goal}}$, $N_{\max}$;
Initialize:: $\mathbf{T} \leftarrow \text{InsertNode}(\mathbf{T}, \emptyset, \mathbf{x}_{\text{ini}})$;
Result: $\mathbf{T}, \Gamma^*$; `/* Γ* - Optimal Path */`
$i = 1$;
while $i < N_{\max}$ **do**
  $i = i + 1$;
  $\mathbf{x}_{\text{rand}} \leftarrow \text{RandomSampling}()$;
  $\mathbf{x}_{\text{nst}} \leftarrow \text{FindNearestNode}(\mathbf{T}, \mathbf{x}_{\text{rand}})$;
  $\mathbf{x}_{\text{new}} \leftarrow \text{Steer}(\mathbf{x}_{\text{nst}}, \mathbf{x}_{\text{rand}})$;
  if $\text{ObstacleFree}(\mathbf{x}_{\text{new}}, \Omega_{\text{os}})$ **then**
    $\mathbf{x}_{\text{nbor}} \leftarrow \text{FindNeighborNodes}(\mathbf{T}, \mathbf{x}_{\text{new}}, r)$;
    `/* See Eq. (4) for r */`
    $\mathbf{x}_p \leftarrow \text{ChooseParent}(\mathbf{x}_{\text{nbor}}, \mathbf{x}_{\text{new}}, \mathbf{x}_{\text{nst}})$;
    $\mathbf{T} \leftarrow \text{InsertNode}(\mathbf{T}, \mathbf{x}_p, \mathbf{x}_{\text{new}})$;
    $\mathbf{T} \leftarrow \text{Rewire}(\mathbf{T}, \mathbf{x}_p, \mathbf{x}_{\text{new}}, \mathbf{x}_{\text{nbor}})$;
  else
    break;
  end
end

C. Surrogate Model Refinement Informed by Path Planning

According to Eq. (25), the pseudo-free space may include obstacles that are not detected by the surrogate model. Hence, we need to verify if the obtained optimal path is reliable or not. In other words, we need to check if the optimal path passes obstacles and use that information to enable targeted refinement of the mobility surrogate model. The path verification is equivalent to verifying if the accuracy of the current surrogate model (i.e., the GPR model) is satisfied. We use the model uncertainty and the U-function in Eq. (20) to check if the GPR model is accurate enough along the optimal path.

We first extract the coordinates of $\Gamma^*|\mathcal{D}$ which are denoted by $\{\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_q\}$. Then, we can find the corresponding random samples at $\{\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_q\}$ by locating the coordinates in the configuration space. All the random samples form a sample pool (MCS population) from which the optimal sample is chosen in active learning. The sample pool is denoted by

$$\mathbb{P} = \left[\{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_1)\}_{i=1}^{N_{\text{mcs}}}, \ldots, \{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_p)\}_{i=1}^{N_{\text{mcs}}}\right]^{\text{T}}, \tag{29}$$

where $\{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)\}_{i=1}^{N_{\text{mcs}}}$ denotes all samples at $\tilde{\mathbf{x}}_j, j = 1, \ldots, p$, $\{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)\}_{i=1}^{N_{\text{mcs}}}$ is given by

$$\{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)\}_{i=1}^{N_{\text{mcs}}} = \begin{bmatrix} s_1^{(1)}(\tilde{\mathbf{x}}_j) & \cdots & s_m^{(1)}(\tilde{\mathbf{x}}_j) \\ \vdots & \ddots & \vdots \\ s_1^{(N_{\text{mcs}})}(\tilde{\mathbf{x}}_j) & \cdots & s_m^{(N_{\text{mcs}})}(\tilde{\mathbf{x}}_j) \end{bmatrix}. \tag{30}$$

Therefore, the total number of samples is $N_T = p \times N_{\text{mcs}}$ and $\mathbb{P} \in \mathbb{R}^{N_T \times m}$.

By evaluating $\mathbb{P}$ using Eq. (13), we have

$$\hat{y}(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)) \sim N\left(\mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)), \sigma_y^2(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right) \tag{31}$$

where $i = 1, \ldots, N_{\text{mcs}}$ represents the index of a MCS sample, and $j = 1, \ldots, q$ represents the index of a location on the path.

We now use active learning to optimally choose a new sample. Substituting $\mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))$ and $\sigma_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))$ into Eq. (20), we have $U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))$. The optimal new training point is identified by

$$\mathbf{s}^* = \min_{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)\in\mathbb{P}} \{U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\}. \tag{32}$$

The corresponding response $\hat{y}^*$ of $\mathbf{s}^*$ is obtained by the high-fidelity model in Eq. (2), Then, the new optimal training point is added to the current training data and the updating of training data is denoted by

$$\mathcal{D} = \{\mathcal{D} \cup [\mathbf{s}^*, \hat{y}^*]\}. \tag{33}$$

Next, we retrain the GPR model using the updated training data following the procedures in Eqs. (13)-(16). The active learning and model updating iteration stops until

$$\min_{\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)\in\mathbb{P}} U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)) \geq 2, \tag{34}$$

or the iteration number exceeds the maximum allowed iteration ($\mathrm{AL}_{\max}$). We have the updated GPR model $G(\mathbf{s})$ upon the end of active learning.

The updating of the reliability map and obstacle map is straightforward by simply following the procedure in Sec. IV-B using the updated GPR model.

D. Convergence Criterion & Tree Updating

After the path verification/surrogate model refinement stage, we have the updated GPR model $G(\mathbf{s})|\mathcal{D}$. Now, we discuss how to determine if the surrogate model is accurate enough for path planning and when to stop the refinement of the surrogate, also known as the convergence criterion. As discussed in Sec. IV-C, we have locations of the optimal path Γ^* given by $\{\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_q\}$. We evaluate the samples $\mathbb{P}$ from Eq. (29) of Γ^* using the updated GPR model, whose responses are denoted by

$$\mathbb{Y} = \left[\{\hat{y}^{(i)}(\tilde{\mathbf{x}}_1)\}_{i=1}^{N_{\mathrm{mcs}}}, \ldots, \{\hat{y}^{(i)}(\tilde{\mathbf{x}}_p)\}_{i=1}^{N_{\mathrm{mcs}}}\right]^{\mathrm{T}}, \tag{35}$$

where $\hat{y}^{(i)} \sim N\left(\mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)), \sigma_y^2(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right)$, $j = 1, \ldots, q$, $i = 1, \ldots, N_{\mathrm{mcs}}$.

