A MODEL FOR OPTIMAL HUMAN NAVIGATION WITH STOCHASTIC EFFECTS*

CHRISTIAN PARKINSON†, DAVID ARNOLD‡, ANDREA L. BERTOZZI§, AND STANLEY OSHER¶

Abstract. We present a method for optimal path planning of human walking paths in mountainous terrain, using a control theoretic formulation and a Hamilton-Jacobi-Bellman equation. Previous models for human navigation were entirely deterministic, assuming perfect knowledge of the ambient elevation data and human walking velocity as a function of local slope of the terrain. Our model includes a stochastic component which can account for uncertainty in the problem, and thus includes a Hamilton-Jacobi-Bellman equation with viscosity. We discuss the model in the presence and absence of stochastic effects, and suggest numerical methods for simulating the model. We discuss two different notions of an optimal path when there is uncertainty in the problem. Finally, we compare the optimal paths suggested by the model at different levels of uncertainty, and observe that as the size of the uncertainty tends to zero (and thus the viscosity in the equation tends to zero), the optimal path tends toward the deterministic optimal path.

Key words. Optimal path planning, stochastic Hamilton-Jacobi-Bellman equation, stochastic optimal control, anisotropic control

AMS subject classifications. 00A69; 34H05; 35F21; 49L20

1. Introduction.

Optimal path planning is a classical problem in control theory and engineering. The basic mathematical formulation of the problem includes an agent whose movement is constrained by an equation of motion, and who desires to optimally travel from one point to another while obeying this equation. We will consider optimal path planning in the context of hikers traversing mountainous terrain.

Early approaches to the optimal path planning problem date back to the 1950's and were originally discrete in nature: discretizing the continuous domain into a weighted graph and then employing Dijkstra's algorithm and its many variants [12]. Several of these discrete or semi-discrete approaches are still being developed and improved today [18, 22, 29, 30, 35, 40]. However, since the late 1990's, there has also been significant interest in solving the problem using variational and partial different equation (PDE) based models. Sethian and Vladimirsky determined optimal paths on manifolds by formulating the dynamic programming principle and Hamilton-Jacobi-Bellman (HJB) equation as a continuous version of Dijkstra's algorithm [43, 44]. In a similar approach, one can use the level set method of Osher and Sethian [33] to resolve optimal paths [10, 37]. In application, variational methods for optimal path planning are extensively used in the context of self-driving vehicles. This problem was first considered by Dubins [13] in discrete form, but later reformulated continuously, and adapted to answer a number of questions involving impassable obstacles and reachable sets, among others [1, 5, 26, 49]. Human movement has also been modeled using HJB equations, whether it be walking while expending minimal energy [17, 36], reach-avoid games like capture-the-flag [9, 51], or pedestrian flow modeling [8].

*Submitted: October 29, 2019

†Department of Mathematics, UCLA, Los Angeles, CA 90095, (chparkin@math.ucla.edu).

‡Department of Mathematics, UCLA, Los Angeles, CA 90095, (darnold@math.ucla.edu). Department of Mathematics and Statistics, Macquarie University, Sydney 2109, Australia (david.arnold@mq.edu.au).

§Department of Mathematics, UCLA, Los Angeles, CA 90095, (bertozzi@math.ucla.edu).

¶Department of Mathematics, UCLA, Los Angeles, CA 90095, (sjo@math.ucla.edu).

One feature of all the models referenced above is that they are completely deterministic. For example, in the simplest reach-avoid games, the strategy of each team is known to the opposing team. Assumptions like this may not be realistic in practice, and thus it is important to incorporate some uncertainty into the models, and this can be done in a number of ways. In the case of reach-avoid games, Gilles and Vladimirsky suggest paths for the attackers or defenders that minimize or maximize risk in different ways [16]. In optimal path planning, Shen and Vladimirsky account for weather effects on a sailboat by designing a piecewise deterministic algorithm, wherein the travel velocity changes randomly but at discrete times [45]. These type of stochastic effects are of special interest to those working on self-driving vehicles wherein misaligned axles, miscalibrated sensors or a host of other variables could significantly perturb an optimal driving path, and these can be modeled using randomness in a number of different ways [3, 23, 25, 27, 28]. Others have suggested optimal path planning for underwater unmanned vehicles using a stochastic drift term to account for the effects of the ocean's current [24, 47, 48]

In this paper, we propose a method for optimal path planning where the walking velocity depends on the local slope in the direction or motion (and is thus anisotropic), and there is uncertainty in the equation of motion of the traveler. That is, rather than an ordinary differential equation, our equation of motion is a stochastic differential equation where Brownian motion can account for uncertainty in human walking speed, the ambient elevation data, terrain traversibility or other uncertainties. In doing so, we follow a similar formulation as in a previous paper [37]. However, since the Hamilton-Jacobi-Bellman equation for the stochastic case has viscosity, the level set method is no longer applicable, so we opt for a model more rooted in control theory. This paper is organized as follows: in section 2, we discuss the setup of our model, including some of the underlying mathematical formalism and how it pertains to optimal path planning. In section 3, we discuss the numerical methods which we used to simulate the model. In section 4, we discuss the results of our simulations. Specifically, we test our model against both synthetic data and real elevation data taken from the area surrounding El Capitan, a mountain in Yosemite National Park. We compare two different notions of optimal paths when stochastic effects are present. Finally, we observe that as the size of the uncertainty tends to zero (and thus the solution of the stochastic HJB equation tends to the viscosity solution of the ordinary HJB equation [11]), the optimal path tends toward the deterministic optimal path.

2. Mathematical Model. We discuss the construction of our model, only briefly mentioning the underlying control theoretic concepts. Treatments of these concepts with varying levels of rigor can be found in several books [6, 7, 15, 31, 38]. The main insights here are how the controlled equation of motion leads to a Hamilton-Jacobi-Bellman equation, and how we can use this equation to determine the optimal control.

2.1. The Deterministic Optimal Path Planning Model. We consider the problem of planning optimal walking paths in mountainous terrain. We imagine that a hiker is standing at a point $x_0 \in \mathbb{R}^2$, and wishes to walk to a point $x_{\text{end}} \in \mathbb{R}^2$. We are given the elevation profile $E : \mathbb{R}^2 \to \mathbb{R}$ which provides the elevation at any point in space, and velocity function $V : \mathbb{R} \to [0,\infty)$ which measures walking velocity as a function of slope. Here slope is in units of grade, so that slope of 1 corresponds to a 45° incline, and slope of -1 corresponds to a 45° decline. Let $\boldsymbol{x} : [0,T] \to \mathbb{R}^2$ represent the current position of the hiker, where $T > 0$ is the total walking time. Note, this terminal time T is a parameter which one specifies at the outset. Our control variable will be the walking direction along the path, $\boldsymbol{s} : [0,T] \to \mathbb{S}^1 := \{a \in \mathbb{R}^2 : \|a\|_2 = 1\}$. Given all of this, the equation of motion for our hiker is

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= V(\nabla E(\boldsymbol{x}(t))\cdot \boldsymbol{s}(t))\boldsymbol{s}(t), \ 0<t\le T,\\ \boldsymbol{x}(0) &= x_0.\end{aligned} \tag{2.1}$$

Note that $\nabla E(\boldsymbol{x})\cdot \boldsymbol{s}$ represents the local slope in the walking direction.

For the cost functional, we simply consider Euclidean distance to the end point. Given a particular control parameter $\boldsymbol{s}:[0,T]\to\mathbb{S}^1$, define

$$C[\boldsymbol{s}(\cdot)] = \|\boldsymbol{x}(T)-x_{\text{end}}\|_2, \tag{2.2}$$

where $\boldsymbol{x}$ is constrained by (2.1). To arrive at the Hamilton-Jacobi-Bellman equation, we invoke the dynamic programming principle, stating that to find a globally optimal path, we need only consider paths which are locally optimal. That is, for any given $x\in\mathbb{R}^2$ and $t\in(0,T]$, one can consider optimal paths on the interval $(t,T]$ with $\boldsymbol{x}(t)=x$. We define the cost function by

$$u(x,t) = \inf_{\boldsymbol{s}}\left\{C_{x,t}[\boldsymbol{s}(\cdot)]\right\}, \tag{2.3}$$

where $C_{x,t}[\boldsymbol{s}(\cdot)]$ denotes the same cost functional, but restricted to paths $\boldsymbol{x}$ on the interval $(t,T]$ with $\boldsymbol{x}(t)=x$. Provided that the map $x\mapsto V(\nabla E(x)\cdot s)$ is sufficiently regular [15], one can show that the cost function u satisfies the equation

$$\begin{aligned}&u_t(x,t)+\inf_{s\in\mathbb{S}^1}\{V(\nabla E(x)\cdot s)[\nabla u(x,t)\cdot s]\}=0,\\ &u(x,T)=\|x-x_{\text{end}}\|_2.\end{aligned} \tag{2.4}$$

This is the Hamilton-Jacobi-Bellman (HJB) equation for this control problem. Note that this is a terminal value problem, for which we have data at time T and we solve backwards in time on the interval $[0,T)$. This correspondence between the control problem and the HJB equation will hold under the condition that $x\mapsto V(\nabla E(x)\cdot s)$ is Lipschitz with a constant which is uniform over $s\in\mathbb{S}^1$ [15]. In our application, this will likely hold, but we cannot guarantee it because we will use real elevation data $E(x)$. Thus this may only be a formal correspondence, though empirically, the method appears to work even when the elevation data is somewhat non-smooth. Another note is that the infimum in (2.4) is being taken over a compact set, so again, under some mild assumptions of continuity, this will be realized as a minimum.

