R2-RRT*: Reliability-Based Robust Mission Planning of Off-Road Autonomous Ground Vehicle Under Uncertain Terrain Environment

Chen Jiang, Zhen Hu, Zissimos P. Mourelatos, David Gorsich, Paramsothy Jayakumar, Yan Fu, and Monica Majcher

***Abstract*— This article presents a reliable and robust rapidly exploring random tree (R2-RRT*) algorithm to tackle challenges in mission planning of off-road autonomous ground vehicles (AGVs) under uncertain terrain environment. Two types of mobility reliability metrics, namely state mobility reliability (SMR) and mission mobility reliability (MMR), are first defined to quantify the mobility reliability of an AGV and to incorporate mobility reliability into mission planning. SMR measures the probability that a vehicle can pass through a specific location on a map of interest, whereas MMR quantifies mobility reliability of a mission path with the consideration of dependence of soil properties and slope over space. Based on the defined SMR and MMR metrics, two reliability-based robust mission planning models are developed to identify optimal paths that have robust travel time and satisfy specific reliability requirements. Moreover, a reliability-based path smoothing algorithm is developed to address the suboptimality of R2-RRT*. Results of a case study demonstrate the efficacy of the proposed models and algorithms.**

***Note to Practitioners*—This article was motivated to explicitly account for the uncertain terrain environment in mission planning of off-road autonomous ground vehicles (AGVs). Existing approaches, e.g., RRT and its variants, in general, oversimplify the uncertainty sources and overlook the reliability of vehicle mobility. These simplifications could lead to failure (i.e., immobility) of off-road AGVs on the obtained paths. This article suggests a reliability-based mission planning model by incorporating the proposed SMR and MMR metrics into mission planning considering the spatial-dependent uncertainty sources. To identify a reliable and robust path, we extend RRT* to R2-RRT* that achieves a tradeoff between mission cost and reliability. The proposed reliability-based mission planning model, however, is not limited to RRT* and can also be integrated with other path planning algorithms. It can be also applied to other unmanned vehicles and robots, such as the motion planning of unmanned aerial vehicles (UAVs) in adverse weather conditions.**

***Index Terms*— Mission mobility reliability (MMR), off-road autonomous ground vehicle (AGV), rapidly exploring random tree (RRT*), reliability-based mission planning, uncertain terrain environment.**

I. Introduction

MISSION planning has attracted much attention in the field of autonomous vehicles, unmanned aerial vehicles (UAVs), and robotics [1], [2]. It plays a vital role in ensuring that the off-road autonomous ground vehicles (AGVs) can accomplish a mission while meeting certain requirements, such as travel time, fuel consumption, and/or risk.

Identifying a high-quality mission is a complicated task that requires a combination of various techniques, such as computer vision, vehicle mobility simulation, and optimization. In this article, it is assumed that the target map and associated terrain information are well characterized by satellite images and geographical surveys (see Fig. 1). The mission quality is then mainly affected by two phases: 1) mobility-analysis phase that predicts the capability of an off-road AGV passing specific off-road locations and 2) mission-planning phase that identifies an optimal path based on mobility analysis. The off-road vehicle mobility simulation model is usually employed in the mobility-analysis phase to map the environmental variables, such as soil properties, into the vehicle mobility, e.g., vehicle maximum attainable speed. Various simulation models, such as the NATO reference mobility model (NRMM) [3], the next-generation NRMM (NG-NRMM) [4], and the others [5], [6], have been developed for mobility prediction under off-road conditions. In the mission-planning phase, many algorithms, including the graph search methods (e.g., A* algorithm [7]), sampling-based methods [8]–[10], and trajectory/heuristic optimization methods [2], [11], have been proposed to find a feasible mission path.

Despite progress in both two phases, mission planning of off-road AGVs remains a challenge due to the presence of heterogeneous uncertainty sources in extreme environments,

Manuscript received November 15, 2020; accepted January 7, 2021. This article was recommended for publication by Associate Editor C. Yang and Editor C. Seatzu upon evaluation of the reviewers' comments. This work was supported in part by the Automotive Research Center (ARC) through the U.S. Army CCDC Ground Vehicle Systems Center (GVSC), Warren, MI, USA, under Cooperative Agreement W56HZV-04-2-0001. *(Corresponding author: Zhen Hu.)*

Chen Jiang and Zhen Hu are with the Department of Industrial and Manufacturing Systems Engineering, University of Michigan–Dearborn, Dearborn, MI 48128 USA (e-mail: chejiang@umich.edu; 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, Paramsothy Jayakumar, and Monica Majcher are with the U.S. Army Combat Capabilities Development Command, Ground Vehicle Systems Center, Warren, MI 48397 USA (e-mail: david.j.gorsich.civ@mail.mil; paramsothy.jayakumar.civ@mail.mil; monica.t.majcher.civ@mail.mil).

Yan Fu is with the Global Data, Insights, and Analytics, Ford Motor Company, Dearborn, MI 48128 USA (e-mail: yfu4@ford.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASE.2021.3050762.

Digital Object Identifier 10.1109/TASE.2021.3050762

such as vehicle parameter uncertainty and the inherent natural variability in off-road conditions [12]. Ignoring the effects of these uncertainty sources could put the off-road AGVs at risk and lead to the failure of a mission. For instance, due to the complexity and inherent uncertainty in the soil properties, the off-road AGVs may get stuck if the soil is too weak as shown in the video of NG-NRMM [13]. Incorporating mobility reliability into the mission planning under uncertain terrain environments is essential to guarantee the success of a mission and is highly desirable in the development of an NG-NRMM. To develop the capability of reliability-based mission planning, two research questions must be answered: 1) how to quantify the effects of uncertainty sources on vehicle mobility prediction and 2) how to perform mission planning under uncertainty.

In order to answer the first research question, different methods have been proposed in recent years. For example, Gonzalez *et al.* [14], [15] proposed geostatistical approaches to perform mobility prediction over very large spatial regions by considering uncertainty in terrain elevation. Choi *et al.* [16] employed a dynamic Kriging surrogate model to efficiently propagate uncertainty in slope and soil maps to vehicle mobility. However, the current methods only considered the uncertainty in mobility prediction at a given location and overlooked the dependence of soil properties over space [17]. Moreover, current approaches cannot directly quantify the effects of uncertainty sources on the end goal of mission planning.

To address the second research question, many sampling-based planners have been proposed in recent years [8]. For instance, a particle rapidly exploring random tree (RRT) method has been developed to generate robust paths for micro aerial vehicles [18]–[20]. A rapidly exploring random belief tree was proposed to predict a distribution over trajectories for candidate nominal paths with the consideration of state uncertainty [21]. A feedback-based information roadmap framework that considers both motion and sensing uncertainties was proposed by generalizing the probabilistic roadmap method [22], [23]. In addition, the mission risk evaluation under extreme weather was embedded into the 3-D receding horizon field D* searching algorithm to find the optimal fly route for UAVs [24]. Sun *et al.* [25] proposed an optimization-based motion planner for aerial robots by modeling the motion uncertainty as Gaussian distributions [26], [27]. The majority of aforementioned algorithms, however, focus on replanning when the planner thinks that there is uncertainty or error in the measurement of work space or vehicle location, which usually stems from sensor errors, imprecise actuators, and noise [8], [28]–[30]. More importantly, even though current methods quantified the uncertainty in a probabilistic manner, they ignored the spatial dependence of uncertainty sources, which could lead to a high risk of failures. A holistic mission planning approach that accounts for uncertainty sources from mobility prediction to the final mission planning is indispensable to guarantee the reliability of off-road AGVs under uncertain environments.

In this article, we incorporate the spatial-dependent uncertainty in the off-road environment into both mobility-analysis phase and mission-planning phase. To this end, we first represent the uncertainties in the slope and soil properties by Gaussian random fields and generate the random realizations of stochastic mobility maps. The effect of the dependence of slope and soil properties over space on a given mission is quantified by mission mobility reliability (MMR), which measures the probability that a vehicle is able to successfully accomplish a certain path without immobility. Subsequently, an MMR-based mission planning model is proposed and implemented through the proposed reliable and robust RRT* (R2-RRT*) algorithm. In addition, a state mobility reliability (SMR)-based mission planning model is investigated and compared with: 1) the proposed MMR-based mission planning and 2) deterministic mission planning, to demonstrate the efficacy of the proposed method. The proposed R2-RRT* algorithm is able to: 1) quantitatively quantify the effects of uncertain slope and soil properties on mission planning through reliability metrics and 2) identify optimal routes that satisfy specific MMR requirement to achieve an acceptable tradeoff between the time cost (or fuel consumption) and reliability of a mission.

The remainder of this article is organized as follows. Section II introduces the deterministic mission planning of off-road AGVs. Section III proposes two reliability metrics and two reliability-based mission planning models. Section IV introduces the proposed R2-RRT* algorithms. Two case studies are used to demonstrate the proposed algorithms in Section V, followed by conclusions in Section VI.

II. Deterministic Mission Planning of Off-Road Autonomous Ground Vehicle

In this section, we briefly introduce the NG-NRMM and the deterministic mission planning of off-road AGVs.

A. Next-Generation NATO Reference Mobility Model

Fig. 1 presents an overview of the NG-NRMM for mobility prediction of off-road AGVs [4]. As shown in Fig. 1(a), it starts with a target satellite map taken from the ARCGIS/ENVI database [37] and the US Geological Survey database [38], [39]. Let the spatial coordinates in a map of interest $\mathbf{\Omega_x}$ be $\mathbf{x} = [x_1, x_2] \in \mathbb{R}^2$, where x_1 and x_2 are, respectively, the spatial coordinates in directions 1 and 2. In Fig. 1(b), for a given $\mathbf{x} \in \mathbf{\Omega_x}$, the slope $Y_1(\mathbf{x})$ and soil-related environment parameters are identified from the target map through image processing techniques. The soil parameters include the cohesive strength $Y_2(\mathbf{x})$, the friction coefficient $Y_3(\mathbf{x})$, and the bulk density $Y_4(\mathbf{x})$. The slope and soil-related parameters and the vehicle-related parameters $\mathbf{W}$ are then fed into a physics-based modeling & simulation [i.e., Fig. 1(c)] to predict the vehicle mobility at $\mathbf{x}$ as

$$V(\mathbf{x}) = G(\mathbf{W}, \mathbf{Y}(\mathbf{x})) \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{1\times n}$ is the vector of vehicle parameters, $\mathbf{Y} = [Y_1(\mathbf{x}), Y_2(\mathbf{x}), \ldots, Y_m(\mathbf{x})] \in \mathbb{R}^{1\times m}$ is a vector of slope/soil properties that are functions of $\mathbf{x}$, and $V(\mathbf{x})$ represents the vehicle mobility, e.g., the maximum attainable speed in this article. $G(\cdot, \cdot)$ denotes the NG-NRMM, and the aforementioned mobility models, such as NRMM [3]

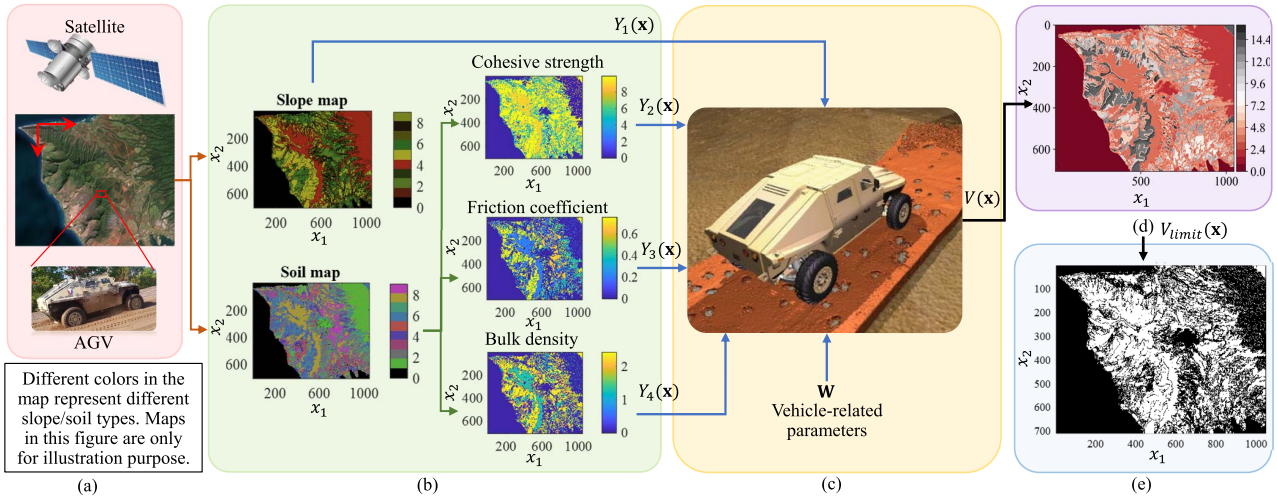


Fig. 1. Illustration of mobility prediction using NG-NRMM. The target map and associated terrain information are characterized from (a) satellite image, including (b) slope-/soil-related environment parameters. The environment parameters and vehicle-related parameters are fed into (c) physics-based modeling and simulation to generate (d) speed-made-good map and (e) GO/NO-GO map.

and Bekker's derived shterramechanics model (BDTM) [4], can be also employed. A more detailed introduction of NG-NRMM is available at [13]. With vehicle mobility quantified in each spatial coordinate of target map, as indicated in Fig. 1(d) and (e), a speed-made-good map and a GO/NO-GO map can be generated.

Note that, for the sake of explanation, $\mathbf{Y}(\mathbf{x})$ in (1) contains both slope-related and soil-related variables. Since vehicle parameters $\mathbf{W}$ depend on vehicle types, this article focuses on only soil/slope-related variables. The proposed approaches, however, can be easily extended to include vehicle parameters.

B. Mission Planning-Based on NG-NRMM

After obtaining the GO/NO-GO mobility map in Fig. 1(e), which is also called configuration/work space, the map of interest is represented as $\Omega_{\mathbf{x}} = \chi_{\text{free}} \cup \chi_{\text{obs}}$, where χ_{free} and χ_{obs} are, respectively, the free space and obstacle space, which are defined by

$$\chi_{\text{free}} = \{\mathbf{x} \in \Omega_{\mathbf{x}} | V(\mathbf{x}) \geq V_{\text{limit}}\} \tag{2}$$

and

$$\chi_{\text{obs}} = \{\mathbf{x} \in \Omega_{\mathbf{x}} | V(\mathbf{x}) < V_{\text{limit}}\} \tag{3}$$

in which V_{limit} is a minimum speed threshold to maintain the vehicle mobility. If the vehicle speed at a location is less than V_{limit}, the vehicle will get stuck at that location. Note that V_{limit} will be one of the mission requirements if the mission task must be accomplished within a certain time. This work is only concerned with whether a vehicle will get stuck on a route.