The estimated reliability at location $\tilde{\mathbf{x}}_j, j = 1, \ldots, q$ is given by

$$R(\tilde{\mathbf{x}}_j) \approx \frac{1}{N_{\mathrm{mcs}}} \sum_{i=1}^{N_{\mathrm{mcs}}} \mathbb{1}_p\left(\mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right), \tag{36}$$

where the indicator function $\mathbb{1}_p\left(\mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right)$ is given by

$$\mathbb{1}_p\left(\mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right) = \begin{cases} 1, & \mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)) \geqslant y_e \\ 0, & \mu_y(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)) < y_e \end{cases}. \tag{37}$$

Due to surrogate model prediction uncertainty, there is uncertainty in the estimated mobility reliability given in Eq. (36). As mentioned previously, the U function given in Eq. (20) can be used to quantify the accuracy of obstacle classification. Therefore, we use a U-map to indicate the accuracy of the reliability estimate, which is given by

$$P_U(\tilde{\mathbf{x}}_j) \approx \frac{1}{N_{\mathrm{mcs}}} \sum_{i=1}^{N_{\mathrm{mcs}}} I_U\left(U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right), \tag{38}$$

where $P_U(\tilde{\mathbf{x}}_j)$ is defined as the percentage that $U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))$ is smaller than 2 (i.e., percentage of samples that may have a risk of mis-classification) to describe the reliability uncertainty in the configuration space, and the indicator function $I_U\left(U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right)$ is given by

$$I_U\left(U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j))\right) = \begin{cases} 1, & U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)) < 2 \\ 0, & U(\mathbf{s}^{(i)}(\tilde{\mathbf{x}}_j)) \geqslant 2 \end{cases}. \tag{39}$$

If the reliability at $\tilde{\mathbf{x}}_j$ is greater than $R_{\lim}$ and the corresponding uncertainty is less than a threshold $P_{U,\lim}$, we can confidently say that $\tilde{\mathbf{x}}_j$ is not an obstacle, which is mathematically given by

$$\tilde{\mathbf{x}}_j \in \Omega_{\mathrm{free}} | \{R(\tilde{\mathbf{x}}_j) \geq R_{\lim} \,\&\, P_U(\tilde{\mathbf{x}}_j) < P_{U,\lim}\}. \tag{40}$$

According to Eq. (27), $\Omega_{\mathrm{free}} \subseteq \hat{\Omega}_{\mathrm{free}}$. This means that the path identified from the pseudo path planning (i.e., Eq. (28)) should be shorter or equal to the underlying true optimal path ($L(\Gamma^*|\mathcal{D}) \leq L(\Gamma^*)$). If it is shorter than the underlying true optimal path, the path from the pseudo path planning would pass at least one obstacle. Or in other words, if the path from the pseudo path planning (i.e., $\Gamma^*|\mathcal{D}$) does not pass any obstacles, the path can be considered as an approximate of the underlying true optimal path (i.e., Γ^*). Therefore, after verifying that all locations $\{\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_q\} \in \Gamma^*|\mathcal{D}$ of the path identified from the pseudo path planning (i.e., Sec. IV-B) are in the free space, we can say that the proposed method converges. Namely, the convergence criterion is given by

$$\{R(\tilde{\mathbf{x}}_j) \geq R_{\lim} \,\&\, P_U(\tilde{\mathbf{x}}_j) < P_{U,\lim}\},\ \forall \tilde{\mathbf{x}}_j \in (\Gamma^*|\mathcal{D}). \tag{41}$$

Otherwise, the iteration continues until the above criterion is met.

Algorithm 2: Trim tree

Data: $\mathbf{T}$, Ω, $\hat{\Omega}_{\mathrm{os}}$, $\mathbf{x}_{\mathrm{ini}}$, $\mathbf{x}_{\mathrm{goal}}$; `/* Ω̂os - Updated pesudo-obstacle map */`
Initialize:: $\mathbf{T}_{\mathrm{trim}} \leftarrow \mathrm{InsertNode}(\mathbf{T}_{\mathrm{trim}}, \emptyset, \mathbf{x}_{\mathrm{ini}})$;
Result: $\mathbf{T}$;
foreach $\mathbf{x}$ in $\mathbf{T}$ **do**
  $\mathbf{x}_{\mathrm{nst}} \leftarrow \mathrm{FindNearestNode}(\mathbf{T}_{\mathrm{trim}}, \mathbf{x})$;
  $\mathbf{x}_{\mathrm{new}} \leftarrow \mathrm{Steer}(\mathbf{x}_{\mathrm{nst}}, \mathbf{x})$;
  if $\mathrm{ObstacleFree}(\mathbf{x}_{\mathrm{new}}, \Omega_{\mathrm{os}})$ **then**
    $\mathbf{x}_{\mathrm{nbor}} \leftarrow \mathrm{FindNeighborNodes}(\mathbf{T}_{\mathrm{tirm}}, \mathbf{x}_{\mathrm{new}}, r)$; `/* See Eq. (4) for r */`
    $\mathbf{x}_p \leftarrow \mathrm{ChooseParent}(\mathbf{x}_{\mathrm{nbor}}, \mathbf{x}_{\mathrm{new}}, \mathbf{x}_{\mathrm{nst}})$;
    $\mathbf{T}_{\mathrm{trim}} \leftarrow \mathrm{InsertNode}(\mathbf{T}_{\mathrm{trim}}, \mathbf{x}_p, \mathbf{x}_{\mathrm{new}})$;
    $\mathbf{T}_{\mathrm{trim}} \leftarrow \mathrm{Rewire}(\mathbf{T}_{\mathrm{trim}}, \mathbf{x}_p, \mathbf{x}_{\mathrm{new}}, \mathbf{x}_{\mathrm{nbor}})$;
  else
    break;
  end
  $\mathbf{T} = \mathbf{T}_{\mathrm{trim}}$;
end

As discussed previously, we make use of the tree ($\mathbf{T}$) generated by RRT* instead of starting a new exploration every time. Since some of the nodes and branches of $\mathbf{T}$ from previous RRT* might fail to pass the obstacle-free check at the