The question remains of how to use (2.4) to determine the optimal control, and thus the optimal trajectory. At each point, the optimal control is given by the infimum in (2.4). Thus if one can solve (2.4), the optimal control $\boldsymbol{s}^*$ is

$$\boldsymbol{s}^*(x,t) = \operatorname*{argmin}_{s\in\mathbb{S}^1}\{V(\nabla E(x)\cdot s)[\nabla u(x,t)\cdot s]\}. \tag{2.5}$$

When this minimization problem is solved, the optimal trajectory is then resolved by solving the differential equation

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= V(\nabla E(\boldsymbol{x}(t))\cdot \boldsymbol{s}^*(\boldsymbol{x}(t),t))\boldsymbol{s}^*(\boldsymbol{x}(t),t), \ 0<t\le T,\\ \boldsymbol{x}(0) &= x_0.\end{aligned} \tag{2.6}$$

It was mentioned above that the terminal time T is a parameter which one must specify beforehand. It is important to note what this means for the model. Notice that for our cost functional, we have chosen a lump sum cost which is the distance from the end of the path $\boldsymbol{x}(T)$ to the desired end point x_{end}. The optimal control problem attempts to minimize this cost. Thus the path that we observe will be the path $\boldsymbol{x}$ whose end point is as close to x_{end} as possible, given time $T>0$. If we select T too small, the path will not reach the end point, and in some cases this can lead to some interesting decisions regarding how the path should be constructed. We discuss this further in section 4.3.

2.2. The Stochastic Optimal Path Planning Model. In the above, we assume that all data is known perfectly and that there is no uncertainty. In reality, weather effects, instrumentation noise, incomplete elevation data or any number of other things could cause uncertainty in the equation of motion. Accordingly, we can account for random effects by considering a stochastic equation of motion

$$\begin{aligned} &d\boldsymbol{X}_t = V(\nabla E(\boldsymbol{X}_t)\cdot \boldsymbol{s}(t))\boldsymbol{s}(t)dt + \sigma(\boldsymbol{X}_t,\boldsymbol{s}(t))d\boldsymbol{W}_t, \quad 0<t\leq T \\ &\boldsymbol{X}_0 = x_0, \end{aligned} \tag{2.7}$$

where $\boldsymbol{W}_t$ is an ordinary two-dimensional Brownian motion whose coordinates $W_t^{(1)}$ and $W_t^{(2)}$ are independent one-dimensional Brownian motions, and σ is some function that determines the uncertainty.

In this case, we define the expected cost function

$$C[\boldsymbol{s}(\cdot)] = \mathbb{E}(\|\boldsymbol{X}_T - x_{\text{end}}\|_2), \tag{2.8}$$

and perform similar steps as above to arrive at a stochastic version of the Hamilton-Jacobi-Bellman equation. That is, we localize the problem and define

$$u(x,t) = \inf_{\boldsymbol{s}} \left\{ \mathbb{E}\big(C_{x,t}[\boldsymbol{s}(\cdot)]\big) \right\}, \tag{2.9}$$

where again $C_{x,t}[\boldsymbol{s}(\cdot)]$ denotes the cost functional restricted to trajectories $\boldsymbol{X}$ on the interval $(t,T]$ with $\boldsymbol{X}_t = x$. A key point in the derivation of the HJB equation invokes the chain rule for the function $t\mapsto u(\boldsymbol{x}(t),t)$, which explains the appearance of u_t and ∇u in the HJB equation. We are now concerned with the map $t\to u(\boldsymbol{X}_t,t)$ and we must consider second order derivatives because of the fundamental relationship $(dW_t^{(i)})^2\sim dt$ for $i=1,2$. Thus the stochastic Hamilton-Jacobi-Bellman equation reads

$$\begin{aligned} &u_t(x,t) + \inf_{s\in\mathbb{S}^1}\left\{ V(\nabla E(x)\cdot s)[\nabla u(x,t)\cdot s] + \frac{1}{2}\sigma(x,s)^2\Delta u(x,t)\right\} = 0, \\ &u(x,T) = \|x - x_{\text{end}}\|_2. \end{aligned} \tag{2.10}$$

Again, this is a terminal value problem, solved backwards from $t=T$ to $t=0$. Thus the positive sign on the diffusion is the correct sign to ensure that the diffusion has a smoothing effect on the solution, and there is no danger of finite-time blow-up.

As above, to determine the optimal control, we solve (2.10), and at each point set

$$\boldsymbol{s}^*(x,t) = \operatorname*{argmin}_{s\in\mathbb{S}^1}\left\{ V(\nabla E(x)\cdot s)[\nabla u(x,t)\cdot s + \frac{1}{2}\sigma(x,s)^2\Delta u(x,t)]\right\}. \tag{2.11}$$

At this point, there is a choice as to how to construct the path. One can use the optimal control computed from the stochastic HJB equation but simulate the deterministic path equation

$$\begin{aligned} &\dot{\boldsymbol{x}}(t) = V(\nabla E(\boldsymbol{x}(t))\cdot \boldsymbol{s}^*(\boldsymbol{x}(t),t))\boldsymbol{s}^*(\boldsymbol{x}(t),t), \quad 0<t\leq T, \\ &\boldsymbol{x}(0) = x_0, \end{aligned} \tag{2.12}$$

to arrive at an optimal path. Alternatively, to compute a single instance of a path, we could simulate the equation

$$\begin{aligned} &d\boldsymbol{X}_t = V(\nabla E(\boldsymbol{X}_t)\cdot \boldsymbol{s}^*(X_t,t))\boldsymbol{s}^*(X_t,t)dt + \sigma(\boldsymbol{X}_t, \boldsymbol{s}^*(\boldsymbol{X}_t,t))d\boldsymbol{W}_t, \quad 0<t\leq T \\ &\boldsymbol{X}_0 = x_0. \end{aligned} \tag{2.13}$$

Because this is only one instance and is subject to randomness, the path will not necessarily connect the points x_0 and x_{end}. However, we can average over many realizations to arrive at an expected optimal path. This is summarized in table 1.

Table 1: The two options for how to compute optimal paths with uncertainty.

Method (i)	Method (ii)
Stochastic HJB Equation (2.10) $\rightarrow$ control values	Stochastic HJB Equation (2.10) $\rightarrow$ control values
Deterministic equation of motion (2.12) $\dot{\boldsymbol{x}} = V(\nabla E(\boldsymbol{x})\cdot \boldsymbol{s})\boldsymbol{s}$ $\rightarrow$ optimal path	Stochastic equation of motion (2.13) $\mathbb{E}\big(\ d\boldsymbol{X}_t = V(\nabla E(\boldsymbol{X}_t)\cdot \boldsymbol{s})\boldsymbol{s}dt + \sigma d\boldsymbol{W}_t\ \big)$ $\rightarrow$ expected optimal path

These methods have different interpretations and may be better suited to modeling different physical scenarios. In using method (i), the uncertainty is factored into the planning of the route, but upon traversing the route, there is no uncertainty in the velocity. This could model a hiker walking through a forest. The hiker does not feel random perturbations in the walking velocity at each step; rather, the uncertainty is in the exact form of the landscape that lies ahead. Method (ii) may be of more practical use to a company shipping goods from one port to another, wherein each boat that makes the trip will actually feel random perturbations in velocity due to wind or currents. We will use both methods to compute paths and compare the results in section 4.

2.3. Our Model. In order to simulate the model, we simply need to determine some parameter values, specifically those of E, V and σ. For the elevation data E, we will begin by using synthetic elevation data which is mostly flat but with a few "mountains" included which we would expect the hiker to avoid. After this, we will use real elevation data in the area surrounding El Capitan, a large granite cliff face in Yosemite National Park in California. We will specify the elevation data which we use for each simulation in section 4.

For the walking velocity function, we use a modified version of the function of Irmischer and Clarke [19]. Our specific velocity function is [37]:

$$V(S) = 1.11 \exp\left(-\frac{(100S+2)^2}{2345}\right). \tag{2.14}$$

Additionally, if we use this velocity function and only consider the walking direction $\nabla E(x)\cdot s$, then the walking velocity in the tangential and normal directions to the path are completely decoupled. Thus one could walk easily in the east-west direction, even when the grade in the north-south direction is arbitrarily steep. To prevent this, we penalize the velocity if the grade is sufficiently steep in any direction [37].