The deterministic mission planning model is then formulated as

$$\begin{aligned} \psi^* = \arg\min_{\psi \in \Omega_{\mathbf{x}}} \; & C(\psi) \\ \text{s.t. } & \psi(t_0) = \mathbf{x}_{\text{init}}, \;\; \psi(t_e) \in \chi_{\text{goal}} \\ & \psi(t_i) \in \chi_{\text{free}}, \;\; t_i \in [t_0, t_e] \\ & \chi_{\text{goal}} = \left\{\mathbf{x} \in \chi_{\text{free}} \middle| \left\|\mathbf{x} - \mathbf{x}_{\text{goal}}\right\| \leqslant \varepsilon\right\} \\ & \chi_{\text{free}} = \{\mathbf{x} \in \Omega_{\mathbf{x}} | V(\mathbf{x}) \geqslant V_{\text{limit}}\} \end{aligned} \tag{4}$$

where ψ denotes a possible mission path on a map of interest $\Omega_{\mathbf{x}}$, $\mathbf{x}_{\text{init}}$ and $\mathbf{x}_{\text{goal}}$ are, respectively, the initial and target coordinates of a mission, $\psi(t_0)$ and $\psi(t_e)$ are, respectively, the first and last element of the coordinate set $\psi \subset \Omega_{\mathbf{x}}$, $\psi(t_i)$ is the element on path ψ at t_i, χ_{goal} denotes the goal region, ε is a small constant to define the size of goal region, χ_{free} denotes the travel-free space, and $C(\psi)$ is a cost function as

$$C(\psi) = \sum_{t_i=t_0}^{t_e} \frac{1}{V(\psi(t_i))} \tag{5}$$

where $V(\psi(t_i)), \;\; \forall t_i \in [t_0, t_e]$ is the maximum attainable speed at the $\psi(t_i)$th element on the mission path ψ, which is predicted using the mobility M&S model given in (1).

Given a target map, $\mathbf{x}_{\text{init}}$, and $\mathbf{x}_{\text{goal}}$, an obstacle-free path can be identified in the work space using path planning algorithms. In this article, RRT* [31] is adopted and is compared with the proposed reliability-based mission planning algorithms. It is worth noting that the deterministic mission planning ignores the effects of uncertain slope-/soil-related environment parameters on mobility prediction, which could lead to catastrophic consequences due to the immobility of off-road AGVs on the battlefield.

III. Reliability-Based Mission Planning of Off-Road AGVs

In this section, we first discuss how to propagate the uncertainty of slope-/soil-related environment parameters to vehicle mobility. Following that, we present the proposed SMR-based and MMR-based mission planning models.

A. Uncertainty in Off-Road Vehicle Mobility Prediction

For given $\mathbf{x} \in \Omega_{\mathbf{x}}$, a slope/soil ID can be identified from the slope/soil map. As shown in Fig. 2, an ID matrix [i.e., Fig. 2(b)] with different IDs representing different slope/soil types is identified from a map of interest [Fig. 2(a)]. Each ID denotes a type of slope/soil, such as sand, grass, and sandstone. Moreover, each slope ID (i.e., each type of slope) is associated with a slope property $Y_1(\mathbf{x})$, and each soil

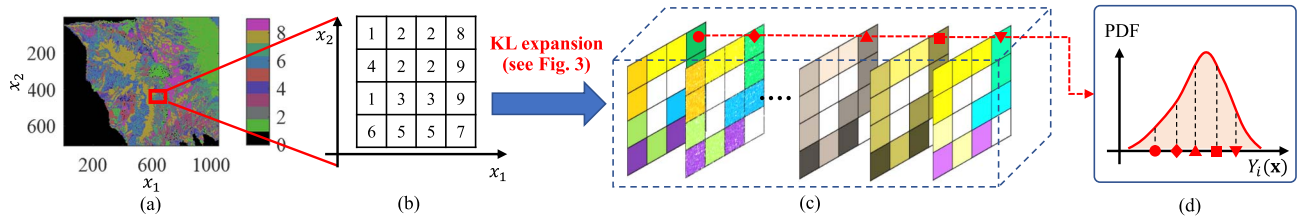


Fig. 2. Uncertainty in the slope-/soil-related environment parameters. (a) Slope/soil map is transformed into (b) slope/soil ID matrix by classifying the slope/soil parameters into several ID types. Each slope ID matrix is associated with one slope property matrix, while each soil ID matrix is associated with three soil property matrices. (c) Random realizations of each property are simulated by KL expansion method (see Fig. 3), and each element of the property matrix is (d) random variable.

ID is associated with three different soil properties, i.e., the cohesive strength $Y_2(\mathbf{x})$, the friction coefficient $Y_3(\mathbf{x})$, and the bulk density $Y_4(\mathbf{x})$, as illustrated in Fig. 1(b). Consequently, there are two ID matrices and four property matrices.

The slope and soil properties vary both over space and for a given location due to a lack of knowledge and natural variability. Moreover, the variability of each property at one location is affected by those at other locations attributed to geostatistical dependencies over space. To account for the dependence of slope/soil properties, the property matrix corresponding to the slope or soil ID matrix can be, respectively, simulated as random fields with many random realizations. Fig. 2(c) shows random realizations obtained from Karhunen–Loève (KL) expansion of one property, as detailed in Fig. 3 and the Appendix. Each element of the property matrix (i.e., at a particular location) is a random variable [see Fig. 2(d)]. Details about the random field modeling are given in the following.

The identified slope or soil ID at a certain coordinate $\mathbf{x}$ is represented as $\alpha(\mathbf{x})$ or $\beta(\mathbf{x})$, respectively. Uncertainty of slope/soil properties $Y_i(\mathbf{x}), i = 1, \ldots, m$ at that coordinate is described as a random variable with cumulative density function (CDF) given by

$$F_{Y_i}(y) = \Pr\{Y_i \leqslant y\} = \int_{-\infty}^{y} f_{Y_i}(\upsilon,\ \theta(\zeta(\mathbf{x})))d\upsilon \tag{6}$$

in which $F_{Y_i}(y)$ is the CDF of $Y_i(\mathbf{x})$, $\Pr\{\cdot\}$ is a probability operator, and $f_{Y_i}(\upsilon,\ \theta(\zeta(\mathbf{x})))$ is the probability density function (pdf) of $Y_i(\mathbf{x})$ with distribution parameters $\theta(\zeta(\mathbf{x}))$ (e.g., mean and variance) governed by $\zeta(\mathbf{x})$, which means that $\theta(\zeta(\mathbf{x}))$ varies with slope/soil IDs. $\zeta(\mathbf{x}) = \alpha(\mathbf{x})$ if $Y_i(\mathbf{x})$ is a slope-related variable (e.g., $Y_1(\mathbf{x})$), and $\zeta(\mathbf{x}) = \beta(\mathbf{x})$ if $Y_i(\mathbf{x})$ is a soil-related variable [e.g., $Y_2(\mathbf{x})$, $Y_3(\mathbf{x})$, or $Y_4(\mathbf{x})$]. To account for the dependence of slope/soil properties over space, $Y_i(\mathbf{x})$ are further described as random fields.

Since there are multiple slope or soil types on a map of interest $\Omega_{\mathbf{x}}$, each $Y_i(\mathbf{x})\ \forall \mathbf{x} \in \psi$ consists of multiple random fields. To address this challenge in the simulation of the slope-/soil-related random fields, we first partition $\Omega_{\mathbf{x}}$ into different subsets according to the slope/soil IDs. Taking the slope parameter as example, the coordinates $\mathbf{x}_{\text{slope}}^{(p)}$ corresponding to the pth slope ID shown in Fig. 3(a) and (b) are identified as

$$\mathbf{x}_{\text{slope}}^{(p)} = \{\mathbf{x} : \text{where } \alpha(\mathbf{x}) = p,\ \forall \mathbf{x} \in \Omega_{\mathbf{x}}\} \tag{7}$$

where $\alpha(\mathbf{x})$ is the slope ID at location $\mathbf{x}$ in the slope map, $p = 1, \ldots, N_{\text{slope}}$, N_{slope} is the number of slope types, and we have

$$\left\{\mathbf{x}_{\text{slope}}^{(1)} \cup \cdots \cup \mathbf{x}_{\text{slope}}^{(N_{\text{slope}})}\right\} = \Omega_{\mathbf{x}}. \tag{8}$$

Similarly, the coordinates of the qth soil ID are identified as

$$\mathbf{x}_{\text{soil}}^{(q)} = \{\mathbf{x} : \text{where } \beta(\mathbf{x}) = q,\ \forall \mathbf{x} \in \Omega_{\mathbf{x}}\} \tag{9}$$

where $\beta(\mathbf{x})$ is the soil ID at the location $\mathbf{x}$ in the soil map, $q = 1, \ldots, N_{\text{soil}}$, N_{soil} is the number of soil types, and $\{\mathbf{x}_{\text{soil}}^{(1)} \cup \cdots \cup \mathbf{x}_{\text{soil}}^{(N_{\text{soil}})}\} = \Omega_{\mathbf{x}}$.

After the partition, the slope/soil property corresponding to a certain slope/soil ID is simulated using the KL expansion method (as detailed in the Appendix) [32]. The KL expansion uses the best possible linear combination of orthogonal basis functions that minimize the mean squared error. The basis functions are determined by the covariance function (i.e., spatial dependence) of random fields, and the coefficients of basis functions are random variables [32]. Even though the Gaussian random fields are adopted in this article, the developed approach is also applicable if non-Gaussian random fields are employed.

As shown in Fig. 3(c), N_{MCS} realizations are simulated for each type of slope property, where N_{MCS} is the number of samples in the Monte Carlo simulation (MCS). $Y_i(\mathbf{x}), \forall \mathbf{x} \in \Omega_{\mathbf{x}}$, which represents the slope or soil-related property over the region of interest $\Omega_{\mathbf{x}}$, can then be expressed as a union of multiple random fields shown in Fig. 3(d) and given as follows:

$$Y_i(\mathbf{x}) = \left\{\mathbf{Y}_{i1} \cup \cdots \cup \mathbf{Y}_{iN_T}\right\} \quad \forall \mathbf{x} \in \Omega_{\mathbf{x}} \tag{10}$$

where N_T is given by

$$N_T = \begin{cases} N_{\text{slope}}, & \text{if } Y_i(\mathbf{x}) \text{ is a slope-related parameter} \\ N_{\text{soil}}, & \text{if } Y_i(\mathbf{x}) \text{ is a soil-related parameter} \end{cases} \tag{11}$$

$\mathbf{Y}_{ij} \triangleq \{Y_{ij}(1), Y_{ij}(2), \ldots, Y_{ij}(N_{ij})\}\ \forall i = 1, \ldots, m;\ j = 1, \ldots, N_T$ is a single random field associated with the jth slope/soil ID, and N_{ij} is the number of elements in subset $\mathbf{x}_{\text{slope}}^{(j)}$ or $\mathbf{x}_{\text{soil}}^{(j)}$ (i.e., the number of locations with the jth slope/soil ID on the map). For the example in Fig. 3, we have $N_T = N_{\text{slope}} = 9$, and N_{i1}, N_{i2}, and N_{i9} equal to 3, 4, and 5, respectively.

For $Y_i(\mathbf{x})$ corresponding to the jth slope/soil ID (i.e., $\mathbf{Y}_{ij}$), its realizations are generated using KL expansion based on the

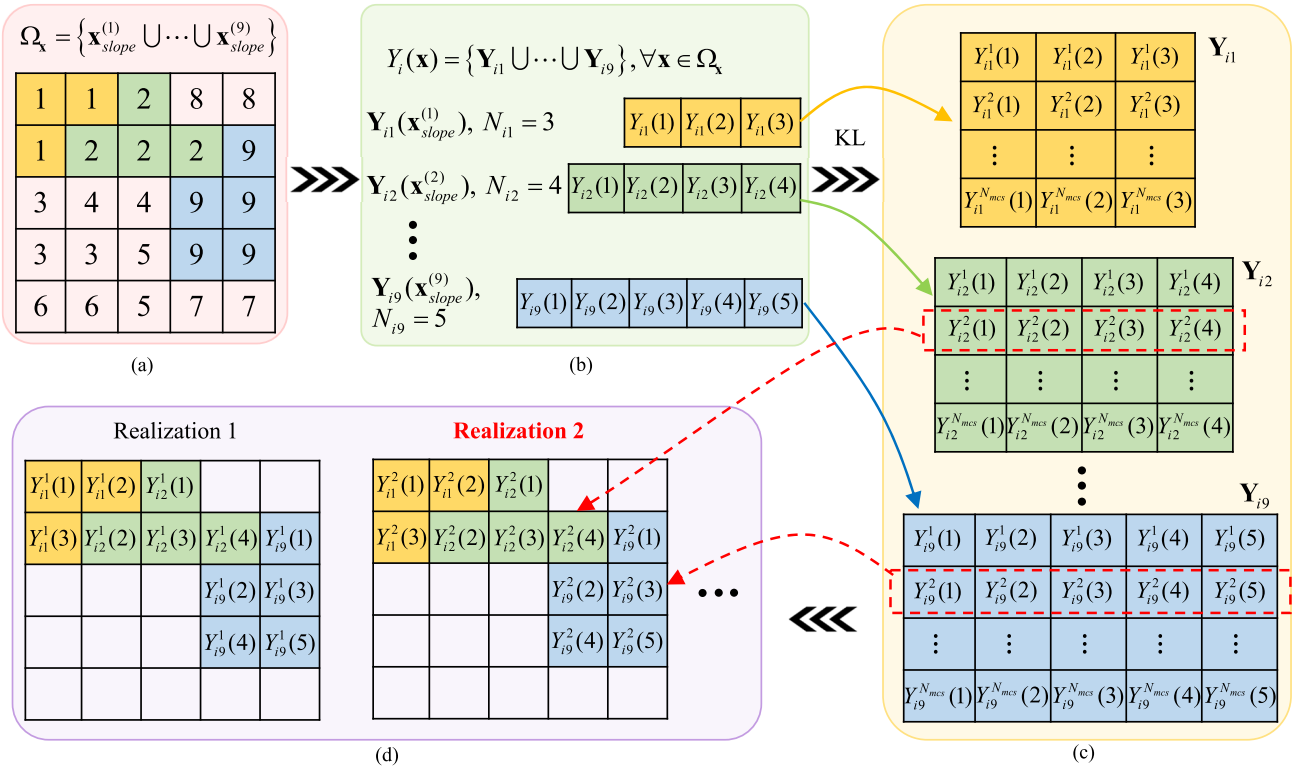


Fig. 3. Random field simulation of slope property using KL expansion. (a) Slope ID matrix is partitioned into nine subsets according to slope IDs. Nine types of slope properties corresponding to different coordinates are obtained for each subset in (b). KL is then used to obtain (c) N_{MCS} random field realizations for each property Y_{ij}. After that, (d) N_{MCS} realizations of slope property matrices are expressed as the union of multiple random fields according to their spatial coordinates.

following correlation matrix:

$$\rho_{ij} = \begin{bmatrix} 1 & \cdots & \rho_{ij}(\mathbf{x}_s^{(1)}, \mathbf{x}_s^{(N_{ij})}) \\ \vdots & \ddots & \vdots \\ \rho_{ij}(\mathbf{x}_s^{(N_{ij})}, \mathbf{x}_s^{(1)}) & \cdots & 1 \end{bmatrix} \tag{12}$$

where $\rho_{ij}(\cdot,\cdot)$ is the correlation function [i.e., (43)] of $Y_{ij}(k)$, $\mathbf{x}_s^{(k)} \triangleq \mathbf{x}_{\text{slope}}^{(j)}(k)$ or $\mathbf{x}_s^{(k)} \triangleq \mathbf{x}_{\text{soil}}^{(j)}(k)$ and $k = 1, \ldots, N_{ij}$. Using (7) through (12), we generate N_{MCS} random realizations for $\mathbf{Y}_{ij}$ $\forall i = 1, \ldots, m$, $j = 1, \ldots, N_T$. In Fig. 3(d), realizations of $\mathbf{Y}_{ij}$ are then assembled together according to their coordinates, and random realizations of $Y_i(\mathbf{x})$, $\forall \mathbf{x} \in \Omega_{\mathbf{x}}$ are generated.