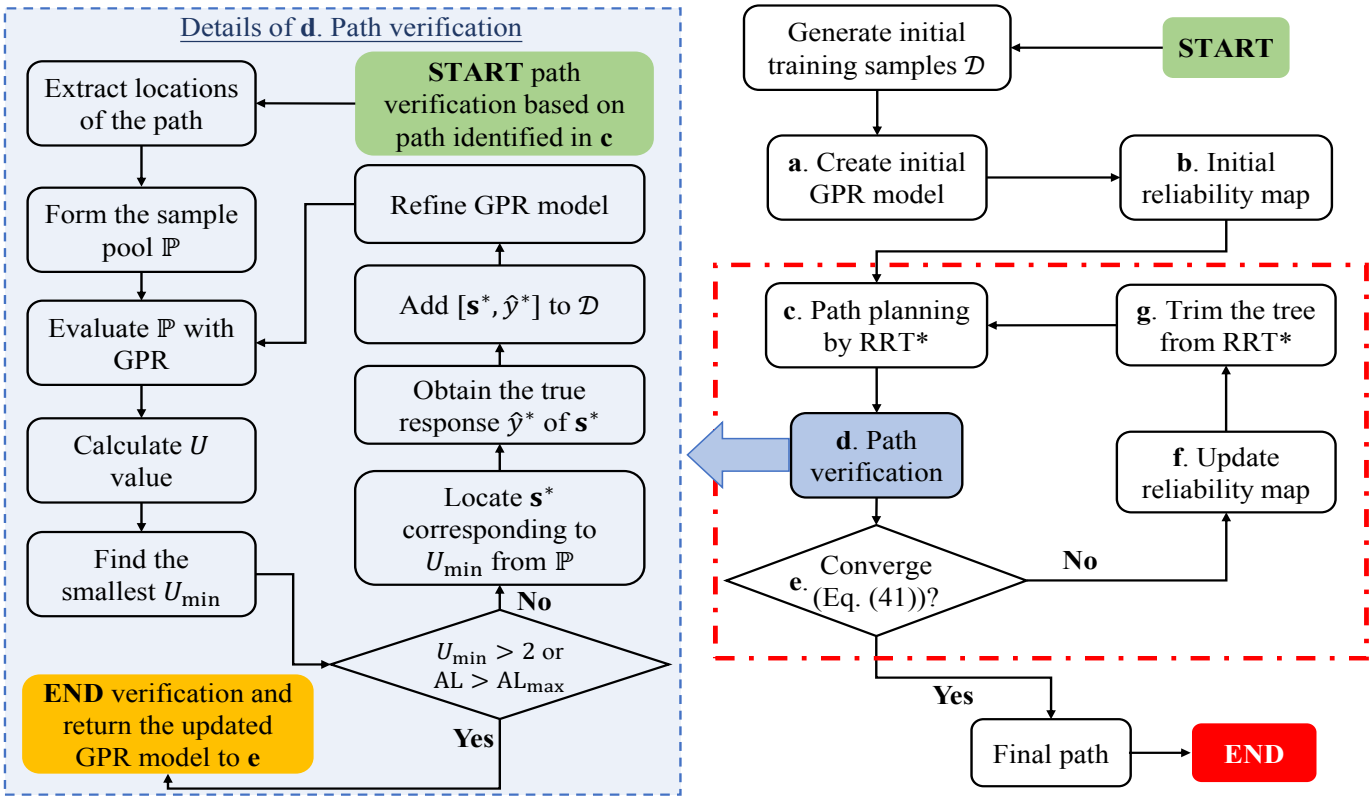

Fig. 7. Flowchart of ER-RRT*. **a**. Creation of an initial surrogate model based on the initial training data; **b**. Generation of reliability information throughout the target map based on the initial GPR model; **c**. Identification of the initial optimal path using RRT*; **d**. Verification of path reliability; **e**. Check if the algorithm converges or not; **f**. Updating of the reliability map based on the updated GPR model from path verification; **g**. Removal of branches of the current tree that are not safe. Detailed descriptions of steps {**a, b, c**}, {**d**}, {**e, f, g**} are given in Section IV-B, Section IV-C, and Section IV-D, respectively.

TABLE I
DISTRIBUTIONS OF SLOPE AND SOIL PARAMETERS.

ID (Slope or Soil)	Slope		Soil Parameters											
	Slope (°)		k_ϕ		k_c		B_n		c (Pa)		Φ (°)		J (m)	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
1	36	2	1×10^9	1,000	5×10^8	500	2.8	0.01	950	1	37.5	0.5	0.048	0.001
2	17.5	1	5×10^8	1,500	1×10^8	750	2.6	0.05	800	2.5	35.0	1.0	0.04	0.002
3	20	2.5	1×10^8	2,000	5×10^7	2,500	2.2	0.1	650	5	32.5	1.5	0.036	0.002
4	40	1.5	5×10^7	500	1×10^7	1,500	2.2	0.02	500	10	30.0	1.0	0.032	0.003
5	60	3	1×10^7	1,000	5×10^6	1,000	2.0	0.1	450	15	27.5	2.0	0.029	0.002
6	56	1.5	5×10^6	2,000	1×10^6	2,000	1.8	0.01	300	5	25.0	0.5	0.024	0.001

updated reliability map and obstacle map, we need to remove those nodes and branches before the next iteration of the path planning.

To achieve this target, we design a trim tree algorithm using the functions of RRT*. The detailed procedure of trim tree is provided in Algorithm 2.

In summary, a flowchart of the proposed method is provided in Fig. 7. In the next section, we use a case study to demonstrate the proposed method.

V. CASE STUDY

We use a 50×50 m map extracted from the ARCGIS/ENVI database to demonstrate the proposed method (ER-RRT*). We compare ER-RRT* with the current two-stage approach to illustrate the advantages of ER-RRT*. The detailed experiment setup of the case study and the results are provided below.

A. Experiment Setup

The target map (configuration space) consists of slope and soil information which are represented by the slope map and soil map. For illustration, an example of the slope map and soil map is provided in Fig. 8. Different colors in the map represent the different slopes and soil types. There are six different slopes and six types of soil exist on the map. Each soil type is characterized by six soil properties, including Bekker coefficients (k_ϕ, k_c, B_n), soil cohesion (c), friction angle (Φ), and Janosi shear displacement (J). Since uncertainty exists in both the slope map and the soil map, we use normal

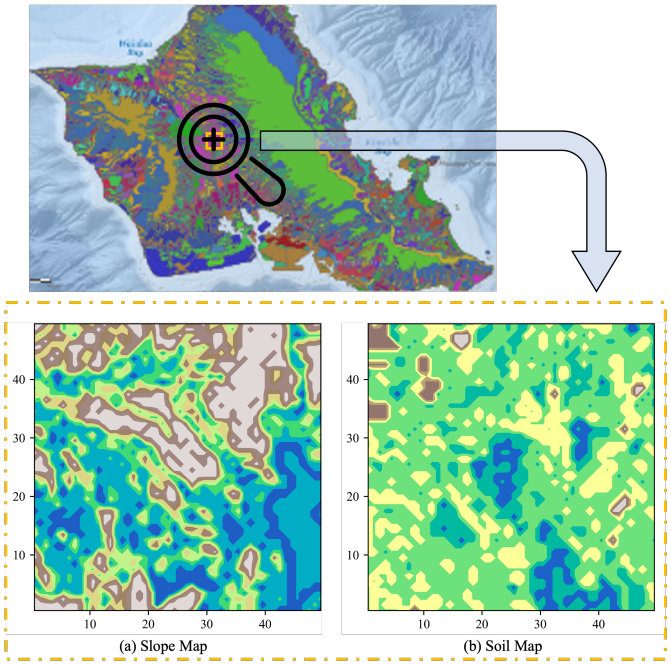

Fig. 8. The slope & soil map extraction

distributions to describe the uncertainty. TABLE I gives the detailed distribution parameters of slope and soil properties.

As discussed previously, the uncertainty is space-dependent and we use the random field to simulate the uncertainty of different parameters. Except for the distributions of different parameters, the correlation length is another important parameter to generate the realizations (samples) of a random field. The detailed correlation lengths of different parameters are provided in TABLE II. The correlation length is represented by a two-dimensional vector since there are two directions on the map. The two elements of a vector represent the correlation length respectively in x_1 and x_2 direction. A smaller value means a larger variation across the configuration space.

TABLE II
CORRELATION LENGTH OF DIFFERENT PARAMETERS IN TWO DIRECTIONS.