Lastly, one must determine the exact form of the uncertainty σ in the stochastic equation of motion. For our purposes, we take σ constant, so that we model uncertainty in the walking velocity in a general sense without specifying the exact nature of the uncertainty. As a consequence, if we reconsider (2.10), we notice that the viscosity term is independent of the control variable s, and thus the equation can be re-written

$$
\begin{aligned}
&u_t(x,t)+\frac{\sigma^2}{2}\Delta u(x,t)+\inf_{s\in\mathbb{S}^1}\{V(\nabla E(x)\cdot s)[\nabla u(x,t)\cdot s]\}=0,\\
&u(x,T)=\|x-x_{\text{end}}\|_2.
\end{aligned} \tag{2.15}
$$

This case is interesting because now the optimal control is resolved exactly as in the deterministic case, and (2.15) is reminiscent of the viscous Hamilton-Jacobi equation considered by Crandall and Lions [11] when establishing the vanishing viscosity method for Hamilton-Jacobi equations. Thus if our Hamiltonian

$$
H(x,p):=\inf_{s\in\mathbb{S}^1}\{V(\nabla E(x)\cdot s)[p\cdot s]\} \tag{2.16}
$$

is continuous, we expect that the solution $u^{(\sigma)}$ to equation (2.15) will converge to the viscosity solution u of (2.4) as $\sigma\searrow 0$ [11]. Again, continuity of $H(x,p)$ will depend on the nature of the elevation data, but we can observe this convergence empirically by considering the optimal path constructed by the the method at varying levels of uncertainty σ.

3. Numerical Methods.

There are several numerical concerns that must be addressed in order to simulate these equation; both general concerns for numerical Hamilton-Jacobi equations and specific concerns relating to our model. We begin by discussing general notions for solving Hamilton-Jacobi type equations numerically.

3.1. An Explicit Scheme for (2.4).

We consider a general Hamilton-Jacobi equation of the form

$$
u_t+H(u_x,u_y)=0,\ \ x\in\mathbb{R}^2,\ t>0, \tag{3.1}
$$

with initial data at $t=0$. Our Hamiltonian (2.16) depends also on x, but for the sake of numerical methods, this is unimportant, so we suppress it here to simplify notation. Since solutions of Hamilton-Jacobi equations have kinks [4, 11, 32], naïve forward, backward and centered differencing methods may fail to accurately simulate the equation. Accordingly, in a numerical scheme, one must replace the Hamiltonian $H(u_x,u_y)$ with a *numerical Hamiltonian* $\hat{H}(u_x^+,u_x^-,u_y^+,u_y^-)$ which deftly combines the forward differences u_x^+,u_y^+ and backwards differences u_x^-,u_y^- so as to smooth the solution, minimize oscillation near kinks, track the characteristics, or otherwise capture the dynamics of the equation.

Following Osher and Shu [34], we use the Godunov numerical Hamiltonian

$$
\hat{H}_G(u_x^+,u_x^-,u_y^+,u_y^-)=\operatorname*{ext}_{u\in I(u_x^-,u_x^+)}\ \operatorname*{ext}_{v\in I(u_y^+,u_y^-)} H(u,v) \tag{3.2}
$$

where

$$
I(a,b)=[\min(a,b),\max(a,b)] \qquad \text{and} \qquad \operatorname*{ext}_{x\in I(a,b)}=\begin{cases}\min_{a\le x\le b} & \text{if } a\le b,\\ \max_{b\le x\le a} & \text{if } a>b.\end{cases} \tag{3.3}
$$

These minima and maxima are designed to track the characteristics of the equation, thus accurately approximating the Hamiltonian without resorting to excessive numerical diffusion as is present in the Lax-Friedrichs scheme; we comment on this further later.

Osher and Shu [34] suggest a method for approximating u_x and u_y with forward and backward difference schemes which are accurate to arbitrarily high order, though we opt for first order approximations which suffice in this application. In the absensce of viscosity, we then explicitly integrate (3.1) in time using explicit Euler time stepping.

Specifically while (2.4) has infinite spatial domain, for the purpose of simulating it, we draw some box $[a,b]\times[c,d]$ containing the points x_0 and x_{end}. We also invert the time variable, so that we instead solve for $u(x,y,T-t)$ on the interval $t\in(0,T]$. Next discritize this box, and the time interval uniformly:

$$\begin{aligned} x_i &= a+i\frac{(b-a)}{N}, \quad i=0,1,\dots,N,\\ y_j &= c+j\frac{(d-c)}{M}, \quad j=0,1,\dots,M,\\ t_k &= \frac{kT}{K}, \qquad\quad k=0,1,\dots,K. \end{aligned} \tag{3.4}$$

Then if $u_{ij}^k := u(x_i,y_j,T-t_k)$, our explicit time stepping scheme is

$$u_{ij}^k = u_{ij}^{k-1} + \Delta t \hat{H}_G(u_x^+,u_x^-,u_y^+,u_y^-)_{ij}^{k-1} \tag{3.5}$$

for $i=1,\dots,N-1, j=1,\dots,M-1$ and $k=1,\dots,K$, where $\Delta t := T/K$. Note, the positive sign in front of $\hat{H}_G$ in (3.5) is due to the time inversion; this would ordinarily be negative when moved to the right hand side of (2.4). If one is resolving the spatial derivatives to higher order accuracy and desires to maintain this accuracy, one can easily replace the explicit Euler time integration with a higher order Runge-Kutta scheme [34].

The layers of nodes corresponding to $i=0, i=N, j=0$ or $j=M$ are necessarily boundary layers, because for example, we cannot compute the backwards difference approximation u_x^- at the nodes where $i=0$. Thus after evaluating (3.5) at each time step, we must enforce some artificial boundary condition, which is not prescribed in the differential equation, but is rather a purely numerical consideration. We use the boundary conditions suggested by Kao et al. [20] which extrapolate while also attempting to minimize oscillation and prevent information from entering through the boundary:

$$\begin{aligned} u_{0,j}^k &= \min(\max(2u_{1,j}^k - u_{2,j}^k, u_{2,j}^k), u_{0,j}^{k-1}),\\ u_{N,j}^k &= \min(\max(2u_{N-1,j}^k - u_{N-2,j}^k, u_{N-2,j}^k), u_{N,j}^{k-1}),\\ u_{i,0}^k &= \min(\max(2u_{i,1}^k - u_{i,2}^k, u_{i,2}^k), u_{i,0}^{k-1}),\\ u_{i,M}^k &= \min(\max(2u_{i,M-1}^k - u_{i,M-2}^k, u_{i,M-2}^k), u_{i,M}^{k-1}). \end{aligned} \tag{3.6}$$

For our particular application, the maximum velocity at which information flows along characteristics is $V_{\max} := \max_S V(S) = 1.11$, and so this scheme will be stable so long as the parameters $(\Delta x, \Delta y, \Delta t) := \left(\frac{b-a}{N}, \frac{d-c}{M}, \frac{T}{K}\right)$ satisfy the CFL condition [32]:

$$\Delta t \cdot V_{\max}\left(\frac{1}{\Delta x}+\frac{1}{\Delta y}\right) < 1. \tag{3.7}$$

3.2. A Semi-Implicit Scheme for (2.15). In order to numerically simulate the reaction-diffusion equation (2.15), one could simply insert the centered difference approximation to Δu and continue with the explicit Euler time stepping. However, this will require exceedingly small time discretization, since the stability condition for forward Euler time stepping for a diffusion operator is of the form $\Delta t = O((\Delta x)^2, (\Delta y)^2)$.

Instead, we resolving the diffusion implicitly. Thus, our scheme for (2.15) is

$$u_{ij}^k - \frac{\sigma^2 \Delta t}{2\Delta x}(u_x^+ - u_x^-)_{ij}^k - \frac{\sigma^2 \Delta t}{2\Delta y}(u_y^+ - u_y^-)_{ij}^k = u_{ij}^{k-1} + \Delta t \hat{H}_G(u_x^+, u_x^-, u_y^+, u_y^-)_{ij}^{k-1}. \tag{3.8}$$

Since implicit Euler time stepping for diffusion is unconditionally stable, our discretization is still only bound by the CFL condition (3.7). For larger values of σ, the resulting diffusion will smooth the solution u, and thus sophisticated numerical Hamiltonians are probably no longer necessary. However, we have implemented this scheme as stated, so that as $\sigma \searrow 0$, the semi-implicit scheme (3.8) reverts to the explicit scheme (3.5), and there are no stability issues. We note that this semi-implicit scheme is only available when the uncertainty σ in the equation of motion is independent of the control variable. If σ depends on s, then the Hamilton-Jacobi-Bellman equation takes the form (2.10), and in this case the Hamiltonian cannot be decoupled from the diffusion term. Thus one would have to resort to explicit time stepping methods, or use alternate numerical methods such as pseudospectral methods [46].