Following the above procedure of Fig. 3, different random realizations of the slope property matrix, the cohesive strength property matrix, the friction coefficient property matrix, and the bulk density property matrix are generated, as shown in Fig. 4(a). These matrices are then fed into the physics-based mobility simulation model to generate random realizations of vehicle mobility map, as illustrated in Fig. 4(c), rather than a single deterministic vehicle mobility map [e.g., Fig. 1(d)]. Next, we will propose two reliability-based mission planning schemes with gradually increased complexity based on the uncertainty propagation of vehicle mobility, as discussed earlier.

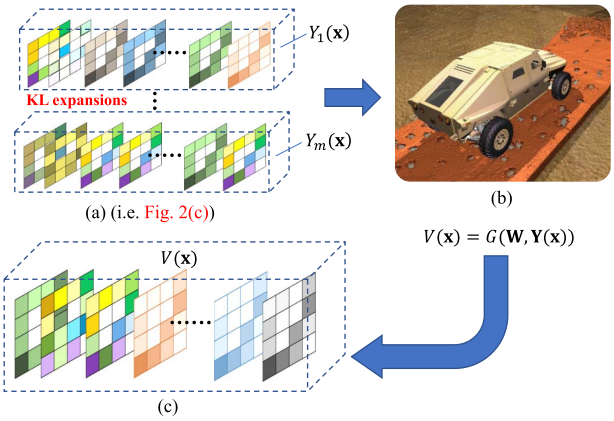


Fig. 4. Uncertainty propagation from environment parameters to mobility. (a) Realizations of slope/soil property matrices obtained in Fig. 2 are fed into (b) physics-based off-road mobility model, yielding (c) random realizations of vehicle mobility.

B. State Mobility Reliability-Based Mission Planning

1) State Mobility Reliability Analysis: As illustrated in Fig. 5(a) and (b), SMR, $R^{\text{SMR}}(\mathbf{x})$, is defined as the probability of maintaining vehicle mobility at a certain location, $\mathbf{x}$, and is given by

$$R^{\text{SMR}}(\mathbf{x}) = \Pr\{V(\mathbf{x}) = G(\mathbf{W}, \mathbf{Y}(\mathbf{x})) \geq V_{\text{limit}}\}. \tag{13}$$

Performing the SMR analysis using (13) involves the evaluation of the following equation:

$$R^{\text{SMR}}(\mathbf{x}) = \int_{V(\mathbf{x}) \geqslant V_{\text{limit}}} f_{\mathbf{Y}}(\upsilon) \mathbf{d}\upsilon \tag{14}$$

where $f_{\mathbf{Y}}(\upsilon)$ is the joint pdf of $\mathbf{Y}(\mathbf{x}) = [Y_1(\mathbf{x}), \ldots, Y_m(\mathbf{x})]$. If $Y_1(\mathbf{x}), \ldots, Y_m(\mathbf{x})$ are considered independent, for any given $\mathbf{x}$, $f_{\mathbf{Y}}(\upsilon)$ is given by

$$f_{\mathbf{Y}}(\upsilon) = \prod_{i=1}^{m} f_{Y_i}(\upsilon_i,\ \theta(\zeta(\mathbf{x}))) \tag{15}$$

in which $f_{Y_i}(\upsilon_i,\ \theta(\zeta(\mathbf{x})))$ is the pdf defined in (6).

By defining the following indicator function:

$$I(V(\mathbf{x})) = \begin{cases} 1, & V(\mathbf{x}) \geqslant V_{\text{limit}} \\ 0, & V(\mathbf{x}) < V_{\text{limit}}. \end{cases} \tag{16}$$

Equation (14) is written as

$$R^{\text{SMR}}(\mathbf{x}) = \int_{\mathbb{R}^d} I(V(\mathbf{x}))\, f_{\mathbf{Y}}(\upsilon)\mathbf{d}\upsilon = E[I(V(\mathbf{x}))] \tag{17}$$

where $E[\cdot]$ is the expectation operator.

Directly solving the multidimensional integral in (17) is computationally prohibitive, especially when $V(\mathbf{x}) = G(\mathbf{W}, \mathbf{Y}(\mathbf{x}))$ is computationally expensive computer simulations, as mentioned earlier. To overcome this challenge, a computationally cheap machine learning surrogate model (e.g., the Gaussian process model and neural networks) is usually constructed to substitute the original model [17]. After that, sampling-based approximation methods, such as MCS, are used to estimate (17). Based on the MCS random realizations generated in Section III-A, (17) is approximated as

$$R^{\text{SMR}}(\mathbf{x}) \approx \frac{1}{N_{\text{MCS}}} \sum_{h=1}^{N_{\text{MCS}}} I(\hat{G}(\mathbf{W}, \mathbf{y}^{(h)}(\mathbf{x}))) \tag{18}$$

where $\mathbf{y}^{(h)}(\mathbf{x}), h = 1, \ldots, N_{\text{MCS}}$, is the hth MCS sample of $\mathbf{Y}(\mathbf{x})$, $\hat{G}(\mathbf{W}, \mathbf{y}^{(h)}(\mathbf{x}))$ is the mobility surrogate model prediction for given $\mathbf{y}^{(h)}(\mathbf{x})$ at location $\mathbf{x}$, and $I(\cdot)$ is the indicator function defined in (16).

2) SMR-Based Mission Planning: By performing SMR analysis at all locations using (13), an SMR map can be generated, as shown in Fig. 5(c). Based on this SMR map, an intuitive way to include reliability into mission planning is to require that the mobility reliability be greater than a certain threshold at all locations on an obstacle-free mission. In SMR-based mission planning, the free space of the mobility map can, therefore, be defined as

$$\chi_{\text{free}} = \left\{\mathbf{x} \in \Omega_{\mathbf{x}} \middle| R^{\text{SMR}}(\mathbf{x}) \geq [R]\right\} \tag{19}$$

where $[R]$ represents a user-defined reliability level. This kind of map is often referred to as a probabilistic mobility map, as shown in Fig. 5(d). For different $[R]$, different probabilistic mobility maps can be generated. The value of $[R]$ is usually determined by the decision-maker according to the level of risk that the decision-maker is willing to take.

After introducing (19) into (4), we have the SMR-based mission planning model as

$$\begin{aligned} \psi^* = \arg\min_{\psi \in \Omega_{\mathbf{x}}}\ & C^{\text{SMR}}_{\text{robust}}(\psi) \\ \text{s.t. } & \psi(t_0) = \mathbf{x}_{\text{init}},\ \ \psi(t_e) \in \chi_{\text{goal}} \\ & \psi(t_i) \in \chi_{\text{free}},\ \ t_i \in [t_0, t_e] \end{aligned}$$

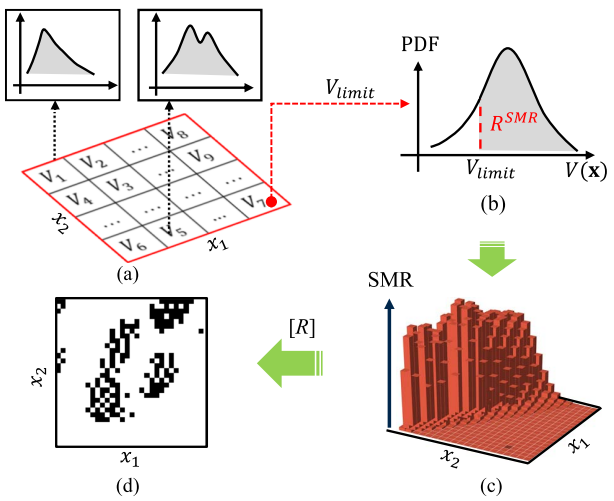


Fig. 5. SMR analysis. Based on (a) uncertain vehicle mobility prediction and (b) SMR at each location is calculated, and thus, we obtain (c) SMR map. Using a user-defined reliability level $[R]$, (d) probabilistic mobility map is obtained.

$$\begin{aligned} \chi_{\text{goal}} &= \left\{\mathbf{x} \in \chi_{\text{free}} \middle| \left\|\mathbf{x} - \mathbf{x}_{\text{goal}}\right\| \leqslant \varepsilon\right\} \\ \chi_{\text{free}} &= \left\{\mathbf{x} \in \Omega_{\mathbf{x}} \middle| R^{\text{SMR}}(\mathbf{x}) \geqslant [R]\right\} \end{aligned} \tag{20}$$

in which $C^{\text{SMR}}_{\text{robust}}(\psi)$ is a travel time cost function formulated from robust design perspective such that the required travel time of the identified mission path is robust to the uncertainty in the mission planning environment.

The robust cost function is defined as follows:

$$C^{\text{SMR}}_{\text{robust}}(\psi) = w\mu_{\text{SMR}}(C(\psi)) + (1-w)\sigma_{\text{SMR}}(C(\psi)) \tag{21}$$

where $C(\psi)$ is calculated by (5) and w is a weight factor balancing the mean and standard deviation of $C(\psi)$. Since the slope/soil properties are assumed to be Gaussian random variables in this article, $\mu_{\text{SMR}}(C(\psi))$ and $\sigma_{\text{SMR}}(C(\psi))$ are given by

$$\mu_{\text{SMR}}(C(\psi)) = \sum_{t_i=t_0}^{t_e} \mu\left(\frac{1}{V(\psi(t_i))}\right) \tag{22}$$

and

$$\sigma_{\text{SMR}}(C(\psi)) = \sqrt{\sum_{t_i=t_0}^{t_e} \sigma^2\left(\frac{1}{V(\psi(t_i))}\right)} \tag{23}$$

where t_0 and t_e are, respectively, the initial and end time nodes of mission path ψ, and $\mu(1/V(\psi(t_i)))$ and $\sigma^2(1/V(\psi(t_i)))$ are, respectively, the mean cost and variance at location $\psi(t_i)$.

Even though SMR-based mission planning takes into account the uncertainty in the slope/soil parameters, it overlooks the spatial dependence of the vehicle mobility caused by the spatial dependence of slope-/soil-related properties. As shown in the results of case studies, ignoring the spatial dependence of vehicle mobility could significantly increase the risk of immobility in mission planning. Next, an MMR-based mission planning approach is proposed to overcome this limitation of SMR-based mission planning.

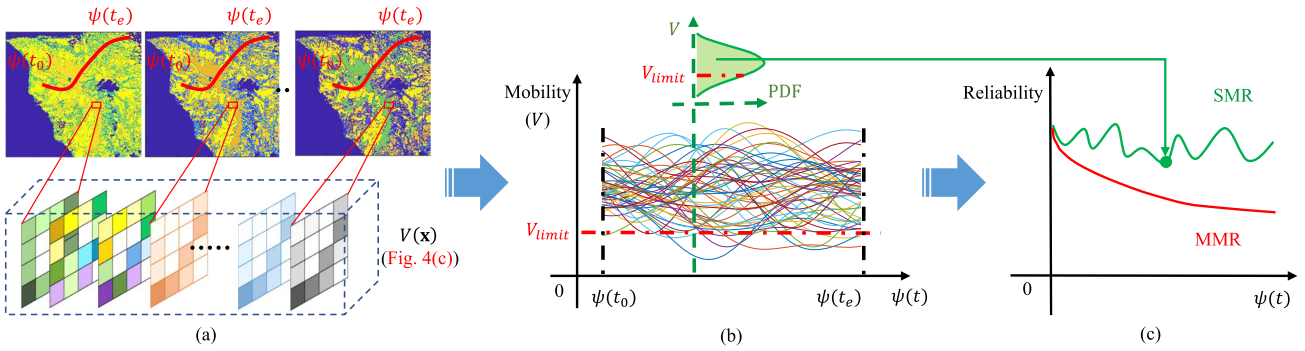


Fig. 6. MMR analysis. (a) Random realizations of a stochastic mobility map are obtained through uncertainty quantification and propagation. For a given mission path from $\psi(t_0)$ to $\psi(t_e)$ [denoted by red line in (a)], (b) random realizations of mobility on a mission path can be extracted from (a). Based on (b) and (c), the MMR of the whole mission and the SMR at each location are estimated.

C. Mission Mobility Reliability-Based Mission Planning

1) Mission Mobility Reliability Analysis: In contrast to deterministic mission analysis and SMR analysis that focus on mobility analysis at individual locations, MMR aims to quantify the probability that a vehicle can successfully accomplish a mission without immobility directly from the mission planning perspective [17]. An MMR for a given mission path, ψ, is defined as follows [17]:

$$R^{\text{MMR}}(\psi) = \Pr\{V(\mathbf{x}) \geq V_{\text{limit}},\ \forall \mathbf{x} \in \psi\} \tag{24}$$

where ψ is a set of location/coordinates $\mathbf{x}$ on a mission path and $\psi \subset \Omega_{\mathbf{x}}$ is a subset of a map/region of interest $\Omega_{\mathbf{x}}$.