ID (Slope or Soil)	Slope	Soil Parameters					
	Slope (°)	k_ϕ	k_c	B_n	c (Pa)	Φ (°)	J (m)
1	[3.3, 3.0]	[40, 39]	[33, 32]	[27, 28]	[27, 28]	[27, 28]	[33, 32]
2	[3.4, 3.6]	[42, 43]	[28, 33]	[23, 27]	[23, 27]	[28, 33]	[28, 33]
3	[4.3, 2, 2]	[41, 41]	[28, 27]	[20, 25]	[28, 27]	[28, 27]	[20, 25]
4	[3.6, 4.4]	[41, 39]	[31, 29]	[26, 24]	[31, 29]	[31, 29]	[26, 24]
5	[3.5, 5.1]	[40, 42]	[33, 32]	[27, 21]	[33, 32]	[33, 32]	[40, 42]
6	[2.8, 5.0]	[38, 37]	[27, 27]	[26, 28]	[27, 27]	[26, 28]	[38, 37]

B. Results

1) The first iteration: Following the procedure in Fig. 7, we generate 50 initial training samples using LHS whose corresponding labels (maximum attainable speed) are obtained from PyChrono simulations. Based on the data, an initial surrogate model is created using Gaussian Process regression. Recall that the AGV loses mobility once the speed is below a speed threshold. The threshold y_e is $2\mathrm{m/s}$. Next, MCS is used to obtain the reliability map and obstacle map for the following pseudo path planning. The number of MCS samples (N_{mcs}) can be determined by

$$COV_{P_f} = \sqrt{\frac{1 - P_f(\mathbf{x})}{P_f(\mathbf{x}) N_{\mathrm{mcs}}}}, \tag{42}$$

where COV_{P_f} is coefficient of variation of MCS, and $P_f(\mathbf{x})$ is the probability of failure. The smaller the COV_{P_f}, the higher the accuracy of MCS and the computational cost. The probability of failure is easily obtained by

$$P_f(\mathbf{x}) = 1 - R_F(\mathbf{x}). \tag{43}$$

In this case study, the minimum reliability requirement along the identified path is 0.9, which means the maximum probability of failure is 0.1. According to Eq. (42), we use $3,600$ MCS samples to achieve a $COV_{P_f} = 0.05$.

By evaluating the MCS samples using the initial GPR, we have the mean predictions and their standard deviations. Then, an uncertainty map (U-map) is generated using Eq. (38) as discussed in Sec. IV-B. The U-map is shown in Fig. 9.

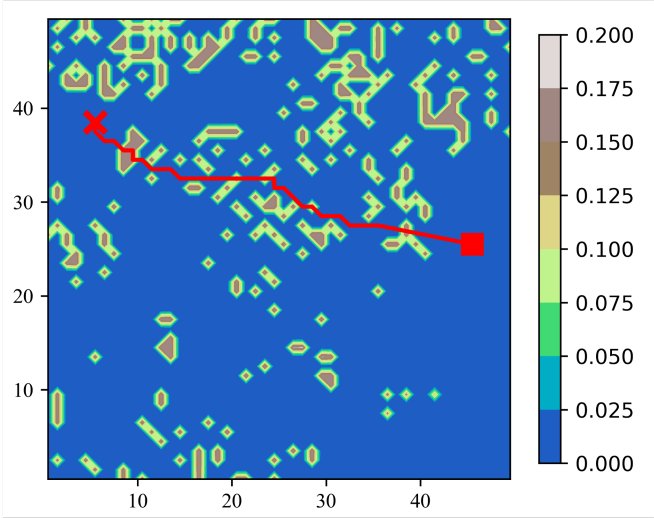

Fig. 9. The U-map generated using the initial GPR model

The U-map indicates the degree of uncertainty in the configuration space with the current surrogate model. Since we do not want to miss the free space due to uncertainty, we use Eq. (24) to generate the pseudo-reliability map. It is found that there is no obstacle identified (the pseudo-reliability is 100% across the map). Based on the pseudo-reliability map, we use the RRT* in Algorithm 1 to identify the first pseudo path which is shown in Fig. 10.

The figure shows that there is no obstacle detected by the initial GPR model. The path identified is close to a straight line from the starting point to the goal. Although the path is short, it may not be reliable. We next verify the path using the method in Sec. IV-C. Following the procedure in Eqs. (29)-(33), we first extract the coordinates (locations) of the path and their corresponding samples to form the sample pool $\mathbb{P}$. Then, the sample pool is evaluated by the GPR model and an

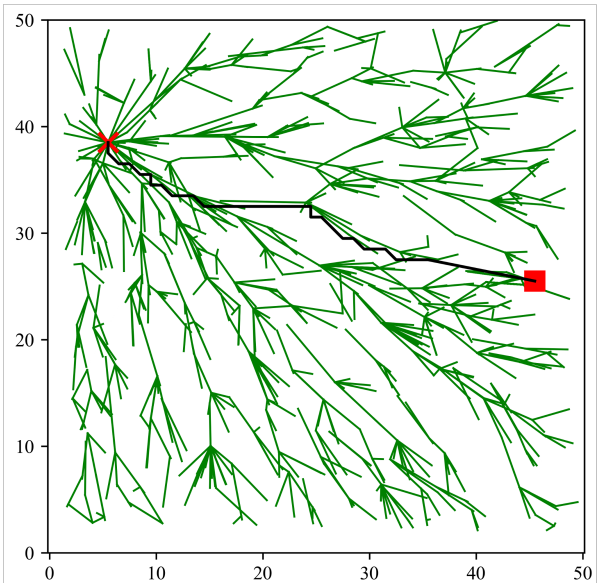

Fig. 10. Initial path obtained by ER-RRT*

optimal new sample is chosen from the sample pool for each iteration of the active learning. The new samples are added to the training data to refine the GPR model. We set the maximum number of allowable iterations of the active learning as 2. After the active learning, the updated reliability map and obstacle map with the current path is shown in Fig. 11.

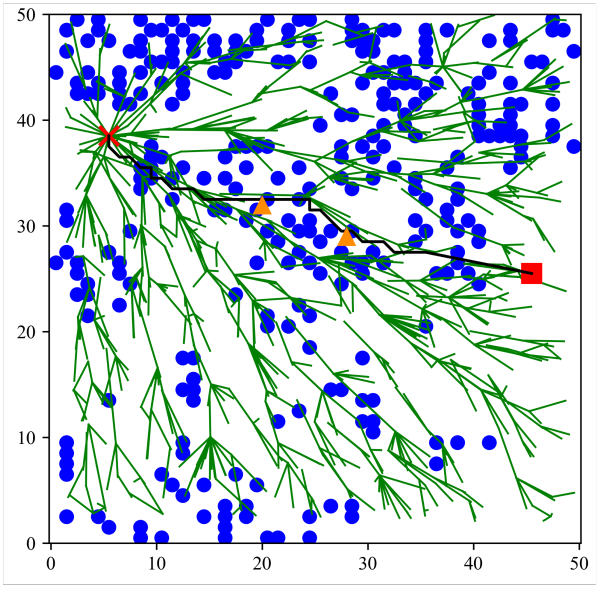

Fig. 11. Verification of the initial path using active learning with updated obstacle map

The locations of the added samples are marked as triangles in Fig. 11. It is also observed that some of the obstacles are detected by the updated GPR model. The current path passes through obstacles as shown in Fig. 11, which means the current path is not reliable and we keep searching of the required path. As discussed in Sec. IV-D, we make use of the current tree by trimming the nodes and branches using Algorithm 2. The trimmed tree is shown in Fig. 12 and serves as the initial tree of the next pseudo plan planning.