3.3. Implementation Concerns. One may question why we have decided to use the Godunov scheme, since for the purposes of implementation, something like the Lax-Friedrichs numerical Hamiltonian may be easier. Indeed, the Lax-Friedrichs Hamiltonian in given by

$$\hat{H}_{LF}(u_x^+, u_x^-, u_y^+, u_y^-) = H\left(\frac{u_x^+ + u_x^-}{2}, \frac{u_y^+ + u_y^-}{2}\right) - \frac{\alpha_1}{2}(u_x^+ - u_x^-) - \frac{\alpha_2}{2}(u_y^+ - u_y^-), \tag{3.9}$$

where α_1, α_2 are bounds on the derivatives of H with respect to the first or second arguement, respectively [34]. The Lax-Friedrichs scheme works by adding numerical diffusion, thus in essence changing equation (3.1) to

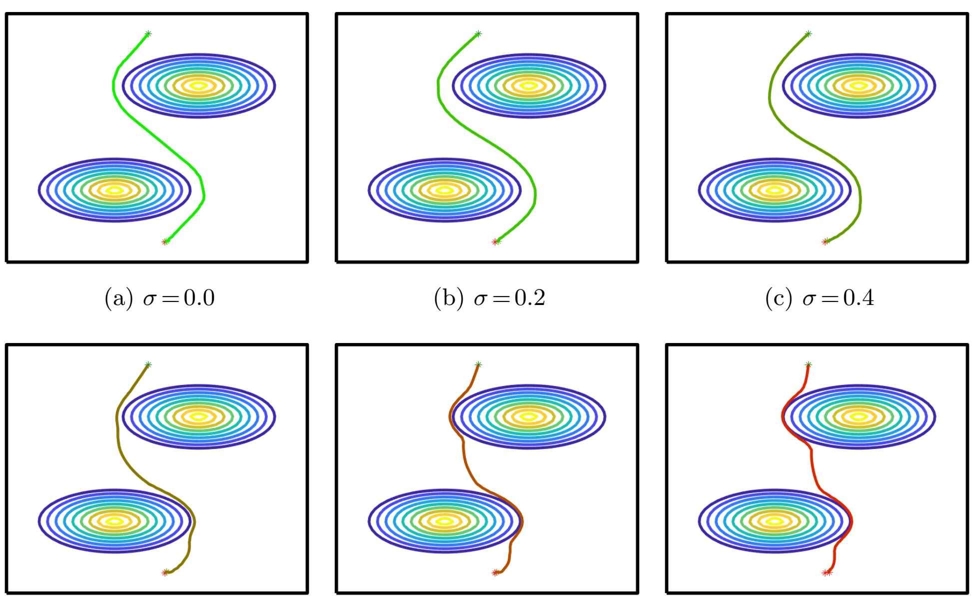
$$u_t - \varepsilon \Delta u + H(u_x, u_y) = 0, \tag{3.10}$$

where $\varepsilon = O(\Delta x, \Delta y)$. In most applications, this is acceptable, since the Lax-Friedrichs Hamiltonian still provides a first order accurate approximation to the Hamilton-Jacobi equation. However, in our case, adding diffusion at level ε to the Hamilton-Jacobi equation is akin to adding uncertainty in the equation of motion at level $\varepsilon^{1/2}$. For example, in the discretization we will use, the numerical diffusion would be on the order of 0.01, which would correspond to uncertainty in the equation of motion on the order of 0.1 m/s. This is a nontrivial level of uncertainty, representing roughly one tenth of the maximum walking velocity. For this reason, minimally diffusive schemes are necessary for our application. A similar roadblock arises when using level set methods, especially when the geometry of the level sets is a crucial aspect of the model like in shock-capturing, image-processing [14] or recent deforestation models [2]. In these cases, numerical diffusion becomes noticeable when level set velocity is near zero.

If the Hamiltonian H is relatively simple, the minima and maxima in the Godunov Hamiltonian can be resolved exactly. For example, in the special case that $H(u,v) = h(u^2, v^2)$ and h is monotone in each argument, the Godunov scheme simplifies significantly [37]. In our application, this situation arises when the elevation is flat so that $\nabla E \equiv 0$, and our Hamiltonian (2.16) becomes $H(x, \nabla u) = -V_{\max}\|\nabla u\|_2$. In general, our Hamiltonian is much more complicated. In order to implement the Godunov scheme for our Hamiltonian, we must resolve three minima or maxima: the minimum involved in the definition of the Hamiltonian, and the two minima/maxima involved in the scheme itself. We resolve all this minima and maxima discretely, by simply sampling points and choosing the correct one. As long as the error from this descrete optimization remains on the order of Δx and Δy, the scheme will remain accurate to first order. The level of resolution needed for the discrete optimization depends somewhat on the problem, but empirically it appears that the most important facet is the regularity of the elevation data E. This makes intuitive sense: for less smooth elevation, the minimization in (2.16) taken with respect to the walking direction will require finer resolution to resolve. Likewise, for less smooth elevation, the discontinuities in the derivative of the solution u become more severe. Thus the optimization sets in the Godunov scheme, which have the form $I(u_x^+, u_x^-)$ and $I(u_y^+, u_y^-)$, become larger.

4. Simulations & Results. We implemented our model in MATLAB using the numerical schemes (3.5, 3.8) to solve the Hamilton-Jacobi-Bellman equations (2.4, 2.15), and the forward Euler method to solve the equation of motion (2.12). In the following figures, we will display elevation contours ranging from blue signifying low elevation to yellow signifying high elevation. The starting point x_0 will be marked with a green star and the desired end point x_{end} will be marked with a red star. The lines representing the walking paths will be plotted in colors ranging from green, symbolizing simulations with smaller σ values, to red, symbolizing larger σ values.

4.1. Synthetic Elevation Data. We began by testing our model against simple synthetic data. In figure 1, we have computed optimal paths with several different levels of uncertainty σ. Referring to table 1, we are using method (i) to compute the optimal paths. That is, we are using the stochastic HJB equation to determine the optimal control values, but then computing the path using the deterministic equation of motion. In this example, the elevation is flat except for two large mountains which lie between the starting point and end point.



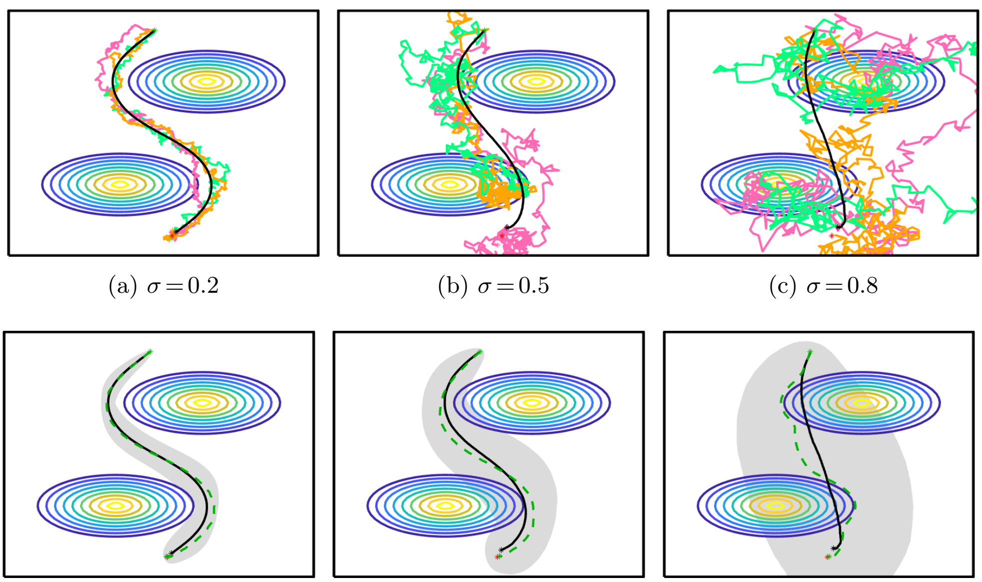
(a) $\sigma = 0.0$ (b) $\sigma = 0.2$ (c) $\sigma = 0.4$

(d) $\sigma = 0.6$ (e) $\sigma = 0.8$ (f) $\sigma = 1.0$

Fig. 1: Optimal paths using different σ values. End time $T = 3.8$ for each plot.

In the deterministic case, plotted in figure 1a, the path curves around the mountains as one would expect: the walking velocity is significantly hampered by the change in elevation, so it is more efficient to avoid those regions. In this case, we see that the optimal path suggested by our algorithm is not particularly sensitive to small changes in σ. The path in figure 1b which has $\sigma = 0.2$ looks very similar to that in figure 1a which has $\sigma = 0$. However, as σ becomes larger, we do see significant changes in the path. The path in figure 1f where $\sigma = 1$ is significantly different from the deterministic case. Here the uncertainty is on roughly the same order as the walking velocity. With this level of uncertainty, one could imagine walking through a forest in a very dense fog. In planning the path, this algorithm suggests that you walk directly toward your destination and adjust as necessary when obstacles arise.