From the generated realizations of mobility map shown in Fig. 4(c), realizations of mobility (i.e., maximum attainable speed) on the mission path ψ, which is a subset of $\Omega_{\mathbf{x}}$, can be extracted as illustrated in Fig. 6(a) and (b). Defining realizations of the mobility on ψ as $\mathbf{V}_{\text{re}}(\psi) = \{v(h, t_i), h = 1, \ldots, N_{\text{MCS}}, t_i = t_0, \ldots, t_e\}$, where $v(h,\ t_i)$ is the hth realization of the mobility at the $\psi(t_i)$th location on the mission path ψ, $\psi(t_0)$ and $\psi(t_e)$ are, respectively, the initial and end locations that ψ passes through, MMR of ψ in (24) is then estimated as [33], [34]

$$\begin{aligned} R^{\text{MMR}}(\psi) &\approx \frac{1}{N_{\text{MCS}}} \sum_{h=1}^{N_{\text{MCS}}} I\left(\min_{t_i=t_0,\ldots,t_e} v(h,\ t_i)\right) \\ &= \frac{1}{N_{\text{MCS}}} \sum_{h=1}^{N_{\text{MCS}}} \prod_{t_i=t_0}^{t_e} I(v(h,\ t_i)) \end{aligned} \tag{25}$$

where $I(v(h, t_i)) = 1$ if $v(h, t_i) \geqslant V_{\text{limit}}$; otherwise, $I(v(h, t_i)) = 0$.

$I(v(h, t_i)) = 0$ denotes the immobility at location $\psi(t_i)$, which results in failure of the mission for the hth realization. Only when the vehicle does not get stuck at all locations $\psi(t_i)$, i.e., $I(v(h, t_i)) = 1, \forall t_i \in [t_0, t_e]$, we will have $\min_{t_i=t_0,\ldots,t_e} v(h, t_i) \geqslant V_{\text{limit}}$, and then, $I(\min_{t_i=t_0,\ldots,t_e} v(h, t_i)) = 1$. Otherwise, $I(\min_{t_i=t_0,\ldots,t_e} v(h, t_i)) = 0$. This indicates that MMR is equal to the ratio of number of failed realizations to N_{MCS}, and the failure of each realization is related to the extreme value (i.e., minimum value in this case) of $v(h,\ t_i)$ on the mission path ψ. For each realization of vehicle mobility on the mission ψ, the possibility of $I(\min_{t_i=t_0,\ldots,t_e} v(h, t_i)) = 0$ is monotonically increasing over the mission path since the extreme value of $v(h, t_i)$ is monotonically decreasing as t_i increases from t_0 to t_e. Therefore, the further the vehicle goes, the lower the MMR is, as depicted in Fig. 6(c). Lower MMR indicates higher risk of failing to accomplish the planned mission.

2) MMR-Based Mission Planning: The basic idea of the MMR-based mission planning is to account for effects of spatial-dependent uncertainty sources on the planner and guarantee the mobility reliability directly from mission planning perspective. In the past, an approach has been developed to account for the risk of a mission in mission planning by approximate the MMR using SMR as follows [24]:

$$\tilde{R}^{\text{MMR}}(\psi) = \prod_{t_i=t_0}^{t_e} R^{\text{SMR}}(\psi(t_i)) \tag{26}$$

where $R^{\text{SMR}}(\psi(t_i))$ is the SMR estimated by (18) at the t_ith location, i.e., $\psi(t_i)$ on the path ψ.

However, the above approximation holds only when the spatial dependence of the vehicle mobility is ignored. Ignoring the spatial dependence will lead to large errors in the MMR prediction and, thus, affect the mobility reliability of the identified path. According to (18), the approximated $\tilde{R}^{\text{MMR}}(\psi)$ is written by

$$\begin{aligned} \tilde{R}^{\text{MMR}}(\psi) &\approx \prod_{t_i=t_0}^{t_e} \frac{1}{N_{\text{MCS}}} \sum_{h=1}^{N_{\text{MCS}}} I(v(h,\ t_i)) \\ &= \frac{1}{N_{\text{MCS}}^{N_\psi}} \prod_{t_i=t_0}^{t_e} \sum_{h=1}^{N_{\text{MCS}}} I(v(h,\ t_i)) \end{aligned} \tag{27}$$

where N_ψ is the number of cells/locations that ψ passes through, i.e., the number of time nodes between t_0 and t_e. It is evident that the results calculated by (27) and (25) are prone to be different, which means that the approximated MMR given in (26) cannot really represent the true MMR.

To address this problem and effectively guarantee MMR during mission planning, the MMR constraint given in (24) is included into the mission planning model. In addition, to reduce the search design space during mission planning, we further introduce an SMR-related constraint into the model

and yield the MMR-based mission planning model as follows:

$$\begin{aligned}\psi^* = &\arg\min_{\psi\in\Omega_{\mathbf{x}}} \ C^{\mathrm{MMR}}_{\mathrm{robust}}(\psi)\\ &\text{s.t. } \psi(t_0)=\mathbf{x}_{\mathrm{init}},\ \ \psi(t_e)\in\chi_{\mathrm{goal}}\\ &\quad \psi(t_i)\in\chi_{\mathrm{free}},\ \ t_i\in[t_0,t_e]\\ &\quad \chi_{\mathrm{goal}}=\left\{\mathbf{x}\in\chi_{\mathrm{free}}\middle|\left\|\mathbf{x}-\mathbf{x}_{\mathrm{goal}}\right\|\leqslant\varepsilon\right\}\\ &\quad \chi_{\mathrm{free}}=\left\{\mathbf{x}\in\Omega_{\mathbf{x}}\middle|R^{\mathrm{SMR}}(\mathbf{x})\geqslant[R]\right\}\\ &\quad R^{\mathrm{MMR}}(\psi)=\Pr\{V(\psi(t_i))\geqslant V_{\mathrm{limit}}\quad\forall t_i\in[t_0,t_e]\}\geqslant[R]\end{aligned} \tag{28}$$

where $C^{\mathrm{MMR}}_{\mathrm{robust}}(\psi)$ is a travel time cost function formulated from robust design perspective by considering the mobility dependence over space. Similar to (21), $C^{\mathrm{MMR}}_{\mathrm{robust}}(\psi)$ is calculated as follows:

$$C^{\mathrm{MMR}}_{\mathrm{robust}}(\psi)=w\mu_{\mathrm{MMR}}(C(\psi))+(1-w)\sigma_{\mathrm{MMR}}(C(\psi)) \tag{29}$$

in which $C(\psi)$ is calculated by (5) and w is a weight factor balancing the mean and standard deviation of $C(\psi)$.

Since $C(\psi)$ is related to the vehicle mobility (i.e., maximum attainable speed), the calculation of $C^{\mathrm{MMR}}_{\mathrm{robust}}(\psi)$ is different from that of $C^{\mathrm{SMR}}_{\mathrm{robust}}(\psi)$ due to the spatial dependence of uncertainty sources. Based on realizations of the mobility on ψ, $\mu_{\mathrm{MMR}}(C(\psi))$ and $\sigma_{\mathrm{MMR}}(C(\psi))$ are given by

$$\mu_{\mathrm{MMR}}(C(\psi))=\frac{1}{N_{\mathrm{MCS}}}\sum_{h=1}^{N_{\mathrm{MCS}}}\sum_{t_i=t_0}^{t_e}\frac{1}{v(h,\ t_i)} \tag{30}$$

and

$$\begin{aligned}&\sigma_{\mathrm{MMR}}(C(\psi))\\ &=\sqrt{\frac{1}{N_{\mathrm{MCS}}}\sum_{h=1}^{N_{\mathrm{MCS}}}\left(\mu_{\mathrm{MMR}}(C(\psi))-\sum_{t_i=t_0}^{t_e}\frac{1}{v(h,\ t_i)}\right)^2}\end{aligned} \tag{31}$$

where t_0 and t_e are, respectively, the initial and end time nodes of mission path ψ, and $\sum_{t_i=t_0}^{t_e}\frac{1}{v(h,\ t_i)}$ is the time travel cost of the hth realization of ψ.

Reducing the search design space using SMR-related constraint $\psi(t_i)\in\chi_{\mathrm{free}},\ \chi_{\mathrm{free}}=\{\mathbf{x}\in\Omega_{\mathbf{x}}|R^{\mathrm{SMR}}(\mathbf{x})\geqslant[R]\}$ helps reduce the computational time during the mission planning since it can provide a prior feasible region without optimization, as explained in detail in the algorithm section. Evaluating the constraint $R^{\mathrm{MMR}}(\psi)\geqslant[R]$ in (28) is equivalent to evaluating the following constraint:

$$R^{\mathrm{MMR}}(\psi)=\Pr\left\{\min_{t_i\in[t_0,t_e]}\{V(\psi(t_i))\}\geqslant V_{\mathrm{limit}}\right\}\geqslant[R]. \tag{32}$$

Since $\min_{t_i\in[t_0,t_e]}\{V(\psi(t_i))\}\leqslant V(\psi(t_i))\forall t_i\in[t_0,t_e]$, comparing (32) with (13) yields

$$R^{\mathrm{MMR}}(\psi)\leqslant R^{\mathrm{SMR}}(\psi(t_i))\quad\forall t_i\in[t_0,t_e]. \tag{33}$$

The above equation implies that $R^{\mathrm{MMR}}(\psi)\geqslant[R]$ is a stricter constraint than $R^{\mathrm{SMR}}(\psi(t_i))=\Pr\{V(\psi(t_i))\geqslant V_{\mathrm{limit}}\}\geqslant[R]\ \forall t_i\in[t_0,t_e]$. If ψ satisfies the constraint $R^{\mathrm{SMR}}(\psi(t_i))\geqslant[R]\ \forall t_i\in[t_0,t_e]$, it may or may not satisfy the constraint that $R^{\mathrm{MMR}}(\psi)\geqslant[R]$. On the other hand, if ψ satisfies the constraint $R^{\mathrm{MMR}}(\psi)\geqslant[R]$, $R^{\mathrm{SMR}}(\psi(t_i))\geqslant[R]\ \forall t_i\in[t_0,t_e]$ will be satisfied automatically. It implies that introducing the SMR-related constraint into the MMR-based mission planning model given in (28) will not affect the optimality of the original model. However, it can help reduce the search space. Next, we provide detailed algorithms of proposed reliability-based mission planning approaches in Section IV-A.

IV. Main R2-RRT* Algorithms

In this section, we discuss how to extend the RRT* algorithm [31] to solve the above-discussed SMR- and MMR-based mission planning models, where the resulting algorithms are named R2-RRT*-1 and R2-RRT*-2, respectively. Following that, a reliability-based path smoothing algorithm is proposed to address the suboptimality of the identified path.

A. Reliability-Based Robust Mission Planning by R2-RRT*

1) R2-RRT-1 for SMR-Based Mission Planning Model:* Algorithm 1 explains the procedure of R2-RRT*-1. It is initialized with a tree containing one initial vertex $\mathbf{x}_{\mathrm{init}}$ and no edge. Given the slope/soil map, a probabilistic mobility map, i.e., work space for R2-RRT*-1, is generated through a function SMR(map, V_{limit}, $[R]$), as shown in Fig. 5(d). In SMR(), we calculate $R^{\mathrm{SMR}}(\mathbf{x})$ at each location using (18) with a predefined V_{limit}. By setting a reliability level $[R]$, the free and obstacle space of target map are generated using (19).

After initialization, the tree is incrementally updated by adding new vertex and updating edges related to the new vertex. To identify the new vertex, a new point $\mathbf{x}_{\mathrm{rand}}$ is randomly sampled by function PSample(χ_{free}, $R^{\mathrm{SMR}}(\mathbf{x})$). In PSample(), we first define a threshold $\lambda\in[0,1)$, and λ is set to 0.1 in this article. In each iteration, if a number randomly generated over range [0, 1] is greater than λ, then the goal point will be considered as $\mathbf{x}_{\mathrm{rand}}$. Otherwise, the probability of sampling a certain point in free space as $\mathbf{x}_{\mathrm{rand}}$ will be related to $R^{\mathrm{SMR}}(\mathbf{x})$ of that point. In other words, the location with higher $R^{\mathrm{SMR}}(\mathbf{x})$ will be more likely to be selected. After that, $\mathbf{x}_{\mathrm{rand}}$ attends to connect the nearest vertex $\mathbf{x}_{\mathrm{nearest}}$ found by the function Nearest($\mathcal{T}$, $\mathbf{x}_{\mathrm{rand}}$) as follows:

$$\mathbf{x}_{\mathrm{nearest}}\leftarrow\arg\min_{\mathbf{x}\in\mathcal{V}}\|\mathbf{x}_{\mathrm{rand}}-\mathbf{x}\| \tag{34}$$

and be adjusted to the new vertex $\mathbf{x}_{\mathrm{new}}$ by function Steer($\mathbf{x}_{\mathrm{nearest}}$, $\mathbf{x}_{\mathrm{rand}}$) [see Fig. 7(a)], i.e., minimizing the distance from $\mathbf{x}_{\mathrm{new}}$ to $\mathbf{x}_{\mathrm{rand}}$ and satisfying $\|\mathbf{x}_{\mathrm{nearest}}-\mathbf{x}_{\mathrm{new}}\|\leqslant\eta$ (η is a predefined maximum extension distance according to the map size) [31].

To update the edges related to the new vertex, i.e., $\mathbf{x}_{\mathrm{new}}$, the vertices $\mathbf{x}_{\mathrm{near}}$ located in a local hypersphere (i.e., χ_{near}) of a certain radius centered at $\mathbf{x}_{\mathrm{new}}$ will be first searched by function Near($\mathcal{T}$, $\mathbf{x}_{\mathrm{new}}$) [31] if an obstacle-free condition, i.e., ObstacleFree($\mathbf{x}_{\mathrm{nearest}}$, $\mathbf{x}_{\mathrm{new}}$), is satisfied. The function ObstacleFree($\mathbf{x}_{\mathrm{from}}$, $\mathbf{x}_{\mathrm{to}}$) will be satisfied if the line connecting $\mathbf{x}_{\mathrm{from}}$ and $\mathbf{x}_{\mathrm{to}}$ does not pass through any obstacle (i.e., any location in the line meets (19) requirement), as shown in Fig. 7(b). Then, two operations are implemented as follows.

1) *Find the Optimal Parent for $\mathbf{x}_{new}$ (see Lines 10–18 of Algorithm 1):* The cost of a certain path from $\mathbf{x}_{\mathrm{init}}$ to

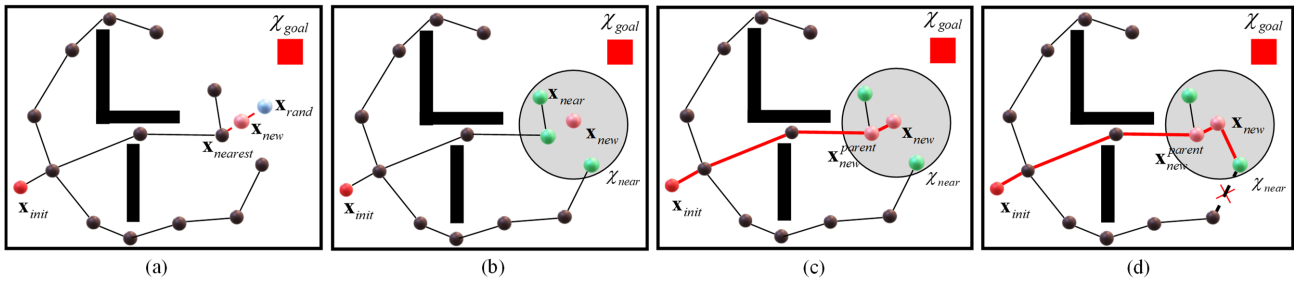


Fig. 7. Illustration of R2-RRT*. In each iteration of R2-RRT*, a new random vertex (denoted by pink) is first generated by collecting the newly sampled random point (denoted by light blue) to its nearest vertex in (a). Several near vertices (denoted by green) in (b) are then identified around the new random vertex, (c) chooses parent for the new vertex and adds an edge between them to the current tree, (d) rewires the near vertices within the local hypersphere around the generated new vertex. For instance, the dashed edge is rewired by the red edge between the pink new vertex and green near the vertex.