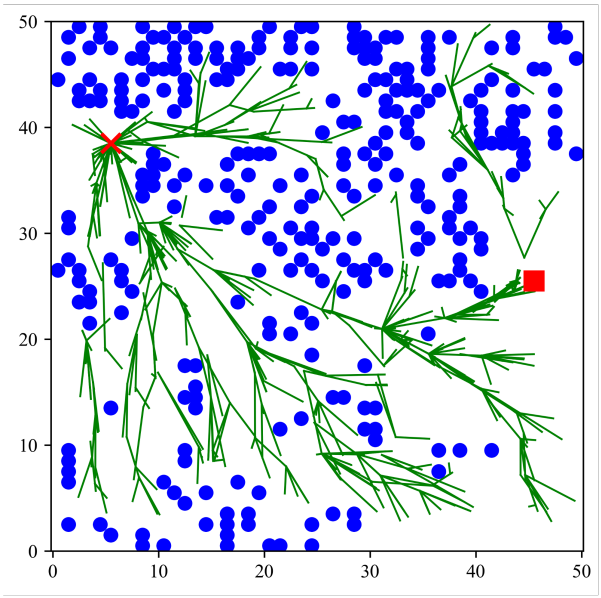

Fig. 12. Trim the tree from the first iteration

2) The second iteration: Based on the initial status in Fig. 12, we use RRT* to perform path planning of the second iteration. Fig. 13 presents the path identified in the second iteration based on the current obstacle map and the uncertainty map. Compared with the uncertainty map in the previous iteration (Fig. 9), the area that the path from the second iteration traverses is less uncertain.

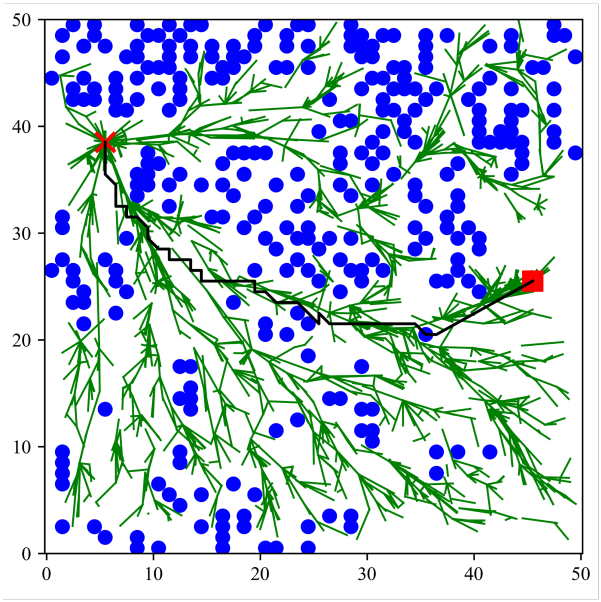

Fig. 13. The second path obtained by ER-RRT* based on the trimmed tree

We verify the path at the current iteration using the same approach as discussed in *the first iteration*. After two active learning iterations, the GPR model is updated. Using the

updated GPR model, we have the updated obstacle map as shown in Fig. 14(a) with the path from the second iteration. Fig. 14(b) shows the path with the updated reliability map. Two random samples as indicated by orange triangles were added in the vicinity of two obstacles as shown in the figure, and the path from the current iteration passes the obstacles.

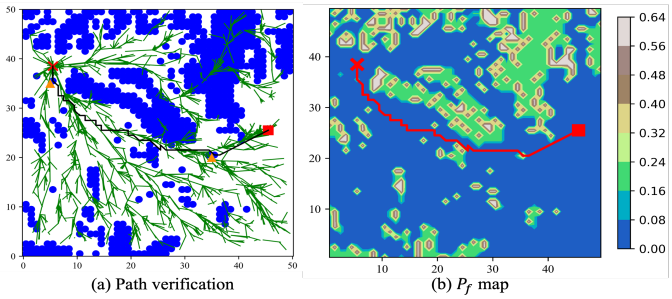

Fig. 14. Verification of the second path with updated obstacle map and P_f map

We trim the current tree again using Algorithm 2. The trimmed tree with the updated obstacle map is shown in Fig. 15. Compared with the tree in Fig. 14, the nodes and branches are trimmed successfully.

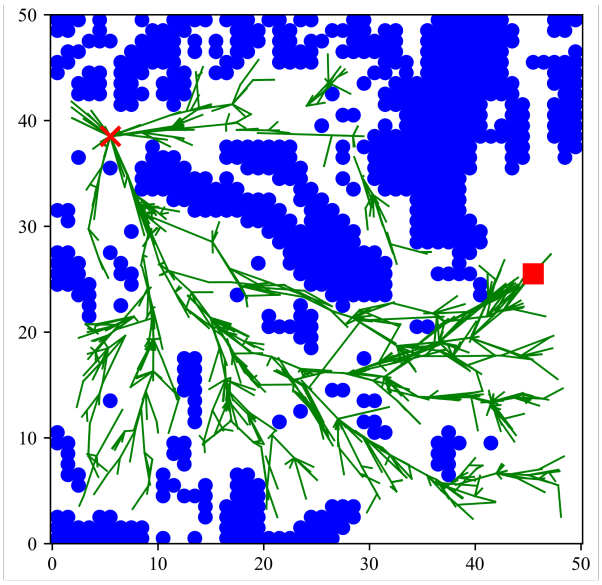

Fig. 15. Trimmed tree of the second iteration

3) Final path: Due to limited space, we only show the details of the first two iterations to illustrate the improvement of the reliability map and obstacle map, and the convergence process of ER-RRT*. After two more iterations, we have the final path as shown in Fig. 16. It is observed that the path identified does not pass any obstacle and is reliable.

We also plot the probability of failure map and the uncertainty map in Fig. 17. The figure shows that the area that the path traverses through is safe with a low probability of failure. Although the uncertainty is not reduced everywhere in the map as shown in Fig. 17(b), the uncertainty has no impact on the identified optimal path. It indicates that the high-fidelity simulation data is used to reduce uncertainty mainly in important regions and thus the proposed ER-RRT* substantially reduces the computational cost for path planning.