Next, we consider method (*ii*) from table 1; that is, we simulate the stochastic equation of motion many times and compute the average path. Here we are still using the forward Euler method for the stochastic ODE and since the coefficient in front of the Brownian motion is independent of X_t, this corresponds with the Milstein method which exhibits strong and weak convergence at first order [21]. In each of the following results, we have simulated the equation of motion 10000 times and taken the average path. Results are displayed in figures 2a-2c. The black line represents the average optimal path and the colored lines represent three individual realizations of the stochastic equation of motion. Here as σ gets larger, the individual realizations become less meaningful, but the average path is still somewhat smooth and roughly connects the starting point to the ending point.



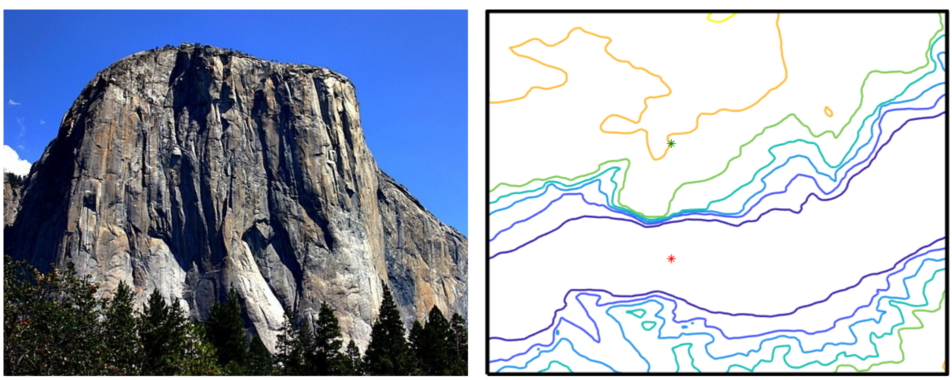
(a) $\sigma = 0.2$ (b) $\sigma = 0.5$ (c) $\sigma = 0.8$

(d) $\sigma = 0.2$ (e) $\sigma = 0.4$ (f) $\sigma = 0.8$

Fig. 2: (a) - (c) Average path over 10000 trials (black), and three realizations of the stochastic equation of motion (colored) at different levels of uncertainty σ. (d) - (f) Average path (black) with standard deviation (grey), and the path computed using method (*i*) (dotted green).

We also calculate a form of confidence interval to evaluate how close a single realization is likely to be to the average. To do this, at each point $(\overline{x},\overline{y})$ along the average path, we calculate the standard deviation (δ_x,δ_y) in each of the coordinates. Then at each point, we plot in light grey the ellipse centered at $(\overline{x},\overline{y})$ with radii (δ_x,δ_y) in the x or y direction, respectively. As we travel along the path plotting these ellipses, the grey envelope represents the set of points within one standard deviation of the average path. This is seen in figures 2d-2f. In these plots the average path is plotted as a solid black line. Now we also display the path which was computed using method (i) using a dotted green line. For small σ, the average path and the determistic path match fairly well. For larger σ, they begin to diverge, but the walking strategy seems similar: for larger σ, the average paths take a much more direct approach, cutting corners more closely, or walking directly over the mountains. In each case, the deterministic path stays well within one standard deviation of the average path. Notice that as σ gets larger, the standard deviation grows very quickly so that in figure 2f the set of possible paths within one standard deviation of the average is quite large, and it may simply be that method (i) provides a more reasonable solution in this application.

4.2. Real Elevation Data. Seeing that our model works correctly for simplified elevation data, we tested the model against real elevation in the area surrounding the mountain El Capitan in Yosemite National Park. The elevation profile of El Capitan, along with the starting an ending points, is pictured in figure 3. Notice that directly in between the starting and ending points, the contour lines lie close together, indicating a sheer cliff face. The starting point is near the summit of the mountain, and the ending point is in the valley to the south of the mountain, so any walking path should choose the gentler grades to the east or west of the cliff face.



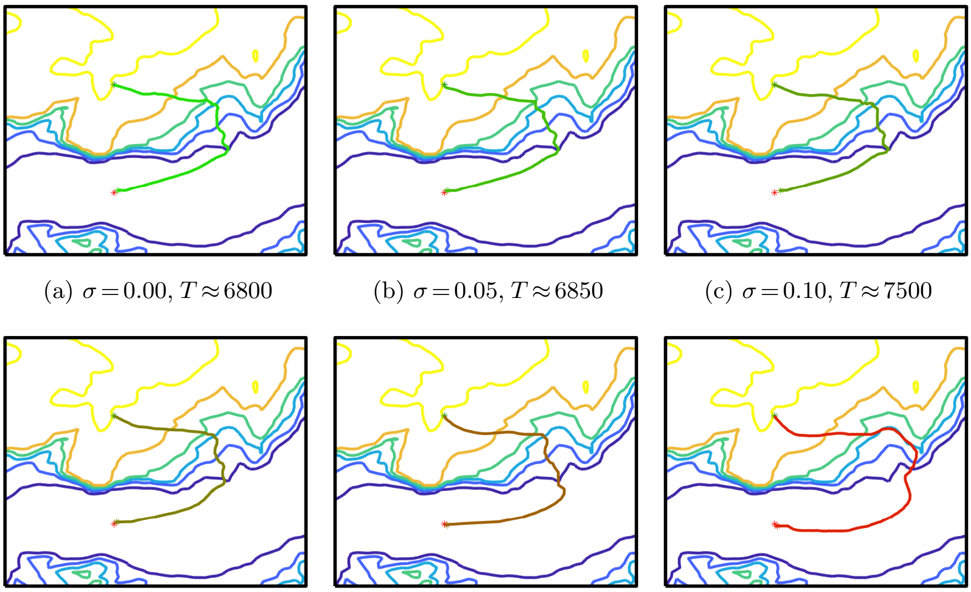
(a) The south-facing cliff face of El Capitan (b) The elevation profile of El Capitan.

Fig. 3: El Capitan, Yosemite National Park, California[1]

Indeed, this is exactly what we observe, as seen in figure 4. These paths were determined using method (i), the deterministic equation of motion. In these simulations all the scales in the problem are completely genuine. The region displayed in these figures is a rectangle roughly 5 kilometers east-to-west and 6 kilometers north-to-south. The

[1]Image courtesy of Mike Murphy, uploaded to Wikipedia Commons under Share-Alike license: https://commons.wikimedia.org/wiki/File:Yosemite_El_Capitan.jpg

starting and ending points are roughly 2 kilometers apart and the terrain is mountainous, and so several thousand seconds are required to traverse a path connecting the points. Here, the elevation data is much less smooth, and this leads to a greater sensitivity to small changes in σ. Figure 4b shows that at $\sigma = 0.05$, the optimal path looks largely the same as in the deterministic case, displayed in figure 4a. However, when $\sigma = 0.3$, the optimal path is quite different, as seen in figure 4f.



(a) $\sigma = 0.00$, $T \approx 6800$

(b) $\sigma = 0.05$, $T \approx 6850$

(c) $\sigma = 0.10$, $T \approx 7500$

(d) $\sigma = 0.15$, $T \approx 7550$

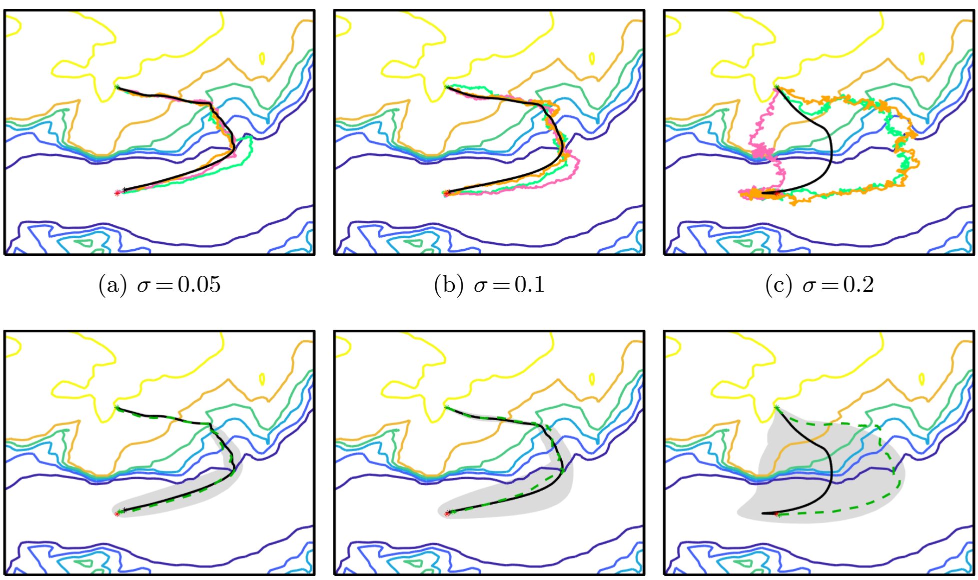
(e) $\sigma = 0.20$, $T \approx 8950$

(f) $\sigma = 0.30$, $T \approx 15550$

Fig. 4: Optimal paths descending El Capitan using different σ values.