$\mathbf{x}_{\text{new}}$ under the assumption that $\mathbf{x}_{\text{near}}$ is considered as parent of $\mathbf{x}_{\text{new}}$ is calculated for all $\mathbf{x}_{\text{near}}$ in χ_{near} by $\text{RCost1}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{near}}, \mathbf{x}_{new})$ based on (21)–(23). The vertex with minimum cost will be selected from χ_{near} and regarded as parent (i.e., $\mathbf{x}_{\text{new}}^{\text{parent}}$) of the newly added $\mathbf{x}_{\text{new}}$ under the condition that $\text{ObstacleFree}(\mathbf{x}_{\text{near}}, \mathbf{x}_{\text{new}})$ is satisfied. After that, an edge from $\mathbf{x}_{\text{new}}^{\text{parent}}$ to $\mathbf{x}_{\text{new}}$ is added to the random tree [see Fig. 7(c)].

2) *Rewire the Edges of Tree Through $\mathbf{x}_{new}$ (see Lines 19–24 of Algorithm 1):* The other neighbor vertices in χ_{near} except for $\mathbf{x}_{\text{new}}^{\text{parent}}$ are rewired. For example, if the path cost from $\mathbf{x}_{\text{init}}$ to $\mathbf{x}_{\text{near}}$ with $\mathbf{x}_{\text{new}}$ assumed to be the parent of $\mathbf{x}_{\text{near}}$ is lower than the current path cost of $\mathbf{x}_{\text{near}}$ (i.e., $\text{RCost1}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{new}}, \mathbf{x}_{\text{near}}) < \text{RCost1}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{near}}^{\text{parent}}, \mathbf{x}_{\text{near}})$), then the parent of $\mathbf{x}_{\text{near}}$ will be replaced by $\mathbf{x}_{\text{new}}$, and the edge from $\mathbf{x}_{\text{near}}^{\text{parent}}$ to $\mathbf{x}_{\text{near}}$ will be replaced by the edge from $\mathbf{x}_{\text{new}}$ to $\mathbf{x}_{\text{near}}$, as shown in Fig. 7(d).

Note that $\text{RCost1}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}})$ returns the cost of a certain path from $\mathbf{x}_{\text{init}}$ to $\mathbf{x}_{\text{temp}}$, and $\mathbf{x}_{\text{temp}}^{\text{parent}}$ is regarded as parent vertex of $\mathbf{x}_{\text{temp}}$ while calculating the cost. These two operations in χ_{near} guarantee the local optimal vertex and edges of the random tree, which plays a vital role in the asymptotic optimality of RRT* [31].

2) R2-RRT-2 for MMR-Based Mission Planning Model:* Algorithm 2 presents the details of R2-RRT*-2. It starts with a similar initialization as that of R2-RRT*-1. Given a slope/soil map, a reduced work space is generated through $SMR(\text{map}, V_{\text{limit}}, [R])$. Moreover, in GetMobility(map), the natural variabilities of the slope/soil properties are first simulated by Gaussian random fields based on (7)–(12), resulting in many random realizations of the slope/soil map. As illustrated in Fig. 4, random realizations of the stochastic mobility map, i.e., $\mathcal{M}_{\text{re}}$, are then obtained through uncertainty propagation from environment parameters to mobility prediction.

After initialization, the remaining operations have two main differences between R2-RRT*-1 and R2-RRT*-2 due to the presence of $\mathcal{M}_{\text{re}}$. First, the way of calculating the robust cost of a certain path is different. In R2-RRT*-2, the $\text{RCost2}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}}, \mathcal{M}_{\text{re}})$ given in (29)–(31) [i.e., $C_{\text{robust}}^{\text{MMR}}(\psi)$], is employed, while R2-RRT*-1 uses $\text{RCost1}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}})$ given in (21)–(23) [i.e., $C_{\text{robust}}^{\text{SMR}}(\psi)$]. Second, the MMR constraint (i.e., $\text{MMR}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}}, \mathcal{M}_{\text{re}}) \geq [R]$) is embedded into R2-RRT*-2 to guarantee the mission reliability level (see Lines 14 and 22 in Algorithm 2). Since MMR decreases over a mission path, it is necessary to check whether the MMR of a certain path satisfies the MMR constraint in (28) in the two core operations of R2-RRT*-2, namely find the optimal parent for $\mathbf{x}_{\text{new}}$ and rewire the edges of tree through $\mathbf{x}_{\text{new}}$.

Algorithm 1 R2-RRT*-1

1 V_{limit}, $[R]$, map, $\mathbf{x}_{init}$, χ_{goal};
2 $\mathcal{V} \leftarrow \mathbf{x}_{init}$, $\mathcal{E} \leftarrow \emptyset$, $\mathcal{T} = (\mathcal{V}, \mathcal{E})$;
3 χ_{free}, $R^{SMR}(\mathbf{x}) \leftarrow SMR(\text{map}, V_{limit}, [R])$;
4 **for** $i \leftarrow 1$ **to** N_{iters} **do**
5 $\quad \mathbf{x}_{rand} \leftarrow PSample(\chi_{free}, R^{SMR}(\mathbf{x}))$;
6 $\quad \mathbf{x}_{nearest} \leftarrow Nearest(\mathcal{T}, \mathbf{x}_{rand})$;
7 $\quad \mathbf{x}_{new} \leftarrow Steer(\mathbf{x}_{nearest}, \mathbf{x}_{rand})$;
8 $\quad$ **if** $ObstacleFree(\mathbf{x}_{nearest}, \mathbf{x}_{new})$ **then**
9 $\quad\quad \chi_{near} \leftarrow Near(\mathcal{T}, \mathbf{x}_{new})$;
10 $\quad\quad \mathbf{x}_{new}^{parent} = \mathbf{x}_{nearest}$;
11 $\quad\quad c_{min} \leftarrow RCost1(\mathbf{x}_{init}, \mathbf{x}_{new}^{parent}, \mathbf{x}_{new})$;
12 $\quad\quad$ **foreach** $\mathbf{x}_{near} \in \chi_{near}$ **do**
13 $\quad\quad\quad$ **if** $ObstacleFree(\mathbf{x}_{near}, \mathbf{x}_{new})$ & $RCost1(\mathbf{x}_{init}, \mathbf{x}_{near}, \mathbf{x}_{new}) < c_{min}$ **then**
14 $\quad\quad\quad\quad \mathbf{x}_{new}^{parent} = \mathbf{x}_{near}$;
15 $\quad\quad\quad\quad c_{min} = RCost1(\mathbf{x}_{init}, \mathbf{x}_{near}, \mathbf{x}_{new})$;
16 $\quad\quad\quad$ **end**
17 $\quad\quad$ **end**
18 $\quad\quad \mathcal{V} \leftarrow \mathcal{V} \cup \mathbf{x}_{new}$, $\mathcal{E} = \mathcal{E} \cup \{(\mathbf{x}_{new}^{parent}, \mathbf{x}_{new})\}$;
19 $\quad\quad$ **foreach** $\mathbf{x}_{near} \in \chi_{near}$ **do**
20 $\quad\quad\quad c_{near} \leftarrow RCost1(\mathbf{x}_{init}, \mathbf{x}_{near}^{parent}, \mathbf{x}_{near})$;
21 $\quad\quad\quad$ **if** $ObstacleFree(\mathbf{x}_{new}, \mathbf{x}_{near})$ & $RCost1(\mathbf{x}_{init}, \mathbf{x}_{new}, \mathbf{x}_{near}) < c_{near}$ **then**
22 $\quad\quad\quad\quad \mathcal{E} \leftarrow (\mathcal{E} \backslash \{(\mathbf{x}_{near}^{parent}, \mathbf{x}_{near})\}) \cup \{(\mathbf{x}_{new}, \mathbf{x}_{near})\}$;
23 $\quad\quad\quad$ **end**
24 $\quad\quad$ **end**
25 $\quad$ **end**
26 **end**
27 **return** $\mathcal{T} = (\mathcal{V}, \mathcal{E})$;

Algorithm 2 R2-RRT*-2

V_{limit}, $[R]$, map, $\mathbf{x}_{init}$, χ_{goal};
$\mathcal{V} \leftarrow \mathbf{x}_{init}$, $\mathcal{E} \leftarrow \emptyset$, $\mathcal{T} = (\mathcal{V}, \mathcal{E})$;
χ_{free}, $R^{SMR}(\mathbf{x}) \leftarrow SMR(\text{map}, V_{limit}, [R])$;
$\mathcal{M}_{re} \leftarrow GetMobility(\text{map})$;
for $i \leftarrow 1$ **to** N_{iters} **do**
  $\mathbf{x}_{rand} \leftarrow PSample(\chi_{free}, R^{SMR}(\mathbf{x}))$;
  $\mathbf{x}_{nearest} \leftarrow Nearest(\mathcal{T}, \mathbf{x}_{rand})$;
  $\mathbf{x}_{new} \leftarrow Steer(\mathbf{x}_{nearest}, \mathbf{x}_{rand})$;
  if $ObstacleFree(\mathbf{x}_{nearest}, \mathbf{x}_{new})$ **then**
    $\chi_{near} \leftarrow Near(\mathcal{T}, \mathbf{x}_{new})$;
    $\mathbf{x}_{new}^{parent} = \mathbf{x}_{nearest}$;
    $c_{min} \leftarrow RCost2(\mathbf{x}_{init}, \mathbf{x}_{nearest}, \mathbf{x}_{new}, \mathcal{M}_{re})$;
    foreach $\mathbf{x}_{near} \in \chi_{near}$ **do**
      if $ObstacleFree(\mathbf{x}_{near}, \mathbf{x}_{new})$ & $RCost2(\mathbf{x}_{init}, \mathbf{x}_{near}, \mathbf{x}_{new}, \mathcal{M}_{re}) < c_{min}$ & $MMR(\mathbf{x}_{init}, \mathbf{x}_{near}, \mathbf{x}_{new}, \mathcal{M}_{re}) \geq [R]$ **then**
        $\mathbf{x}_{new}^{parent} = \mathbf{x}_{near}$;
        $c_{min} \leftarrow RCost2(\mathbf{x}_{init}, \mathbf{x}_{near}, \mathbf{x}_{new}, \mathcal{M}_{re})$;
      end
    end
    $\mathcal{V} \leftarrow \mathcal{V} \cup \mathbf{x}_{new}$, $\mathcal{E} = \mathcal{E} \cup \{(\mathbf{x}_{new}^{parent}, \mathbf{x}_{new})\}$;
    foreach $\mathbf{x}_{near} \in \chi_{near}$ **do**
      $c_{near} \leftarrow RCost2(\mathbf{x}_{init}, \mathbf{x}_{near}^{parent}, \mathbf{x}_{near}, \mathcal{M}_{re})$;
      if $ObstacleFree(\mathbf{x}_{new}, \mathbf{x}_{near})$ & $RCost2(\mathbf{x}_{init}, \boldsymbol{x}_{new}, \boldsymbol{x}_{near}, \mathcal{M}_{re}) < c_{near}$ & $MMR(\mathbf{x}_{init}, \mathbf{x}_{new}, \mathbf{x}_{near}, \mathcal{M}_{re}) \geq [R]$ **then**
        $\mathcal{E} \leftarrow (\mathcal{E} \backslash \{(\mathbf{x}_{near}^{parent}, \mathbf{x}_{near})\}) \cup \{(\mathbf{x}_{new}, \mathbf{x}_{near})\}$;
      end
    end
  end
end
return $\mathcal{T} = (\mathcal{V}, \mathcal{E})$;

If $\text{MMR}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}}, \mathcal{M}_{\text{re}}) \geq [R]$, vertex $\mathbf{x}_{\text{temp}}^{\text{parent}}$ is considered as the parent of $\mathbf{x}_{\text{temp}}$ and is used to define a potential path $\mathbf{x}_{\text{init}} \rightarrow \mathbf{x}_{\text{temp}}^{\text{parent}} \rightarrow \mathbf{x}_{\text{temp}}$. Based on the definition in (32), MMR of this path is then calculated by

$$\begin{aligned} & R^{\text{MMR}}\left(\psi\left(\mathbf{x}_{\text{init}} \rightarrow \mathbf{x}_{\text{temp}}^{\text{parent}} \rightarrow \mathbf{x}_{\text{temp}}\right)\right) \\ & = \Pr\left\{V(\psi(t_i)) \geqslant V_{\text{limit}} \quad \forall t_i \in [t_0, t_{\text{temp}}]\right\} \\ & = \Pr\left\{V_{\min} = \min_{t_i = t_0, \ldots, t_{\text{temp}}^{\text{parent}}, \ldots, t_{\text{temp}}} \{V(\psi(t_i))\} \geqslant V_{\text{limit}}\right\} \end{aligned} \tag{35}$$

where $\psi(t_0) = \mathbf{x}_{\text{init}}$, $\psi(t_{\text{temp}}^{\text{parent}}) = \mathbf{x}_{\text{temp}}^{\text{parent}}$, and $\psi(t_{\text{temp}}) = \mathbf{x}_{\text{temp}}$.

Meanwhile, the MMR of path $\mathbf{x}_{\text{init}} \rightarrow \mathbf{x}_{\text{temp}}^{\text{parent}}$ is calculated by

$$\begin{aligned} & R^{\text{MMR}}\left(\psi\left(\mathbf{x}_{\text{init}} \rightarrow \mathbf{x}_{\text{temp}}^{\text{parent}}\right)\right) \\ & = \Pr\left\{V_{\min} = \min_{t_i = t_0, \ldots, t_{\text{temp}}^{\text{parent}}} \{V(\psi(t_i))\} \geqslant V_{\text{limit}}\right\}. \end{aligned} \tag{36}$$

By comparing (35) with (36), it is observed that the minimum speed (i.e., $V_{\min}$) on ψ will be updated if the vehicle speed over $[\psi(t_{\text{temp}}^{\text{parent}}), \psi(t_{\text{temp}})]$ of ψ is lower. Therefore, MMR is prone to decrease when adding new edges $\mathbf{x}_{\text{temp}}^{\text{parent}} \rightarrow \mathbf{x}_{\text{temp}}$ to the path $\mathbf{x}_{\text{init}} \rightarrow \mathbf{x}_{\text{temp}}^{\text{parent}}$. If MMR constraint is satisfied, Algorithm 2 will return that the potential path $\mathbf{x}_{\text{init}} \rightarrow \mathbf{x}_{\text{temp}}^{\text{parent}} \rightarrow \mathbf{x}_{\text{temp}}$ is feasible.