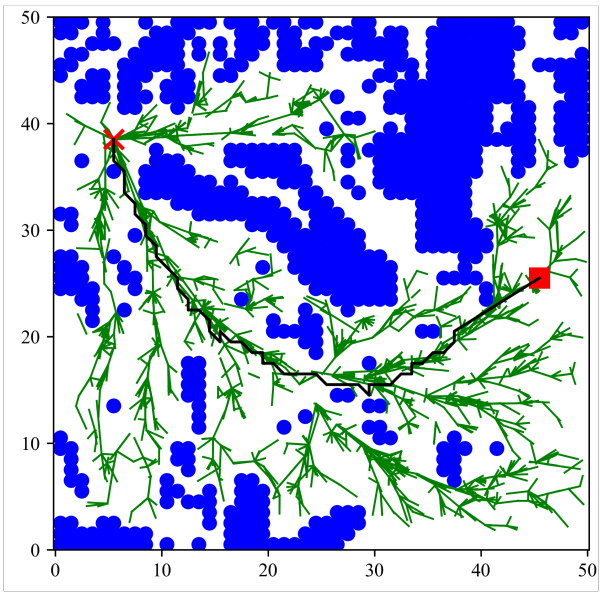

Fig. 16. The final path and obstacle with four iterations

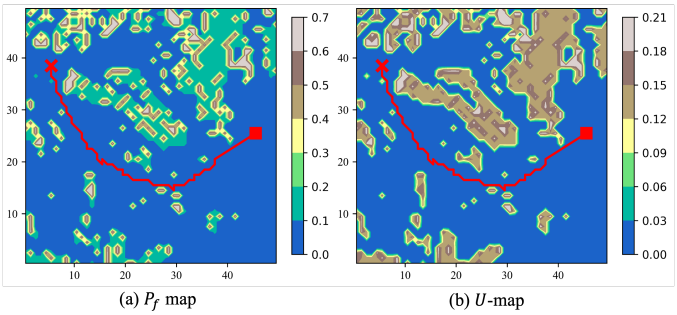

Fig. 17. The final P_f-map and U-map

4) Comparison study: We now quantitatively compare ER-RRT* with the two-stage method and investigate the influence of the maximum allowable active learning iterations on ER-RRT*.

The results of different methods are shown in TABLE III. The two-stage method is the most time-consuming (11.10 h) requiring 469 active learning (AL) iterations to obtain an accurate surrogate model. In addition, the path obtained from the two-stage method is not the shortest one, which may be caused by the random nature of the RRT* method. It has been observed that increasing the maximum allowable active learning iterations results in a reduction in computational time, while decreasing the number of path-planning iterations. This is because an iteration of path planning takes more time than an iteration of active learning in this case. As the maximum allowable active learning iterations increase, the

required computational time also increases. Both the two-stage and the ER-RRT* methods identify paths with similar lengths, with a slight difference. This is because these methods are developed based on RRT* which uses random sampling to generate the paths.

TABLE III
COMPARISON OF DIFFERENT METHODS AND THE INFLUENCE OF MAXIMUM ALLOWED ACTIVE LEARNING (AL) ITERATION.

Method	Max AL iteration	Actual AL iteration	Path planning iteration	Computation time (h)	Path length (m)
Two-stage	1,000	469	1	11.10	63.34
ER-RRT*	1	3	4	0.26	62.21
ER-RRT*	2	6	4	0.33	63.48
ER-RRT*	5	10	3	0.21	58.43
ER-RRT*	10	10	2	0.16	60.23
ER-RRT*	1,000	166	1	0.73	58.21

The computation time of the two-stage method is 11.10 h which is about 14 times longer than that of the proposed ER-RRT* method (0.73 h), if the maximum-allowable number of AL iterations is 1,000. However, the actual number of AL iterations of the two-stage method is 469 which is only about three times the number of AL iterations used in ER-RRT* (166). Intuitively, the number of AL iterations should be proportional to the computation time. The reasons causing these counter-intuitive results are as follows.

- With more samples from active learning added to the training data, a longer time is needed to train a GPR model.
- The two-stage method requires a larger sample pool which means much more samples are evaluated by the GPR model at each AL iteration, thereby a longer computation time for each active learning iteration in the two-stage method.

To investigate the impact of various uncertainty sources on path planning, the proposed method (ER-RRT*) is also compared with two other cases: one that does not take into account uncertainty related to slope and soil parameters, and another that does not consider uncertainty associated with the surrogate model (Fig. 18). For the case where uncertainty related to the slope and soil parameters is not considered, we use the mean values of these parameters from Table I to create a deterministic obstacle map. An optimal path is then identified based on this map. Fig. 18 shows the path in the true obstacle map. It indicates that the deterministic obstacle map without considering uncertainty in the slope and soil parameters fails to detect all potential obstacles. The resulting path is therefore unsafe and passes through obstacles in the true obstacle map. The second case assumes that the surrogate model is accurate after training. A path with a higher risk is identified because some of the obstacles are not identified by the surrogate model. Therefore, it is critical to consider the environmental uncertainty of AGVs to identify potential obstacles and reduce the risk of failure during a mission. The surrogate model constructed by GPR with a limited number of training data may not be accurate in the whole space, which leads to prediction errors and an inaccurate obstacle map. Hence, refining the surrogate model informed by model uncertainty is necessary for accurate and reliable path planning.

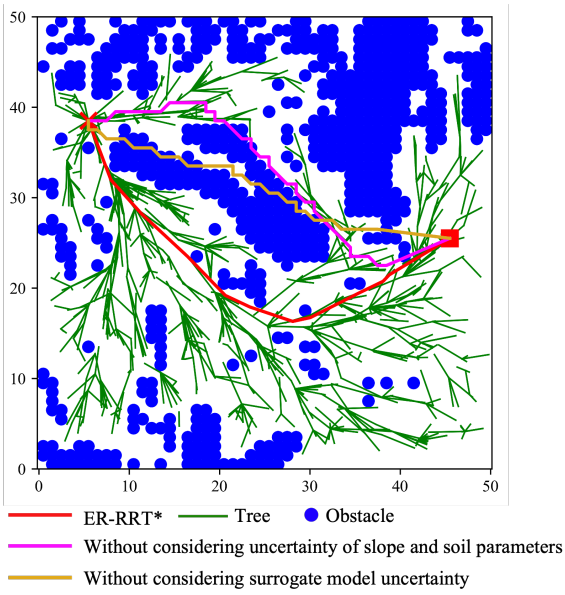

Fig. 18. Comparison study of the identified paths with and without considering different uncertainty sources.

Several case studies were conducted to test the proposed ER-RRT* algorithm, and their results align with the findings presented in the paper. Due to limitations in space, we have only included one typical example with all the necessary details in the paper. This is intended to assist readers in understanding and applying the method more easily.

VI. CONCLUSION

This work proposes an efficient path planning method, named ER-RRT*, for off-road autonomous ground vehicles operating in uncertain working environments. This method couples the adaptive surrogate modeling technique with the rapidly-exploring random trees star algorithm. The results of the case study led to the following conclusions.

- ER-RRT* can successfully identify a reliable path after several iterations of path planning and path verification.
- ER-RRT* is much more efficient than the current two-stage method.
- The maximum number of allowable active learning iterations at the path verification is related to the number of path planning iterations. Reducing the maximum allowable number could lead to an increase in the number of iterations for path planning.

ER-RRT* results in a larger cost due to a longer path for satisfying the reliability constraint. The locations where the reliability does not meet the requirement are treated as obstacles. Therefore, more locations are identified as obstacles than the approach without considering uncertainty. Besides, more computational time is needed if uncertainty is considered in the slope and soil parameters. However, ER-RRT* is still much more efficient than the current two-stage reliability-based path planning approach.