One significant note here: at different levels of σ, there are different optimal terminal times T. Recall, the parameter T must be chosen before simulating the model. In the case of the synthetic data in figure 1, the terminal time T is not particularly sensitive to changes in σ, since qualitatively the paths are all similar and the amount of time that is "wasted" by taking a non-optimal path is not significant. In that case, we set $T = 3.8$ and any path with $\sigma \in [0,1]$ had sufficient time to reach the endpoint. This is not the case in figure 4, where small changes in σ lead to more significant qualitative changes in the paths. Indeed, the greedy strategy of taking a more direct route and then adjusting as necessary can be very costly in the case of El Capitan where it is very easy to get stuck in regions of severe grades, and be nearly unable to move. In this case, if one is reasonably uncertain about walking velocity as in figure 4f, the algorithm suggests one should allow ample time, and take a safer route which more deliberately avoids regions with large changes in elevation. Because this route is significantly different, it requires a terminal time of roughly $T = 15550$ seconds, as opposed to a terminal time closer to $T = 6800$ seconds as in figure 4a. For larger values of σ (for example $\sigma > 0.5$), the path will not make it down the mountain even given exorbitantly large terminal time, because it will walk too close to the cliff, become stuck, and have insufficient time to adjust. We say more about the role of the parameter T, especially as it pertains to impassable obstacles such as the El Capitan cliff face, in section 4.3.

As in the previous section, we would also like to use method (ii) to construct a path. In figures 5a-5c, we plot the average path along with three realizations in the case that $\sigma = 0.05, 0.1$ and 0.2. When $\sigma = 0.05, 0.1$, each of these realizations is fairly close to the average path, and the results are similar to those obtained using method (i). We have also included the region which is one standard deviation away from the average path, as seen in figures 5d-5f. Even when σ is very small, the variance in how the paths descend the mountain is fairly large. This is because small perturbations in that region will more qualitatively change the course of path. The results for $\sigma = 0.2$—displayed in figures 5c,5f—do not seem particularly meaningful. In this case, the uncertainty in the walking velocity is large enough that if the path approaches the large cliff face, the random perturbation can cause the path to move down the cliff. In this region, the walking velocity is approximately zero, and so the random effects are the driving force for the movement. In those simulations, a large enough portion of the paths descended the cliff in this manner, leading to a skewed average path, and an enormously large standard deviation. Similar problems may arise whenever there are regions where the walking velocity is very small. In such cases, it seems that using the deterministic equation of motion with the stochastic control values (as in figure 4) will give a much more meaningful result.



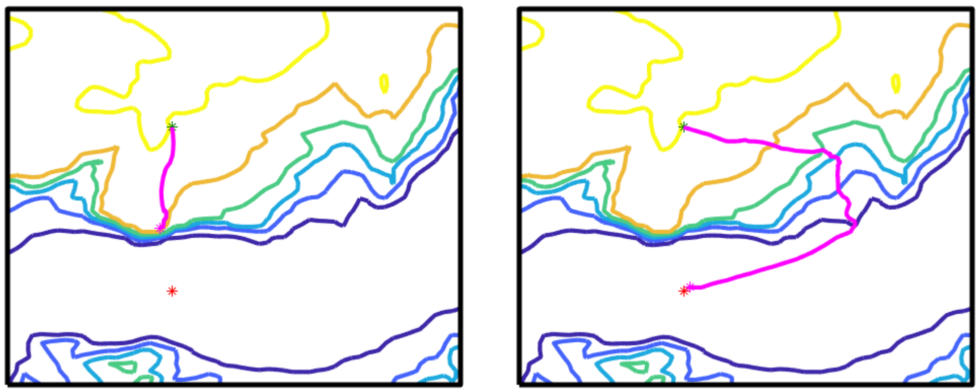
(a) $\sigma = 0.05$ (b) $\sigma = 0.1$ (c) $\sigma = 0.2$

(d) $\sigma = 0.05$ (e) $\sigma = 0.1$ (f) $\sigma = 0.2$

Fig. 5: (a) - (c) Average path over 10000 trials (black), and three realizations of the stochastic equation of motion (colored) at different levels of uncertainty σ. (d) - (f) Average path (black) with standard deviation (grey), and the path computed using method (i) (dotted green).

4.3. Impassable Obstacles and the Role of the Parameter T. We remarked in section 2.1 that different choices for the terminal time T can lead to qualitative changes in how the path is constructed. We can observe this in the example of El

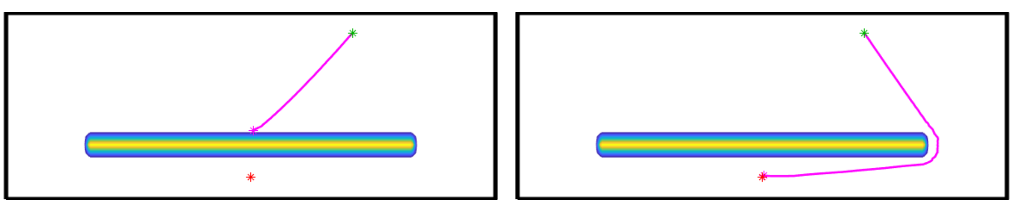
Capitan. In figure 6, we used $\sigma = 0$ so that the model is fully deterministic, and we simulated the model with two different terminal times T. In figure 6a, we see that with terminal time $T \approx 2000$ seconds, the path simply walks to the cliff face and stays put. However, given $T \approx 6800$ seconds, the path descends the eastern slope and finds the desired end point as seen in figure 6b



(a) Optimal path given $T = 2000$ seconds.   (b) Optimal path given $T = 6800$ seconds.

Fig. 6: Optimal paths in the vicinity of El Capitan with different terminal times.

We can recreate this scenario using synthetic elevation data by placing a large wall directly between the starting point and end point as in figure 7. The elevation is incredibly steep in the colored region, meaning that any optimal path would surely avoid the wall. In figure 7b, where $T = 4.25$, this is exactly the behavior we observe; the path curves around the obstacle. However, in figure 7a, where $T = 2$, the path walks toward the obstacle, stopping at the edge because velocity is near zero there.



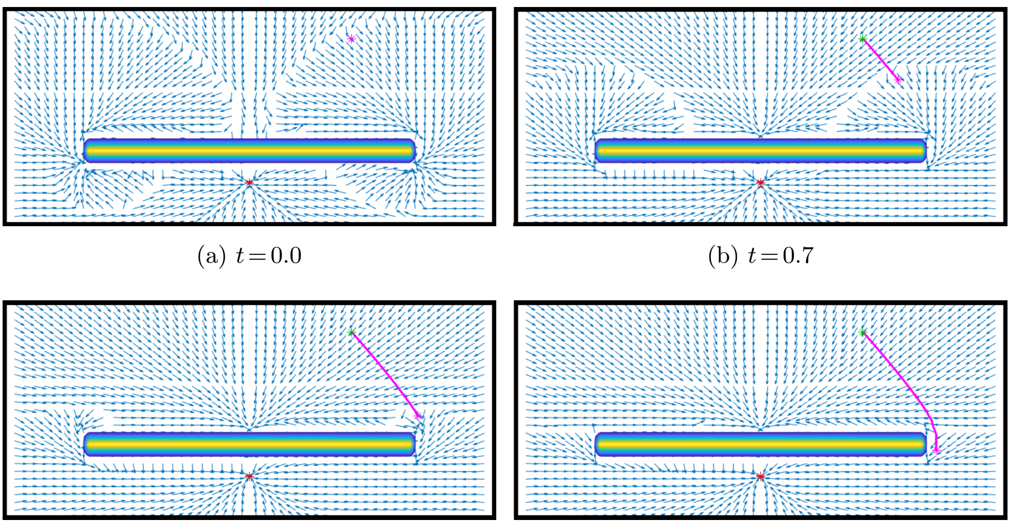
(a) Optimal path given $T = 2$ seconds.   (b) Optimal path given $T = 4.25$ seconds.

Fig. 7: Optimal paths using different end time values. The colored region is a wall.

Recall, our model constructs the path which ends as close (in Euclidean distance) to the desired end point as possible in the time allotted. When $T = 2$ in figure 7a, there is not enough time to walk around the wall and instead, to get as close to the desired end point as possible, the path walks directly toward the wall. When situations like this arise, there is some critical amount of time $T^* > 0$ such that, given $T > T^*$, the path will walk around the obstacle, but given $T < T^*$, the path will walk toward the obstacle because it will not be able to get close enough to the end point if it walks around.

We can see this more explicitly if we plot the actual control values $\boldsymbol{s}^*(x,t)$ as well, as is done in figure 8. In this example, the critical time is roughly $T^* = 3.4$, so we have plotted the path created by the algorithm with final time of $T = 3.5$, but we have plotted the path at times $t = 0, 0.7, 1.4, 2.1$. The arrows in the pictures are the values of $\boldsymbol{s}^*(x,t)$. Notice that in figure 8a, there appears to be a discontinuity in the optimal control value. The deciding factor for whether the path will walk around the wall or walk toward the wall is where the starting point lies relative to this discontinuity. As time advances, the discontinuity in $\boldsymbol{s}^*(x,t)$ propagates, and since the starting point lies below the discontinuity, the path follows the arrows and walks around the obstacle. In the case when $T = 2$, the starting point is above the discontinuity, and thus the path walks toward the obstacle, rather than around it.