3) Completeness, Optimality, and Computational Complexity: In R2-RRT* algorithms, $\text{PSample}(\chi_{\text{free}}, R^{\text{SMR}}(\mathbf{x}))$ can sample any potential location in obstacle-free space according to the mobility reliability. Similar to RRT*, the tree of R2-RRT* tries to connect all samples (i.e., vertices) from $\mathbf{x}_{\text{init}}$ toward the goal region χ_{goal}. If a feasible solution exists, the probability that R2-RRT* finds the feasible path approaches one as the number of iterations approaches infinity. The following probabilistic completeness will be naturally guaranteed similar to RRT* [31]:

$$\lim_{n \rightarrow \infty} \Pr(\mathcal{V} \cap \chi_{\text{goal}} \neq \emptyset) = 1 \tag{37}$$

where n denotes the number of iterations.

R2-RRT* employs SMR and MMR constraints to reduce the free space and calculates the robust mission cost instead of deterministic cost. Reduction of free space only leads to a stricter condition that a feasible path exists. Changing the mission planning goal from the minimization of deterministic cost to the minimization of robust cost only affects the selection of the final solution when multiple feasible paths are found in the free space. Therefore, R2-RRT* inherits the asymptotic optimality characteristic of RRT* under the following condition according to [35, Theorem 38]:

$$\gamma > (2(1 + 1/d))^{1/d} \left(\frac{\mu(\chi_{\text{free}})}{\xi_d}\right)^{1/d} \tag{38}$$

where γ is related to the search radius in $\text{Near}(\mathcal{T}, \mathbf{x}_{\text{new}})$, d denotes the number of dimensionality of work space, $\mu(\chi_{\text{free}})$ denotes the Lebesgue measure of obstacle-free space, and ξ_d denotes the volume of the unit ball in the d-dimensional Euclidean space. This means the probability that the final solution of R2-RRT* will have minimum robust cost approaches one as the number of iterations approaches infinity.

According to the computational complexity analysis in [31], we have time complexity of RRT* given by $O(n \log n \cdot \log^d m)$, where n is the number of iterations and m is the number of obstacles. Comparing R2-RRT*-1 with RRT*, the time complexity of the newly added $SMR(\text{map}, V_{\text{limit}}, [R])$, $\text{PSample}(\chi_{\text{free}}, R^{\text{SMR}}(\mathbf{x}))$, and $\text{RCost1}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}})$ is, respectively, $O(N_{\text{MCS}})$, $O(1)$, and $O(1)$. Since N_{MCS} is usually smaller than $n \log n \cdot \log^d m$, the time complexity of R2-RRT*-1 is, therefore, also $O(n \log n \cdot \log^d m)$. For R2-RRT*-2, the time complexity of the new functions $\text{RCost2}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}}, \mathcal{M}_{\text{re}})$ and $\text{MMR}(\mathbf{x}_{\text{init}}, \mathbf{x}_{\text{temp}}^{\text{parent}}, \mathbf{x}_{\text{temp}}, \mathcal{M}_{\text{re}})$ are all $O(N_{\text{MCS}})$. Therefore, the time complexity of R2-RRT*-2 is $O(\max(O(n \log n \cdot \log^d m), O(n \log n \cdot N_{\text{MCS}})))$.

B. Reliability-Based Path Smoothing Using B-Spline Curve

While R2-RRT* algorithms can find feasible paths satisfying mobility reliability requirements, the identified path needs to be further optimized due to the suboptimality of RRT* [8].

Algorithm 3 Reliability-Based Path Smoothing

1 $\psi^*_{R2-RRT^*}$, V_{limit}, $[R]$, $\mathbf{x}_{init}$, χ_{goal}, χ_{free}, $\chi_{obstacle}$;
2 $P_{control} \leftarrow RemoveVertex(\psi^*_{R2-RRT^*}, \chi_{obstacle})$;
3 $\psi^*_{smooth} \leftarrow Bspline(P_{control}, l)$;
4 **while** $R^{SMR}(\mathbf{x}) < [R], \exists \mathbf{x} \in \psi^*_{smooth}$ *(R2-RRT*-1) or* $R^{MMR}(\psi^*_{smooth}) < [R]$ *(R2-RRT*-2)* **do**
5 $\quad \mathbf{x}^{\max}_{R2-RRT^*} \leftarrow MaxDiff()$;
6 $\quad P_{control} \leftarrow InsertVertex(P_{control}, \mathbf{x}^{\max}_{R2-RRT^*})$;
7 $\quad P_{control} \leftarrow InsertMidpoint(P_{control})$;
8 $\quad \psi^*_{smooth} \leftarrow Bspline(P_{control}, l)$;
9 **end**
10 **return** ψ^*_{smooth};

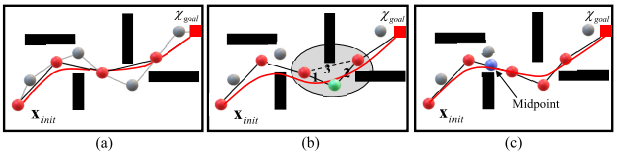


Fig. 8. Illustration of reliability-based smoothing. (a) Obtains the initial control points (denoted by red) by removing the redundant vertices (denoted by gray) of R2-RRT* (Line 2 in Algorithm 3). (b) adds more control points from the vertices of R2-RRT* to guarantee the reliability level of the smoothed path (Lines 5 and 6). For instance, the path, including segments 1 and 2, is safer than the path, including segment 3 (dashed line). Thus, the green vertex is added to the control point set. (c) inserts midpoints (denoted by blue) for some important control points to guarantee the reliability level of the smoothed path (Line 8).

In this section, the B-spline curve is employed for path smoothing, and a reliability-based path smoothing algorithm is proposed to consider not only the obstacle-free criterion but also the SMR or MMR criterion during smoothing.

In Algorithm 3, we start with an initialization by removing redundant vertices through function RemoveVertex($\psi^*_{\text{R2-RRT}^*}, \chi_{\text{obstacle}}$). In RemoveVertex(), letting all the vertices in $\psi^*_{\text{R2-RRT}^*}$ from start to goal position be denoted by $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_\psi}\}$, and the initial control point set, P_{control}, be an empty set, we first add $\mathbf{x}_j$ to control point set P_{control} by setting $j = N_\psi$. After that, for each $i \in [1, j-1]$, we check whether the line between $\mathbf{x}_i$ and $\mathbf{x}_j$ passes through the obstacle space χ_{obstacle} and stop on the first $\mathbf{x}_i$ without collision. The identified vertex $\mathbf{x}_i$ is subsequently added to P_{control}. Following that, we set $j = i$ and repeat the checking process until $\mathbf{x}_1$, i.e., the start position, is added to P_{control} [35]. With the identified control points shown in Fig. 8(a), a smoothed path can be obtained by a l-degree B-spline parametric curve Bspline(P_{control}, l).

Although removing redundant vertices is efficient, the smoothed path may not meet the SMR or MMR requirement. In Algorithm 3, if the reliability constraint is not satisfied, we will let ψ^*_{smooth} have the same number of segments as $\psi^*_{\text{R2 - RRT}^*}$ and find the segment (i.e., vertex $\mathbf{x}^{\max}_{\text{R2 - RRT}^*}$) where the difference of SMR between ψ^*_{smooth} and $\psi^*_{\text{R2 - RRT}^*}$ is the largest by a function called MaxDiff($R^{\text{SMR}}(\mathbf{x}_{\text{smooth}})$, $R^{\text{SMR}}(\mathbf{x}_{\text{R2 - RRT}^*})$), which is defined as

$$\mathbf{x}^{\max}_{\text{R2-RRT}^*} \in \mathbf{x}_{\text{R2-RRT}^*} \leftarrow \underset{i=1,2,\ldots,N_\psi}{\arg\max} \left(R^{\text{SMR}}(\mathbf{x}^i_{\text{smooth}}), R^{\text{SMR}}(\mathbf{x}^i_{\text{R2 - RRT}^*})\right) \quad (39)$$

where $\mathbf{x}^i_{\text{smooth}} \in \mathbf{x}_{\text{smooth}}$, $\mathbf{x}^i_{\text{R2 - RRT}^*} \in \mathbf{x}_{\text{R2 - RRT}^*}$, $i = 1, 2, \ldots, N_\psi$, and $R^{\text{SMR}}(\mathbf{x}^i_{\text{smooth}})$ or $R^{\text{SMR}}(\mathbf{x}^i_{\text{R2-RRT}^*})$ is calculated by (18). By sequentially inserting the vertex $\mathbf{x}^{\max}_{\text{R2 - RRT}^*}$ found by MaxDiff() into P_{control} according to the original order of vertices in $\psi^*_{\text{R2 - RRT}^*}$, the reliability level of newly smoothed path will gradually reach to that of the path found by R2-RRT* and, thus, satisfy the SMR or MMR requirement. As illustrated in Fig. 8(b), the green vertex found by MaxDiff() is inserted into P_{control} since the path with segments "1 + 2" (denoted by solid black lines) is safer than the path with segment "3" (denoted by dashed lines).

Moreover, P_{control} is further updated by inserting midpoint between two consecutive control points to avoid collision, such as those shown in Fig. 8(b) [36], before the path is smoothed by Bspline(P_{control}, l). In function InsertMidpoint(P_{control}), midpoints will be inserted at the control points only if the following criterion is satisfied:

$$\min(\|\mathbf{x}_{\text{control}} - \mathbf{x}_{\text{obstacle}}\|) \leqslant \xi, \mathbf{x}_{\text{obstacle}} \in \chi_{\text{obstacle}} \quad (40)$$

where ξ represents a certain distance related to the size of a single obstacle (i.e., resolution of target map). As illustrated in Fig. 8(c), only one blue node is inserted into P_{control}, resulting in a collision-free smoothed path (red curve) comparing with that in Fig. 8(b).

Through the reliability-based mission smoothing, the suboptimality of R2-RRT* is addressed. A smooth, continuous, and small-curvature path will avoid suddenly slowing down at sharp turns and improve the success rate of accomplishing a mission for the off-road AGVs. It is worth noting that, if the resolution of the target map is low, path smoothing may not be necessary, and discrepancies between the smoothed and nonsmoothed paths will be eliminated.

V. Case Study

For demonstration, a 50 m × 50 m target map is taken from a terrain map of interest obtained from ARCGIS/ENVI database [37] with data provided in raster format. As shown in Fig. 9, the slope of the target map can be calculated by ARCGIS using the elevation raster, and the soil types of the target map can be identified with the property data provided by the US Geological Survey database [38], [39]. The size of the cells (i.e., squares in Fig. 9), which affects the computational efforts, can be defined according to the resolution of the target map, size of map, or computational limit. In this work, the resolution of the target map is set to 1 m, and there are 2500 cells in Fig. 9. The off-road vehicle mobility model is assumed to be a model as follows:

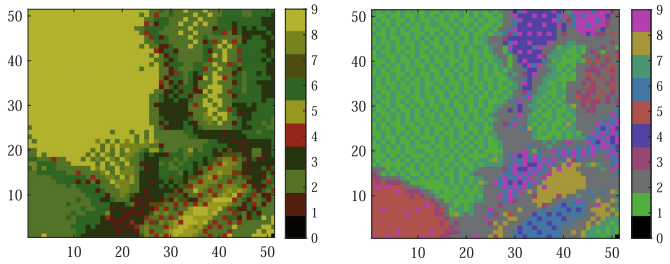
$$V(\mathbf{x}) = G(\mathbf{W}, \mathbf{Y}(\mathbf{x})) = 0.85e^{Y_1(\mathbf{x})/15 - Y_3(\mathbf{x})} + (Y_2(\mathbf{x})/5 - Y_4(\mathbf{x}))^2 + 0.7Y_2(\mathbf{x})Y_3(\mathbf{x})Y_4(\mathbf{x}) \quad (41)$$

where $\mathbf{Y}(\mathbf{x}) = [Y_1(\mathbf{x}), Y_2(\mathbf{x}), Y_3(\mathbf{x}), Y_4(\mathbf{x})]$ are, respectively, the slope, cohesive strength, friction coefficient, and bulk density. All these environment parameters are assumed to be

TABLE I
STATISTICAL PARAMETERS OF SLOPE AND SOIL PROPERTIES

Slope/Soil ID	slope		Soil					
			Cohesive strength		Friction coefficient		Bulk density	
	μ/σ	β_1/β_2	μ/σ	β_1/β_2	μ/σ	β_1/β_2	μ/σ	β_1/β_2
0	Water		Water					
1	5/0.2	3.3/3.0	0.2/0.01	40/39	0.01/0.001	33/32	0.05/0.001	27/28
2	10/0.5	3.4/3.6	1/0.1	42/43	0.8/0.05	28/33	1.2/0.1	23/27
3	12/0.5	4.3/2.2	5/0.4	41/41	0.7/0.02	28/27	2.1/0.1	20/25
4	14/1	3.6/4.4	6/0.2	41/39	0.76/0.02	31/29	2.3/0.1	26/24
5	16/2	3.5/5.1	2/0.1	40/42	0.56/0.01	33/32	1.6/0.04	27/21
6	18/2	2.8/5.0	11/0.2	38/37	0.8/0.02	27/27	2.1/0.05	26/28
7	22/2	3.2/2.2	5/0.02	39/42	0.7/0.03	28/30	1.2/0.02	24/28
8	24/1	4.7/4.0	4/0.1	37/39	0.45/0.005	29/28	1.45/0.01	24/27
9	28/1	4.5/6.7	8/0.4	42/35	0.78/0.03	32/32	2.35/0.05	26/24

μ is mean, σ is standard deviation, β_1 and β_2 are the correlation length related parameters in coordinate 1 and coordinate 2 respectively.



(a) Slope map (b) Soil map

Fig. 9. (a) Slope and (b) soil maps used in case study.

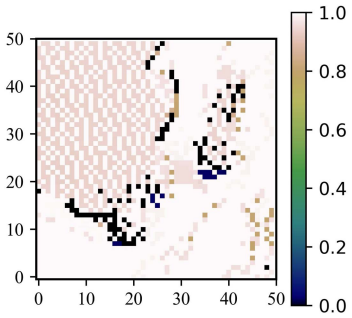


Fig. 10. Probabilistic map of R2-RRT*.