ACKNOWLEDGEMENT

This work was supported in part by the Automotive Research Center (ARC) in accordance with Cooperative Agreement W56HZV-19-2-0001 U.S. Army DEVCOM Ground Vehicle Systems Center (GVSC), Warren, MI, and by the U.S. National Science Foundation under Grant CMMI-2301012. The support is gratefully acknowledged.

REFERENCES

[1] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on intelligent transportation systems*, vol. 17, no. 4, pp. 1135–1145, 2015.

[2] A. Al-Hilo, M. Samir, C. Assi, S. Sharafeddine, and D. Ebrahimi, "Uav-assisted content delivery in intelligent transportation systems-joint trajectory planning and cache management," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 5155–5167, 2020.

[3] H. Wang, Y. Huang, A. Khajepour, Y. Zhang, Y. Rasekhipour, and D. Cao, "Crash mitigation in motion planning for autonomous vehicles," *IEEE transactions on intelligent transportation systems*, vol. 20, no. 9, pp. 3313–3323, 2019.

[4] Y. Rasekhipour, A. Khajepour, S.-K. Chen, and B. Litkouhi, "A potential field-based model predictive path-planning controller for autonomous road vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1255–1267, 2016.

[5] M. Wang, H. Shan, R. Lu, R. Zhang, X. Shen, and F. Bai, "Real-time path planning based on hybrid-vanet-enhanced transportation system," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 1664–1678, 2014.

[6] M. Samir, C. Assi, S. Sharafeddine, D. Ebrahimi, and A. Ghrayeb, "Age of information aware trajectory planning of uavs in intelligent transportation systems: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 12 382–12 395, 2020.

[7] M. Zhu, X.-Y. Liu, and X. Wang, "An online ride-sharing path-planning strategy for public vehicle systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 616–627, 2018.

[8] J. Chen, C. Du, Y. Zhang, P. Han, and W. Wei, "A clustering-based coverage path planning method for autonomous heterogeneous uavs," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[9] H. Jula, M. Dessouky, and P. A. Ioannou, "Truck route planning in nonstationary stochastic networks with time windows at customer locations," *IEEE Transactions on intelligent transportation systems*, vol. 7, no. 1, pp. 51–62, 2006.

[10] L. Ma, J. Xue, K. Kawabata, J. Zhu, C. Ma, and N. Zheng, "Efficient sampling-based motion planning for on-road autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1961–1976, 2015.

[11] A. Thelen, X. Zhang, O. Fink, Y. Lu, S. Ghosh, B. D. Youn, M. D. Todd, S. Mahadevan, C. Hu, and Z. Hu, "A comprehensive review of digital twin—part 2: roles of uncertainty quantification and optimization, a battery digital twin, and perspectives," *Structural and multidisciplinary optimization*, vol. 66, no. 1, pp. 1–43, 2023.

[12] J. E. Naranjo, M. Clavijo, F. Jiménez, O. Gomez, J. L. Rivera, and M. Anguita, "Autonomous vehicle for surveillance missions in off-road environment," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 98–103.

[13] Q. Liu, L. Zhao, Z. Tan, and W. Chen, "Global path planning for autonomous vehicles in off-road environment via an a-star algorithm," *International Journal of Vehicle Autonomous Systems*, vol. 13, no. 4, pp. 330–339, 2017.

[14] R. Chai, A. Tsourdos, H. Gao, Y. Xia, and S. Chai, "Dual-loop tube-based robust model predictive attitude tracking control for spacecraft with system constraints and additive disturbances," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 4, pp. 4022–4033, 2021.

[15] R. Chai, A. Tsourdos, H. Gao, S. Chai, and Y. Xia, "Attitude tracking control for reentry vehicles using centralised robust model predictive control," *Automatica*, vol. 145, p. 110561, 2022.

[16] T. Oksanen and A. Visala, "Coverage path planning algorithms for agricultural field machines," *Journal of field robotics*, vol. 26, no. 8, pp. 651–668, 2009.

[17] R. Chai, D. Liu, T. Liu, A. Tsourdos, Y. Xia, and S. Chai, "Deep learning-based trajectory planning and control for autonomous ground vehicle parking maneuver," *IEEE Transactions on Automation Science and Engineering*, 2022.

[18] R. Chai, H. Niu, J. Carrasco, F. Arvin, H. Yin, and B. Lennox, "Design and experimental validation of deep reinforcement learning-based fast trajectory planning and control for mobile robot in unknown environment," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[19] R. Chai, A. Tsourdos, S. Chai, Y. Xia, A. Savvaris, and C. P. Chen, "Multiphase overtaking maneuver planning for autonomous ground vehicles via a desensitized trajectory optimization approach," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 74–87, 2022.

[20] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, Y. Xia, and C. P. Chen, "Design and implementation of deep neural network-based control for automatic parking maneuver process," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 4, pp. 1400–1413, 2020.

[21] C. Jiang, Z. Hu, Z. P. Mourelatos, D. Gorsich, P. Jayakumar, Y. Fu, and M. Majcher, "R2-rrt*: reliability-based robust mission planning of off-road autonomous ground vehicle under uncertain terrain environment," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 2, pp. 1030–1046, 2021.

[22] Z. Hu, Z. P. Mourelatos, D. Gorsich, P. Jayakumar, and M. Majcher, "Testing design optimization for uncertainty reduction in generating off-road mobility map using a bayesian approach," *Journal of Mechanical Design*, vol. 142, no. 2, 2020.

[23] Y. Liu, C. Jiang, Z. P. Mourelatos, D. Gorsich, P. Jayakumar, Y. Fu, M. Majcher, and Z. Hu, "Simulation-based mission mobility reliability analysis of off-road ground vehicles," *Journal of Mechanical Design*, vol. 143, no. 3, 2021.

[24] Y. Liu, C. Jiang, X. Zhang, Z. P. Mourelatos, D. Barthlow, D. Gorsich, A. Singh, and Z. Hu, "Reliability-based multivehicle path planning under uncertainty using a bio-inspired approach," *Journal of Mechanical Design*, vol. 144, no. 9, p. 091701, 2022.

[25] R. González, P. Jayakumar, and K. Iagnemma, "Stochastic mobility prediction of ground vehicles over large spatial regions: a geostatistical approach," *Autonomous Robots*, vol. 41, no. 2, pp. 311–331, 2017.

[26] E. Petrick, Z. Janosi, and P. Haley, "The use of the nato reference mobility model in military vehicle procurement," SAE Technical Paper, Tech. Rep., 1981.

[27] J. Hetherington, "The applicability of the mmp concept in specifying off-road mobility for wheeled and tracked vehicles," *Journal of terramechanics*, vol. 38, no. 2, pp. 63–70, 2001.

[28] D. Rowland, "Tracked vehicle ground pressure and its effect on soft ground performance," in *Proceedings of the 4th International ISTVS Conference April*, 1972, pp. 24–28.

[29] J. Wong, P. Jayakumar, E. Toma, and J. Preston-Thomas, "A review of mobility metrics for next generation vehicle mobility models," *Journal of Terramechanics*, vol. 87, pp. 11–20, 2020.