Discontinuities in $\boldsymbol{s}^*(x,t)$ are to be expected and relate to non-uniqueness of the optimal path. If x_0 lay directly on the discontinuity in $\boldsymbol{s}^*(x,0)$, then either walking around the obstacle or toward it would be equally optimal, since both would result in a path that ends the same distance from the desired end point. Mathematically, one reason to expect discontinuities in $\boldsymbol{s}^*(x,t)$ is because $\boldsymbol{s}^*(x,t)$ is closely related to the gradient $\nabla u(x,t)$ of the solution to the HJB equation. Indeed, in the case of isotropic



(a) $t = 0.0$

(b) $t = 0.7$

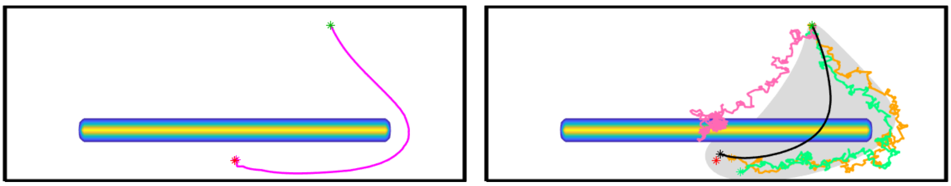
(c) $t = 1.4$

(d) $t = 2.1$

Fig. 8: The discontinuity in the control value $\boldsymbol{s}^*(x,t)$ propagates as time increases

motion (that is, when the equation of motion has the form $\dot{\boldsymbol{x}} = f(\boldsymbol{x})\boldsymbol{s}$), we will have $\boldsymbol{s}^*(x,t) = -\nabla u(x,t)/\|\nabla u(x,t)\|_2$. When the motion is anisotropic, as is the case in our model, the relationship between $\boldsymbol{s}^*(x,t)$ and $\nabla u(x,t)$ is no longer so explicit, but we can still anticipate that discontinuities in $\nabla u(x,t)$ will give rise to discontinuities in $\boldsymbol{s}^*(x,t)$, and as stated above, we expect the solutions of Hamilton-Jacobi equations to have discontinuities in their derivatives [4, 11, 32].

When we add uncertainty, the path planning strategy becomes more greedy, walking directly toward the end point and adjusting to avoid obstacles as is seen in figure 1. When there is a wall, this strategy is costly because if one walks toward the wall, there may be insufficient time to adjust the route, and thus the critical time T^* required to walk around the wall increases rapidly. This is why the large increase in T is necessary in the example of El Capitan in figure 4. We observe the same behavior in this synthetic example with the wall, though the increase in T as not as pronounced as in the case of El Capitan. In figure 9a, we set $\sigma = 0.3$ and notice that to wrap around the wall and reach the end point, the optimal path computed using method (i) requires an end time of $T = 4.75$ rather than $T = 4.25$ in the deterministic case. In figure 9b, we use method (ii), computing the average path over 10000 trials, and the path cuts through the wall since enough individual paths were pushed off course due to the random perturbations, as is seen with the pink path in the figure. As in figure 5f, an individual could not realistically traverse the average path, since the wall is impassable. Thus, it seems that method (i) gives to a more practical result.



(a) Method (i), $\sigma = 0.3$, $T = 4.75$.

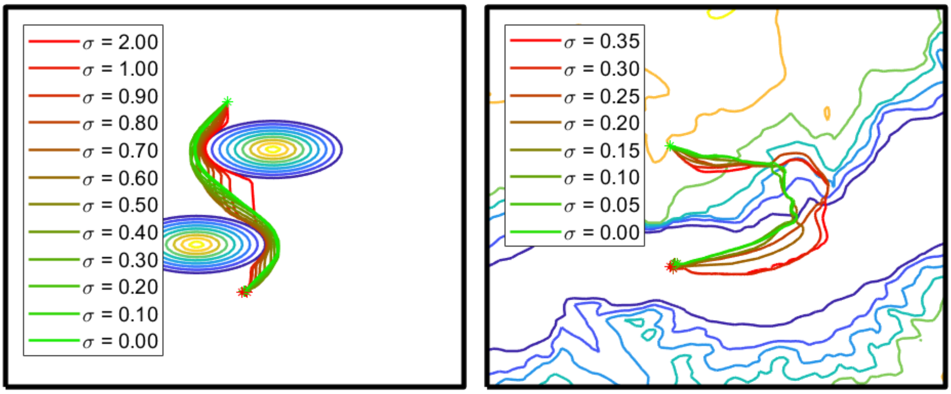
(b) Method (ii), average path, standard deviation and three realizations.

Fig. 9: Optimal paths around the wall with uncertainty.

The dependence of the model on the parameter T is a major qualitative difference between this optimal path planning model and the model presented in [37]. That model deals only with the fully deterministic case, and uses a level set formulation wherein level sets representing optimal travel evolve outward from the starting point, and the terminal time T is defined as the time required for the level sets to envelop the end point. However, when we add uncertainty to the model, we introduce diffusion in the HJB equation and lose the level set intepretation of the equation. Thus while the model in [37] has the advantage of not depending on T, this model is more generally applicable.

4.4. Convergence to the Deterministic Path as $\sigma \searrow 0$. As stated in section 2.3, given mild regularity conditions on our Hamiltonian, the solution to the stochastic HJB equation (2.15) will converge to the viscosity solution to the ordinary HJB equation (2.4) as $\sigma \searrow 0$ [11]. We can see this empirically, not by observing the solution itself, but by examining the optimal path suggested by our algorithm at different levels of σ. This is shown in figure 10. Here we have plotted many paths on the same figure, each computed using method (i) with a different σ value. As before, paths plotted in green were computed using smaller σ values, and those in red were computed using larger σ values. In figure 10a, we see a very clear color gradient: the red paths computed with larger σ clearly tend toward the green path as σ decreases to zero. In figure 10b, this is less obvious, especially since, for larger σ, the path takes a qualitatively different route. However, we do see that for smaller σ (greener paths), there is a tendency toward the deterministic optimal path.

5. Conclusions. Path planning algorithms have wide-reaching applications in self-driving vehicles, reach-avoid games, pedestrian flow modeling and many other areas. Many previous models for path planning are completely deterministic, while in reality stochastic effects may be present and can significantly alter the motion along the path.



(a) Convergence of paths with synthetic elevation data.

(b) Convergence of paths at El Capitan.

Fig. 10: As $\sigma \searrow 0$, the stochastic optimal path converges back to the deterministic optimal path.

In this paper we have developed a method for optimal path planning of human walking paths in mountainous terrain using a control theoretic approach and a Hamilton-Jacobi-Bellman (HJB) equation and allowing for uncertainty in the controlled equation of motion. The walking speed in our model depends on local slope in the direction of travel, giving rise to an anisotropic control problem. In the HJB equation, the uncertainty presents itself in the form of diffusion, leading to a viscous Hamilton-Jacobi-type equation. We suggest numerical methods for solving these equations, opting for a semi-implicit numerical scheme with a minimally diffusive numerical Hamiltonian, since any spurious numerical diffusion could be intepreted as nontrivial amounts of uncertainty in the equation of motion. After solving the HJB equation numerically, we suggest two methods for resolving the optimal path. First, we use the optimal control values resolved via the stochastic HJB equation, but simulate a deterministic equation of motion. This could simulate a person walking through a dark room or a dense forest, wherein they are cognizant of some uncertainty as they are planning the route, but do not feel random perturbations in velocity as they walk along a path. Second, we integrate the stochastic differential equation many times and arrive at a single path by averaging the results. This could model scenarios such as underwater unmanned vehicles, wherein the traveler actually feels the stochastic effects on the travel velocity.

We test our algorithm, including both methods for resolving the path, with synthetic elevation data first, and then with real elevation data in the area surrounding El Capitan. We compare these two notions of optimal path, and conclude that in the case of real elevation data or impassable barriers, the first notion gives a more meaningful result. We also observe that in these cases, there will be discontinuities in the optimal control parameter and, especially in the presence of large walls, the position of these discontinuities can determine the walking strategy. Finally, we simulate the model at different levels of uncertainty in the equation of motion and observe that as uncertainty tends to zero, the optimal path path suggested by the model converges back to the deterministic optimal path.

6. Acknowledgements. Financial support for this work was furnished by the Department of Defense and the NSF (DMS-1737770).

The elevation data for Yosemite National Park was downloaded from the United States Geological Survey national map viewer [50]. The data was processed and reformatted using QGIS, an open source geological data processing tool [39]. The data was imported into MATLAB using TopoToolbox [41, 42].

References.