Gaussian random fields, and Table I shows the statistical information of these Gaussian random fields. With these assumed parameters, the output of the assumed vehicle mobility model $V(\mathbf{x})$ is in the range of [0, 20] m/s. Note that the distributions of slope and soil properties, and vehicle mobility model are assumed for demonstration and repeatability purposes. The vehicle mobility model given in (41) can be replaced by any terramechanics model or simulation models, as illustrated in Fig. 1, such as the Nevada automotive test center wheeled vehicle platform developed by Advanced Science and Automation Corporation [40]. The immobility event in this case study is defined as the maximum attainable speed being less than 2 m/s (i.e., $V_{\text{limit}} = 2$ m/s).

Fig. 10 shows the probabilistic map obtained by following the procedure of SMR analysis (see Fig. 5 and Section III-B), which quantifies the probability of maintaining vehicle mobility at each location. Based on the probabilistic map, different target reliability levels [R] will yield different work spaces for path planner [see Figs. 11(b) and (c) and 15(a) and (b)]. Two scenarios of [R] are studied in this article, namely, 90% and 95%. Table II lists the simulation parameters of RRT* and R2-RRT*, and the results of RRT* are based on the PythonRobotics project [41].

TABLE II
MAIN SIMULATION PARAMETER SETTINGS

Parameter		Parameter	
Map size	50×50	Iteration	5000
Resolution	1	Expand size (η)	4
$\mathbf{x}_{init}$	[3, 15]	$\mathbf{x}_{goal}$	[32, 40]
Goal sameple rate (λ)	10%	B-spline degree (l)	2

A. Scenario 1: 90% Minimum Target Reliability

Fig. 11 shows three different planning results of RRT* and R2-RRT*, where the green segments represent the edges of the random tree, and the blackline and red line represent the paths before and after mission smoothing, respectively. The work space of RRT* is obtained by using the mean values of slope/soil properties to generate the deterministic mobility map, while the work space of R2-RRT*-1/2 is obtained by dividing the probabilistic map in Fig. 10 into an obstacle and free space according to the required reliability level [R]. We can observe that the work spaces of R2-RRT*-1/2 are the same and have more obstacles than that of RRT*

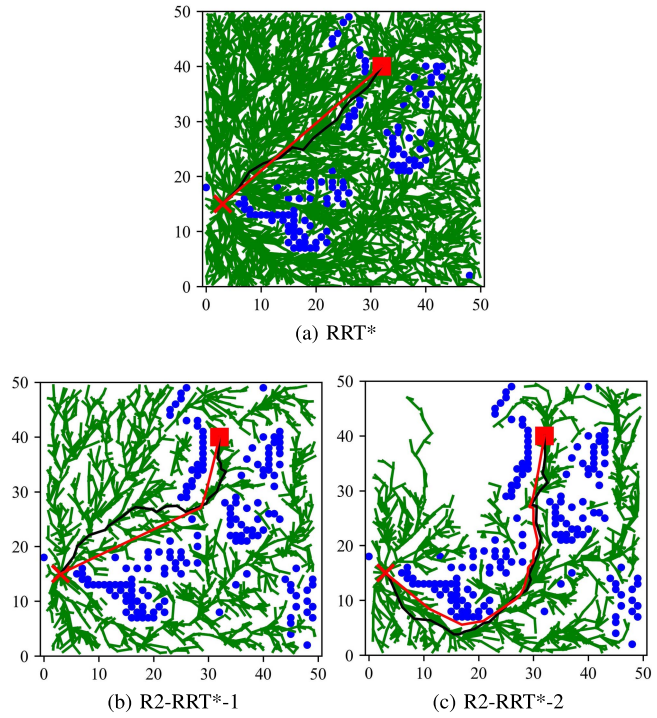


Fig. 11. Random trees and paths for Scenario 1 (blue points are obstacles). (a) RRT*. (b) R2-RRT*-1. (c) R2-RRT*-2.

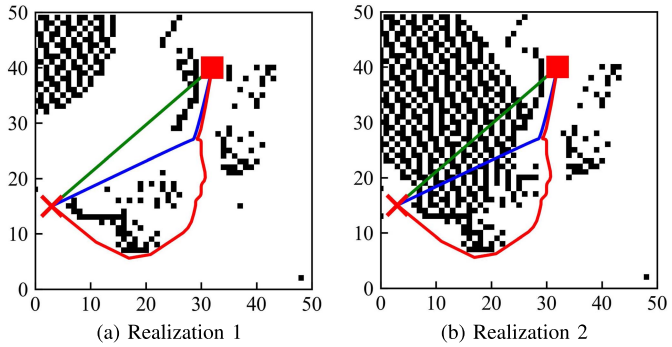


Fig. 12. Smoothed paths in two different realizations of maps for Scenario 1. (green: RRT*; blue: R2-RRT*-1; and red: R2-RRT*-2). (a) Realization 1. (b) Realization 2.

since both of them employ an SMR constraint to reduce the free space. As a result, the path found by RRT* is not feasible in the reduced free space of R2-RRT*-1/2. Moreover, the path found by R2-RRT*-2 is different from that of R2-RRT*-1 due to the MMR constraint in R2-RRT*-2. Fig. 12 shows the smoothed paths in two realizations of the stochastic mobility map obtained through MMR analysis (as illustrated in Fig. 6 and Section III-C). It indicates that the path identified by RRT* fails in both realizations. The path identified by R2-RRT*-1 fails in the second realization, which is manifested as the path passes through the obstacles. This is attributed to the fact that R2-RRT*-1 ignores the spatial dependence of uncertainty sources even if the uncertainty in vehicle mobility is considered. It also explains why the MMR of R2-RRT*-1 is lower than that of the path found by R2-RRT*-2 (see Table III).

TABLE III
RESULTS OF DIFFERENT METHODS FOR SCENARIO 1

Scenario 1	Method		
	RRT*	R2-RRT*-1	R2-RRT*-2
Mean cost (no smoothing) (s)	9.71	9.85	11.29
Mean cost (smoothing) (s)	10.30	11.95	13.41
MMR (no smoothing)	58.57%	83.99%	**92.39%**
MMR (smoothing)	69.56%	85.16%	**91.71%**

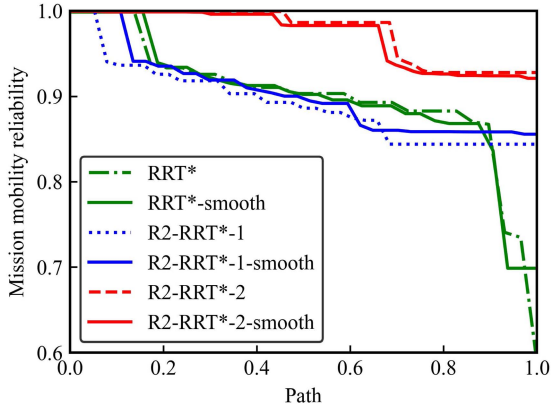


Fig. 13. MMR for Scenario 1.

Table III compares the mean cost and MMR of different paths calculated based on random realizations of the stochastic vehicle mobility map. The mean cost increases from RRT* to R2-RRT*-1 and to R2-RRT*-2, whereas the MMR sees a significant decrease from R2-RRT*-2 to R2-RRT*-1 and RRT*. The path found by RRT* in the deterministic mobility map will get the vehicle stuck on the path with a probability of about 40% (i.e., MMR equals 58.57%). Although R2-RRT*-1 finds a more reliable path than RRT* by accounting for SMR during path planning, the MMR is still lower than the required reliability level, i.e., 90%. Only R2-RRT*-2 can accurately meet the mobility reliability requirement. The results show that mission reliability is gradually increased by adding the SMR and MMR constraints, and MMR-based mission planners can more effectively guarantee the mobility reliability of off-road AGVs than RRT* and SMR-based mission planners.

Figs. 13 and 14 depict more features of MMR. As shown in Fig. 13, MMR gradually decreases as the accumulation of collision risk, which validates what we have explained in Section III-C (i.e., the further the vehicle goes, the lower the MMR is). Following that, Fig. 14 shows the distribution of the minimum vehicle mobility (i.e., maximum attainable speed) over the whole mission obtained from MCS, as illustrated in Fig. 4. The MMR calculated by (25) is equal to the probability that the minimum vehicle speed is greater than the speed threshold. These two figures show that R2-RRT*-2 well satisfies the mobility reliability requirement, which further validates the efficacy of the proposed reliability-based mission planning.

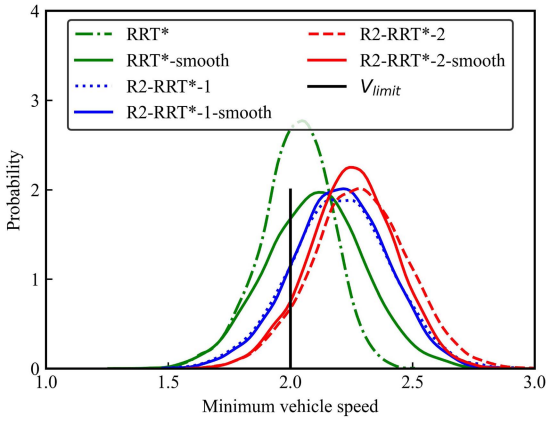


Fig. 14. Extreme speed distributions (i.e., minimum maximum attainable speed) obtained from MCS for Scenario 1.

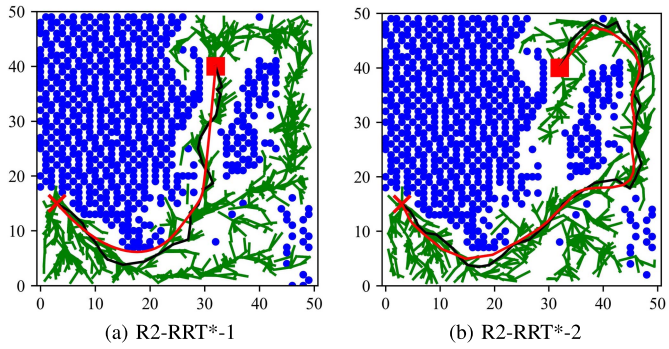


Fig. 15. Random trees and paths for Scenario 2. (a) R2-RRT*-1. (b) R2-RRT*-2.

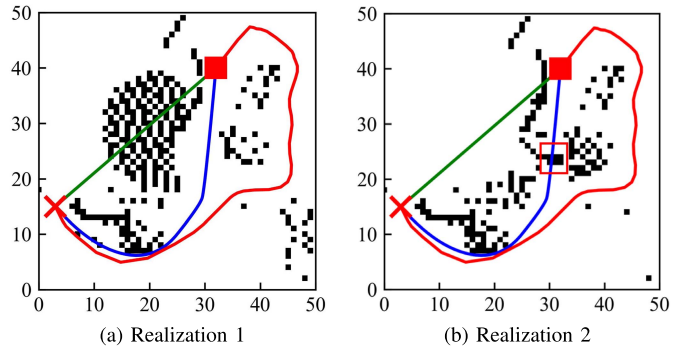


Fig. 16. Smoothed paths in two different realizations of maps for Scenario 1 (green: RRT*; blue: R2-RRT*-1; and red: R2-RRT*-2). (a) Realization 1. (b) Realization 2.

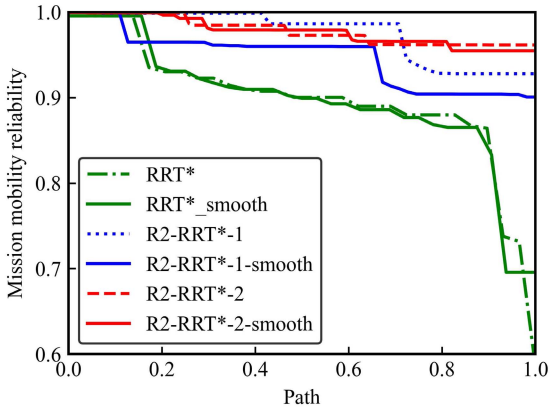


Fig. 17. Mission mobility reliabilities for Scenario 2.

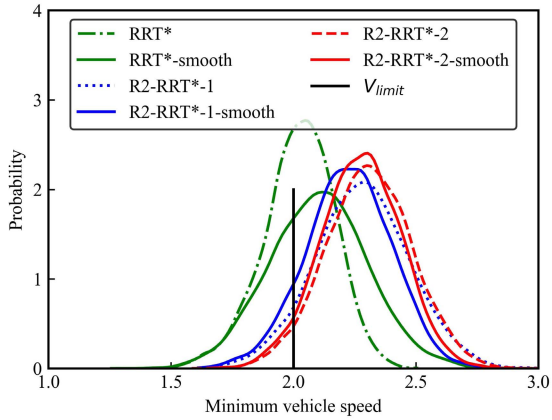


Fig. 18. Extreme speed distributions (i.e., minimum maximum attainable speed) obtained from MCS for Scenario 2.

TABLE IV
RESULTS OF DIFFERENT METHODS FOR SCENARIO 2

Scenario 2	Method		
	RRT*	R2-RRT*-1	R2-RRT*-2
Cost (No smoothing) (s)	9.71	11.35	15.28
Cost (Smoothing) (s)	10.30	13.47	16.17
MMR(No smoothing)	58.57%	92.39%	**95.76%**
MMR(Smoothing)	69.56%	89.67%	**95.08%**

B. Scenario 2: 95% Minimum Target Reliability

In this scenario, the deterministic result of RRT* is the same as that in Scenario 1, as shown in Fig. 11(a). In Fig. 15, the minimum target reliability $[R]$ is increased from 90% to 95% for R2-RRT*-1/2, further reducing the free space compared to RRT*. Results of R2-RRT*-1 and R2-RRT*-2 are different from each other and different from those of scenario 1. Fig. 16 shows smoothed paths in two realizations of the stochastic mobility map. Similar to Scenario 1, the path of RRT* fails in both realizations since RRT* cannot satisfy the SMR constraint. Although both R2-RRT*-1 and R2-RRT*-2 can meet the SMR constraint by finding feasible paths in free space, as shown in Fig. 15, the MMR of R2-RRT*-1 is lower than the required target level, as depicted in Table IV. Therefore, we can obtain some realizations, such as those in Fig. 15(b) where the path identified by R2-RRT*-1 will result in the vehicle getting stuck in the narrow region [i.e., red rectangle in Fig. 16(b)]. Different decisions of path planners result in differences in the mean cost and MMR in Table IV, and only R2-RRT*-2 satisfies the MMR requirement. Figs. 17 and 18, respectively, present the iteration history of MMR and extreme speed distribution over the whole mission obtained

from MCS. Similar conclusions can be obtained as that from Scenario 1.