[30] M. McCullough, P. Jayakumar, J. Dasch, and D. Gorsich, "The next generation nato reference mobility model development," *Journal of Terramechanics*, vol. 73, pp. 49–60, 2017.

[31] M. Bradbury, J. Dasch, R. Gonzalez, H. Hodges, A. Jain, K. Iagnemma, M. Letherwood, M. Mccllough, J. Priddy, B. Wojtysiak *et al.*, "Next-generation nato reference mobility model (ng-nrmm)," Tank Automotive Research, Development and Engineering Center (TARDEC) WARREN . . . , Tech. Rep., 2016.

[32] M. McCullough, P. Jayakumar, J. Dasch, and D. Gorsich, "Developing the next generation nato reference mobility model," US ARMY TARDEC WARREN United States, Tech. Rep., 2016.

[33] K. Xia, "Finite element modeling of tire/terrain interaction: Application to predicting soil compaction and tire mobility," *Journal of Terramechanics*, vol. 48, no. 2, pp. 113–123, 2011.

[34] A. Recuero, R. Serban, B. Peterson, H. Sugiyama, P. Jayakumar, and D. Negrut, "A high-fidelity approach for vehicle mobility simulation: Nonlinear finite element tires operating on granular material," *Journal of Terramechanics*, vol. 72, pp. 39–54, 2017.

[35] K. Choi, P. Jayakumar, M. Funk, N. Gaul, and T. M. Wasfy, "Framework of reliability-based stochastic mobility map for next generation nato reference mobility model," *Journal of Computational and Nonlinear Dynamics*, vol. 14, no. 2, 2019.

[36] J. Yin, D. Shen, X. Du, and L. Li, "Distributed stochastic model predictive control with taguchi's robustness for vehicle platooning," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[37] D. Shen, J. Yin, X. Du, and L. Li, "Distributed nonlinear model predictive control for heterogeneous vehicle platoons under uncertainty," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 3596–3603.

[38] L. Xiao, R. Wang, B. Dai, Y. Fang, D. Liu, and T. Wu, "Hybrid conditional random field based camera-lidar fusion for road detection," *Information Sciences*, vol. 432, pp. 543–558, 2018.

[39] A. Tasora, R. Serban, H. Mazhar, A. Pazouki, D. Melanz, J. Fleischmann, M. Taylor, H. Sugiyama, and D. Negrut, "Chrono: An open source multi-physics dynamics engine," in *International Conference on High Performance Computing in Science and Engineering*. Springer, 2015, pp. 19–49.
[40] R. Serban, M. Taylor, D. Negrut, and A. Tasora, "Chrono:: Vehicle: template-based ground vehicle modelling and simulation," *International Journal of Vehicle Performance*, vol. 5, no. 1, pp. 18–39, 2019.
[41] C. Z. Mooney, *Monte carlo simulation*. Sage, 1997, no. 116.
[42] J. Yin and X. Du, "Active learning with generalized sliced inverse regression for high-dimensional reliability analysis," *Structural Safety*, vol. 94, p. 102151, 2022.
[43] B. Echard, N. Gayton, and M. Lemaire, "Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation," *Structural Safety*, vol. 33, no. 2, pp. 145–154, 2011.
[44] J. Yin and X. Du, "High-dimensional reliability method accounting for important and unimportant input variables," *Journal of Mechanical Design*, vol. 144, no. 4, 2022.
[45] A. Thelen, X. Zhang, O. Fink, Y. Lu, S. Ghosh, B. D. Youn, M. D. Todd, S. Mahadevan, C. Hu, and Z. Hu, "A comprehensive review of digital twin—part 1: modeling and twinning enabling technologies," *Structural and Multidisciplinary Optimization*, vol. 65, no. 12, pp. 1–55, 2022.
[46] J. Yin and X. Du, "Uncertainty quantification by convolutional neural network gaussian process regression with image and numerical data," in *AIAA SCITECH 2022 Forum*, 2022, p. 1100.
[47] J. Kittler and P. C. Young, "A new approach to feature selection based on the karhunen-loeve expansion," *Pattern recognition*, vol. 5, no. 4, pp. 335–352, 1973.
[48] J. Quinonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate gaussian process regression," *The Journal of Machine Learning Research*, vol. 6, pp. 1939–1959, 2005.

Jianhua Yin (Member) received the B.E. and M.S. degree in Geotechnical Engineering from Chang'an University (2014), Xi'an and Nanjing University (2017), Nanjing, respectively, in China. He received the Ph.D. degree in School of Mechanical engineering, Purdue University, West Lafayette, in 2022. He is currently a research fellow at the University of Michigan. His current research interests include machine learning, uncertainty quantification, and connected autonomous vehicles under uncertainty.

Zhen Hu received B.E. degree in mechanical engineering from Central South University, Changsha, China, in 2008, Master's degree in mechatronics engineering from Huazhong University of Science and Technology, Wuhan, China, in 2011, and Ph.D. degree in mechanical engineering from Missouri University of Science and Technology, Rolla, MO, USA, in 2014. His current research interests include vehicle mobility reliability analysis, design under uncertainty, uncertainty quantification, Bayesian inference, and additive manufacturing.

Zissimos P. Mourelatos is currently a Distinguished Professor of Mechanical Engineering at Oakland University. He has served as the Chair of the Mechanical Engineering Department (2010-2014), and the John F. Dodge Chair of Engineering (2012-2016) at Oakland University. Before joining Oakland University, he spent 18 years at the General Motors Research and Development Center. He received his Ph.D. from the University of Michigan in 1985. Dr. Mourelatos conducts research in the areas of design under uncertainty, reliability-based design optimization, vibrations and dynamics. He has published over 240 journal and conference publications, and is a Fellow of ASME and SAE.

David Gorsich is the Army's Chief Scientist of U.S. Army Combat Capabilities Development Command (DEVCOM) Ground Vehicle Systems Center (GVSC). He received his Ph.D. in applied mathematics from MIT. His current research interests are vehicle dynamics and structural analysis, reliability based design optimization, underbody blast modeling, terrain modeling and spatial statistics. Dr. Gorsich has published more than 150 conference and journal articles. Dr.Gorsich was named a SAE Fellow in 2008 and a ASME Fellow in 2020. He has served on the SAE Technical Standards Board, been the chair for the SAE International Standards Committee for Ground Vehicle Reliability and also on the SAE Board of Directors.

Amandeep Singh is the Branch Chief at US Army, GVSC Vehicle Dynamics and Mobility M&S team. He has over twenty-seven years of technical leadership and research experience in vehicle dynamics, mobility, and reliability working at Government and automotive industry. He received his MS from Clemson University in 1995, and PhD from Oakland University, Michigan in 2010, both in Mechanical Engineering. He is a registered Professional Engineer with the State of Michigan.

Seth Tau is a Mechanical Engineer at the U.S. Army Combat Capabilities Development Command (DEVCOM) Ground Vehicle Systems Center (GVSC). He received his Master's degree in mechanical engineering, in 2019, and his Ph.D. in mechanical engineering, in 2021, both from The Pennsylvania State University. His current research interests include vehicle dynamics and mobility, autonomous vehicle planning algorithms, machine learning, and uncertainty quantification..