[1] P. K. Agarwal and H. Wang. Approximation algorithms for curvature-constrained shortest paths. *SIAM Journal on Computing*, 30(6):1739–1772, 2001.
[2] D. J. Arnold, D. Fernandez, R. Jia, C. Parkinson, D. Tonne, Y. Yaniv, A. L. Bertozzi, and S. J. Osher. Modeling environmental crime in protected areas using the level set method. *SIAM Journal on Applied Mathematics*, 79(3):802–821, 2019.
[3] A. Aurnhammer and K. Marti. Adaptive optimal stochastic trajectory planning. In M. Grötschel, S. O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems*, pages 521–543. 2001.
[4] M. Bardi and I. Capuzzo-Dolcetta. Continuous viscosity solutions of Hamilton-Jacobi equations. In *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, pages 25–96. Birkhäuser Boston, Boston, MA, 1997. ISBN 978-0-8176-4755-1.
[5] A. Bayen, I. M. Mitchell, M. Oishi, and C. J. Tomlin. Aircraft autolander safety analysis through optimal control-based reach set computation. In *Journal of Guidance Control and Dynamics*, volume 30, pages 68–77. 01 2007.
[6] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2 edition, 2000. ISBN 1886529094.
[7] R. Carmona and F. Delarue. *Probabilistic Theory of Mean Field Games with Applications I*. Springer International Publishing, 1 edition, 2018. ISBN 978-3-319-56437-1.
[8] E. Cartee and A. Vladimirsky. Anisotropic challenges in pedestrian flow modeling. *Communications in Mathematical Sciences*, 16(4):1067–1093, 10 2018.
[9] E. Cartee, L. Lei, Q. Song, and A. Vladimirsky. Time-dependent surveillance-evasion games. Preprint. URL https://arxiv.org/abs/1903.01332.
[10] T. Cecil and D. Marthaler. A variational approach to path planning in three dimensions using level set methods. *Journal of Computational Physics*, 211:179–197, 01 2005.
[11] M. G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton-Jacobi equations. *Transactions of the American Mathematical Society*, 277(1):1–42, 1983.
[12] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numer. Math.*, 1(1):269–271, Dec. 1959.
[13] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.
[14] R. P. Fedkiw, G. Sapiro, and C.-W. Shu. Shock capturing, level sets, and pde based methods in computer vision and image processing: a review of Osher's contributions. *Journal of Computational Physics*, 185(2):309 – 341, 2003.
[15] W. H. Fleming and R. W. Rishel. *Deterministic and Stochastic Optimal Control*. Springer-Verlag New York, 1 edition, 1975. ISBN 978-0-387-90155-8.
[16] M. A. Gilles and A. Vladimirsky. Surveillance-evasion games under uncertainty. Preprint. URL https://arxiv.org/abs/1812.10620.
[17] M. Hardt, K. Kreutz-Delgado, and J. W. Helton. Optimal biped walking with

a complete dynamical model. In *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, volume 3, pages 2999–3004, 1999.

[18] C. Hirsch, D. Neuhuser, C. Gloaguen, and V. Schmidt. Asymptotic properties of Euclidean shortest-path trees in random geometric graphs. *Statistics & Probability Letters*, 107:122 – 130, 2015.

[19] I. J. Irmischer and K. C. Clarke. Measuring and modeling the speed of human navigation. *Cartograpghy and Geographic Information Science*, 45(2):177–186, 2017.

[20] C. Y. Kao, S. Osher, and J. Qian. Lax–Friedrichs sweeping scheme for static Hamilton–Jacobi equations. *Journal of Computational Physics*, 196(1):367–391, 2004.

[21] P. E. Kloeden and E. Platen. Strong taylor approximations. In *Numerical Solution of Stochastic Differential Equations*, pages 339–371. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992. ISBN 978-3-662-12616-5.

[22] G. Krishnaswamy and A. Stentz. Resolution independent grid-based path planning. Technical report, Carnegie-Mellon University Robotics Institute, 1995.

[23] S. Lim and D. Rus. Stochastic motion planning with path constraints and application to optimal agent, resource, and route planning. In *2012 IEEE International Conference on Robotics and Automation*, pages 4814–4821, 2012.

[24] T. Lolla, P. H. Jr., and P. Lermusiaux. Path planning in multi-scale ocean flows: Coordination and dynamic obstacles. *Ocean Modelling*, 94:46 – 66, 2015.

[25] R. Luna, M. Lahijanian, M. Moll, and L. E. Kavraki. Fast stochastic motion planning with optimality guarantees using local policy reconfiguration. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3013–3019, 2014.

[26] J. Lygeros. On reachability and minimum cost optimal control. *Automatica*, 40(6): 917 – 927, 2004.

[27] K. Marti. Path planning for robots under stochastic uncertainty. *Optimization*, 45 (1-4):163–195, 1999.

[28] K. Marti. Adaptive optimal stochastic trajectory planning and control. In *Stochastic Optimization Methods: Applications in Engineering and Operations Research*, pages 119–194. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[29] K. R. Memon, S. Memon, B. Memon, A. R. Memon, and S. M. Z. A. Shah. Real time implementation of path planning algorithm with obstacle avoidance for autonomous vehicle. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 2048–2053, March 2016.

[30] J. S. Mitchell and M. Sharir. New results on shortest paths in three dimensions. In *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, pages 124–133. ACM, 2004.

[31] M. Nisio. *Stochastic Control Theory*. Springer Japan, 2 edition, 2009. ISBN 978-4-431-55122-5.

[32] S. Osher and R. P. Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153 of *Applied Mathematical Sciences*. Springer–Verlag, 2003.

[33] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[34] S. Osher and C.-W. Shu. High order essentially non–oscillatory schemes for Hamilton–Jacobi equations. *SIAM Journal of Numerical Analysis*, 28(4):907–922, August 1991.

[35] C. H. Papadimitriou. An algorithm for shortest-path motion in three dimensions.

Information Processing Letters, 20(5):259 – 263, 1985.
[36] A. V. Papadopoulos, L. Bascetta, and G. Ferretti. Generation of human walking paths. *Autonomous Robots*, 40(1):59–75, Jan 2016.
[37] C. Parkinson, D. Arnold, A. Bertozzi, Y. Chow, and S. Osher. Optimal human navigation in steep terrain: a Hamilton-Jacobi-Bellman approach. *Communications in Mathematical Sciences*, 17:227–242, 01 2019.
[38] H. Pham. *Continuous-time Stochastic Optimal Control and Optimization with Financial Applications*. Springer-Verlag Berlin Heidelberg, 1 edition, 2009. ISBN 978-3-540-89499-5.
[39] Quantum GIS Development Team. Quantum GIS geographic information system, 2017. URL http://qgis.osgeo.org.
[40] C. Saranya, M. Unnikrishnan, S. A. Ali, D. Sheela, and D. V. Lalithambika. Terrain based D* algorithm for path planning. *IFAC-PapersOnLine*, 49(1):178 – 182, 2016. 4th IFAC Conference on Advances in Control and Optimization of Dynamical Systems ACODS 2016.
[41] W. Schwanghart and N. Kuhn. TopoToolbox: a set of MATLAB functions for topographic analysis. *Environmental Modelling & Software*, 25(6):770–781, 2010.
[42] W. Schwanghart and D. Scherler. TopoToolbox 2: MATLAB-based software for topographic analysis and modeling in Earth surface sciences. *Earth Surface Dynamics*, 2(1):1–7, 2014.
[43] J. A. Sethian and A. Vladimirsky. Ordered upwind methods for static Hamilton–Jacobi equations. *Proceedings of the National Academy of Sciences*, 98(20): 11069–11074, 2001.
[44] J. A. Sethian and A. Vladimirsky. Ordered upwind methods for static Hamilton–Jacobi equations: Theory and algorithms. *SIAM Journal on Numerical Analysis*, 41(1):325–363, 2003.
[45] Z. Shen and A. Vladimirsky. Piecewise-deterministic optimal path planning (preprint). *submitted to Journal of Optimization Theory and Applications*. URL https://arxiv.org/abs/1512.08734.
[46] W. Song and S. J. Dyke. Application of pseudospectral method in stochastic optimal control of nonlinear structural systems. In *Proceedings of the 2011 American Control Conference*, pages 2504–2509, June 2011.
[47] D. N. Subramani and P. F. Lermusiaux. Energy-optimal path planning by stochastic dynamically orthogonal level-set optimization. *Ocean Modelling*, 100:57 – 77, 2016.
[48] D. N. Subramani, Q. J. Wei, and P. F. Lermusiaux. Stochastic time-optimal path-planning in uncertain, strong, and dynamic flows. *Computer Methods in Applied Mechanics and Engineering*, 333:218 – 237, 2018.
[49] R. Takei, R. Tsai, H. Shen, and Y. Landa. A practical path-planning algorithm for a simple car: a Hamilton-Jacobi approach. In *Proceedings of the 2010 American Control Conference*, pages 6175–6180, June 2010.
[50] U.S. Geological Survey. The National Map, 2017, 3DEP products and services: The National Map, 3D Elevation Program Web page: https://nationalmap.gov/3DEP/3dep_prodserv.html.
[51] Z. Zhou, J. Ding, H. Huang, R. Takei, and C. Tomlin. Efficient path planning algorithms in reach-avoid problems. *Automatica*, 89:28 – 36, 2018.