VI. Conclusion

In this article, we proposed two reliability-based robust mission planning models and algorithms for off-road AGVs under uncertain terrain environment. In the mission planning models, the SMR analysis that focuses on the point reliability of each location was implemented to obtain the probabilistic map and reduce the search space. After that, the MMR analysis was employed to obtain realizations of stochastic mobility maps and to quantify the spatial-dependent mission reliability. To find a feasible path under the proposed planning models, the incrementally sampling RRT* algorithm was extended to R2-RRT*-1 and R2-RRT*-2 algorithms, followed by a reliability-based mission smoothing algorithm using the B-spline curve. The efficacy of proposed R2-RRT* algorithms was validated through a case study, including two scenarios that have two different MMR requirements.

In summary, this article embeds SMR and MMR analyses into mission planning for the first time. It provides new insights into this field for mission planning with the consideration of mobility reliability. The proposed models and algorithms are not limited to RRT*. They can also be extended to other mission planners introduced in Section I. The proposed models and algorithms can be applied to online mission planning by integrating the reliability-based mission planning algorithms with dynamic updating approaches from the following perspectives.

1) In the mobility-analysis phase, the observed online data (e.g., vehicle speed and environment information) can be used to update vehicle mobility reliability through uncertainty reduction of the uncertainty sources. For instance, the uncertainty in the terrain parameters can be dynamically updated during the mission using Bayesian methods [17]. The discrepancy between the mobility prediction obtained from simulation and the underlying true mobility can be reduced through mobility model correction using the observed vehicle speed data [42].
2) In the mission-planning phase, the updated mobility analysis can be implemented repeatedly for mission replanning considering the kinodynamic constraints and dynamic obstacles [8], [43].

Appendix
Karhunen–Loève Expansion

The KL expansion [32] represents a Gaussian/non-Gaussian random field as follows:

$$Y(\mathbf{x}) = \mu_Y(\mathbf{x}) + \sigma_Y(\mathbf{x}) \sum_{h=1}^{N_{KL}} \sqrt{\lambda_h}\xi_h \eta_h(\mathbf{x}) \tag{42}$$

where $\mu_Y(\mathbf{x})$ and $\sigma_Y(\mathbf{x})$ are, respectively, the mean and standard deviation of $Y(\mathbf{x})$, $\xi_h,\ h = 1, 2, \ldots, N_{KL}$ are standard normal random variables, N_{KL} is the number of expansion terms, and λ_h and $\eta_h(\mathbf{x})$ are the eigenvalues and eigenvectors of a correlation matrix with $(i,\ j)$, element given by

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = \exp\left[-\beta_1(x_{1i} - x_{1j})^2 - \beta_2(x_{2i} - x_{2j})^2\right] \quad \forall i,\ j = 1, 2, \ldots, N_{\text{discrete}} \tag{43}$$

where $\mathbf{x}_i = [x_{1i}, x_{2i}]$ are spatial coordinates at location i, β_1 and β_2 are the correlation length parameters, and N_{discrete} is the number of discrete spatial points. In this article, N_{discrete} equals to the size of slope/soil ID matrix in each coordinate.

Acknowledgment

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited. OPSEC 4349.

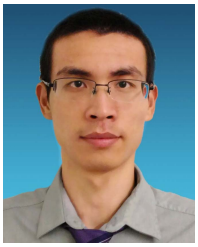
References

[1] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous UAV guidance," *J. Intell. Robot. Syst.*, vol. 57, no. 1, p. 65, Nov. 2009.
[2] D. Gonzalez, J. Perez, V. Milanes, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1135–1145, Apr. 2016.
[3] E. Petrick, Z. Janosi, and P. Haley, "The use of the NATO reference mobility model in military vehicle procurement," SAE Tech. Paper 810373, 1981.
[4] M. McCullough, P. Jayakumar, J. Dasch, and D. Gorsich, "The next generation NATO reference mobility model development," *J. Terramech.*, vol. 73, pp. 49–60, Oct. 2017.
[5] 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," *J. Terramech.*, vol. 72, pp. 39–54, Aug. 2017.
[6] R. Serban, D. Negrut, A. Recuero, and P. Jayakumar, "An integrated framework for high-performance, high-fidelity simulation of ground vehicle-tyre-terrain interaction," *Int. J. Vehicle Perform.*, vol. 5, no. 3, pp. 233–259, 2019.
[7] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. 4, no. 2, pp. 100–107, Jul. 1968.
[8] M. Elbanhawi and M. Simic, "Sampling-based robot motion planning: A review," *IEEE Access*, vol. 2, pp. 56–77, 2014.
[9] L. Janson, E. Schmerling, A. Clark, and M. Pavone, "Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions," *Int. J. Robot. Res.*, vol. 34, no. 7, pp. 883–921, Jun. 2015.
[10] J. Wang, W. Chi, C. Li, C. Wang, and M. Q.-H. Meng, "Neural RRT*: Learning-based optimal path planning," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1748–1758, Oct. 2020.
[11] 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, Aug. 2014.
[12] 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," *J. Mech. Des.*, vol. 142, no. 2, Feb. 2020, Art. no. 021402.
[13] *N.-N. Introduction*. Accessed: Apr. 12, 2020. [Online]. Available: https://www.youtube.com/watch?v=JxZAMUZE37k
[14] R. González, P. Jayakumar, and K. Iagnemma, "Stochastic mobility prediction of ground vehicles over large spatial regions: A geostatistical approach," *Auto. Robots*, vol. 41, no. 2, pp. 311–331, Feb. 2017.
[15] R. Gonzalez, P. Jayakumar, and K. Iagnemma, "Generation of stochastic mobility maps for large-scale route planning of ground vehicles: A case study," *J. Terramech.*, vol. 69, pp. 1–11, Feb. 2017.
[16] K. 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," *J. Comput. Nonlinear Dyn.*, vol. 14, no. 2, Feb. 2019, Art. no. 021012.
[17] Y. Liu *et al.*, "Simulation-based mission mobility reliability analysis of off-road ground vehicles," *J. Mech. Des.*, vol. 143, no. 3, Mar. 2021, Art. no. 031701.

[18] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proc. ICRA. Millennium Conf. IEEE Int. Conf. Robot. Automat. Symposia*, vol. 2, Apr. 2000, pp. 995–1001.
[19] N. A. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 1617–1624.
[20] M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "Path planning for motion dependent state estimation on micro aerial vehicles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 3926–3932.
[21] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 723–730.
[22] A.-A. Agha-mohammadi, S. Chakravorty, and N. M. Amato, "FIRM: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements," *Int. J. Robot. Res.*, vol. 33, no. 2, pp. 268–304, Feb. 2014.
[23] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, Aug. 1996.
[24] B. Zhang, L. Tang, and M. Roemer, "Probabilistic planning and risk evaluation based on ensemble weather forecasting," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 2, pp. 556–566, Apr. 2018.
[25] W. Sun, J. van den Berg, and R. Alterovitz, "Stochastic extended LQR for optimization-based motion planning under uncertainty," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 437–447, Apr. 2016.
[26] J. van den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 895–913, Jun. 2011.
[27] J. van den Berg, S. Patil, and R. Alterovitz, "Motion planning under uncertainty using iterative local optimization in belief space," *Int. J. Robot. Res.*, vol. 31, no. 11, pp. 1263–1278, Sep. 2012.
[28] B. Zhang, L. Tang, J. DeCastro, M. J. Roemer, and K. Goebel, "A recursive receding horizon planning for unmanned vehicles," *IEEE Trans. Ind. Electron.*, vol. 62, no. 5, pp. 2912–2920, May 2015.
[29] S. Koenig and M. Likhachev, "Fast replanning for navigation in unknown terrain," *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 354–363, Jun. 2005.
[30] W. Sun, S. Patil, and R. Alterovitz, "High-frequency replanning under uncertainty using parallel sampling-based motion planning," *IEEE Trans. Robot.*, vol. 31, no. 1, pp. 104–116, Feb. 2015.
[31] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, Jun. 2011.
[32] S. Huang, S. Mahadevan, and R. Rebba, "Collocation-based stochastic finite element analysis for random field problems," *Probabilistic Eng. Mech.*, vol. 22, no. 2, pp. 194–205, Apr. 2007.
[33] C. Jiang, H. Qiu, L. Gao, D. Wang, Z. Yang, and L. Chen, "Real-time estimation error-guided active learning kriging method for time-dependent reliability analysis," *Appl. Math. Model.*, vol. 77, pp. 82–98, Jan. 2020.
[34] Z. Hu and S. Mahadevan, "A single-loop kriging surrogate modeling for time-dependent reliability analysis," *J. Mech. Des.*, vol. 138, no. 6, Jun. 2016, Art. no. 061406.
[35] K. Yang and S. Sukkarieh, "3D smooth path planning for a UAV in cluttered natural environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 794–800.
[36] M. Elbanhawi, M. Simic, and R. N. Jazar, "Continuous path smoothing for car-like robots using B-Spline curves," *J. Intell. Robotic Syst.*, vol. 80, no. S1, pp. 23–56, Dec. 2015.
[37] *ArcGIS/ENVI*. Accessed: Feb. 15, 2020. [Online]. Available: https://www.arcgis.com/features/index.html
[38] *GeotechData*. Accessed: Feb. 15, 2020. [Online]. Available: http://www.geotechdata.info/
[39] U. SSURGO. *Web Soil Survey*. Accessed: Feb. 20, 2020. [Online]. Available: https://websoilsurvey.sc.egov.usda.gov/App/HomePage.htm
[40] T. M. Wasfy, P. Jayakumar, D. Mechergui, and S. Sanikommu, "Prediction of vehicle mobility on large-scale soft-soil terrain maps using physics-based simulation," *Int. J. Vehicle Perform.*, vol. 4, no. 4, pp. 347–381, 2018.
[41] A. Sakai, D. Ingram, J. Dinius, K. Chawla, A. Raffin, and A. Paques, "PythonRobotics: A Python code collection of robotics algorithms," 2018, *arXiv:1808.10703*. [Online]. Available: http://arxiv.org/abs/1808.10703
[42] C. Jiang, Z. Hu, Y. Liu, Z. P. Mourelatos, D. Gorsich, and P. Jayakumar, "A sequential calibration and validation framework for model uncertainty quantification and reduction," *Comput. Methods Appl. Mech. Eng.*, vol. 368, Aug. 2020, Art. no. 113172.
[43] A. Ravankar, A. Ravankar, Y. Kobayashi, Y. Hoshino, and C.-C. Peng, "Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges," *Sensors*, vol. 18, no. 9, p. 3170, Sep. 2018.

Chen Jiang received the B.E. degree in engineering mechanics from Hunan University, Changsha, China, in 2015, and the Ph.D. degree in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2020.

He is currently a Post-Doctoral Research Fellow with the University of Michigan–Dearborn, Dearborn, MI, USA. His current research interests include motion and path planning, Bayesian inference, uncertainty quantification, surrogate modeling, and design under uncertainty.



Zhen Hu received the B.E. degree in mechanical engineering from Central South University, Changsha, China, in 2008, the master's degree in mechatronics engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2011, and the Ph.D. degree in mechanical engineering from the Missouri University of Science and Technology, Rolla, MO, USA, in 2014. He is currently an Assistant Professor with the Department of Industrial and Manufacturing Systems Engineering, University of Michigan-Dearborn, Dearborn, MI, USA.

His current research interests include vehicle mobility reliability analysis, design under uncertainty, uncertainty quantification, the Bayesian inference, and additive manufacturing.

Zissimos P. Mourelatos received the Ph.D. degree from the University of Michigan, Ann Arbor, MI, USA, in 1985.

He was the Chair of the Mechanical Engineering Department from 2010 to 2014 and the John F. Dodge Chair of Engineering from 2012 to 2016 at Oakland University, Rochester, MI, USA. Before joining Oakland University, he spent 18 years at the General Motors Research and Development Center, Warren, MI, USA. He is currently a Professor of Mechanical Engineering with Oakland University. He has published over 200 journal and conference publications. He conducts research in the areas of the design under uncertainty, reliability-based design optimization, vibrations, and dynamics.

Dr. Mourelatos is also a fellow of the American Society of Mechanical Engineers (ASME) and the Society of Automotive Engineering (SAE).

David Gorsich received the Ph.D. degree in applied mathematics from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2000.

He is currently the Army's Chief Scientist of the U.S. Army Combat Capabilities Development Command (CCDC), Ground Vehicle Systems Center (GVSC), Warren, MI, USA. He has published more than 150 conference papers and journal articles. His current research interests are vehicle dynamics and structural analysis, reliability-based design optimization, underbody blast modeling, terrain modeling, and spatial statistics.

Dr. Gorsich was a fellow of the Society of Automotive Engineering (SAE) in 2008 and the American Society of Mechanical Engineers (ASME) in 2020. He has served on the SAE Technical Standards Board. He has been the Chair of the SAE International Standards Committee for Ground Vehicle Reliability and on the SAE Board of Directors.

Paramsothy Jayakumar received the B.Sc.Eng. from the University of Peradeniya, Peradeniya, Sri Lanka, in 1982, and the M.S. and Ph.D. degrees in structural dynamics from the California Institute of Technology, Pasadena, CA, USA, in 1984 and 1987, respectively.

He has worked in research, development, and engineering of ground vehicle mobility, on-road and off-road, and intelligent vehicles. He is currently a Senior Technical Expert with the U.S. Army Combat Capabilities Development Command, GVSC, Warren, MI, USA. He has written more than 200 technical publications, including two book chapters, 80 journal articles, and 100 conference papers. He has delivered 50 invited speeches. He holds two patents.

Yan Fu received the B.S. degree from Tsinghua University, Beijing, China, in 1994, the M.S. degree from the Chinese Academy of Sciences, Beijing, in 1996, and the Ph.D. degree from Carnegie Mellon University, Pittsburgh, PA, USA, in 2000.

She is currently the Manager of the Strategy and Enterprise Analytics Group, Global Data, Insight and Analytics (GDIA), Ford Motor Company, Dearborn, MI, USA. She is a distinguished technical and managerial leader to innovative technologies, including artificial intelligence, big data analytics, safety optimization and robustness, and multidisciplinary design optimization. She has authored or coauthored over 150 technical articles in peer-reviewed journals and conference proceedings.

Dr. Fu is also a fellow of the Society of Automotive Engineering (SAE) and the American Society of Mechanical Engineers (ASME). She was a recipient of numerous honors and awards, including some of the most prestigious technical achievement awards, such as the 2011 SAE Henry Ford II Distinguish Award for Excellence in Automotive Engineering and the 2005, 2008, and 2018 Henry Ford Technology Awards.

Monica Majcher received the Ph.D. degree from Oakland University, Rochester, MI, USA, in 2016. Her Ph.D. dissertation is titled Time-Dependent Reliability Methods for Dynamic Systems and Linear Random Vibrations.

She is currently a Mechanical Engineer and a Research Scientist with the U.S. Army Ground Vehicle Systems Center (GVSC), Warren, MI, USA. She supports projects in a number of areas, including time-dependent reliability methods, reliability-centered maintenance strategies and optimization, operations data analysis, design for supportability, and uncertainty quantification.