

# DISCRETE APPROXIMATIONS TO CONTINUOUS SHORTEST-PATH: APPLICATION TO MINIMUM-RISK PATH PLANNING FOR GROUPS OF UAVS\*

Jongrae Kim      João P. Hespanha

Dept. Electrical and Computer Engineering, Univ. of California

Santa Barbara, CA 93106-9560

Tel: +1 (805) 893-7042    Fax: +1 (805) 893-3262

Email: {jongrae,hespanha}@ece.ucsb.edu

March 8, 2003

## Abstract

This paper addresses the weighted anisotropic shortest-path problem on a continuous domain, i.e., the computation of a path between two points that minimizes the line integral of a cost-weighting function along the path. The cost-weighting depends both on the instantaneous position and direction of motion. We propose an algorithm for the computation of shortest path that reduces the problem to an optimization over a finite graph that can be efficiently solved.

The resulting path is not optimal but the cost penalty can be made arbitrarily small at the expense of increased computation. The algorithm proposed restricts the search to paths formed by the concatenation of straight-line segments between points from a suitably chosen discretization of the continuous region. To maximize efficiency, the discretization should not be uniform. We propose a novel “honeycomb” sampling algorithm that minimizes the cost penalty introduced.

This methodology is applied to the computation of paths for groups of Unmanned Air Vehicles (UAVs) that minimize the risk of being destroyed by ground defenses. We show that this problem can be formulated as a shortest path optimization and show that the algorithm proposed can efficiently produce low-risk paths.

## 1 Introduction

Consider a compact region  $\mathcal{R} \subset \mathbb{R}^n$  and two points  $x_i, x_f \in \mathcal{R}$ . Our goal is to compute a continuous path  $\rho$  from  $x_i$  to  $x_f$  that minimizes the line integral over  $\rho$  of a cost-weighting function  $\ell$  that is position and direction of motion dependent. Motivated by the observation that the optimal path is a straight-line from  $x_i$  to  $x_f$  when the cost-weighting is constant, one often refers to this problem as a *shortest-path* optimization. To emphasize the fact that the cost-weighting is not uniform and that

---

\*This paper is based upon work supported by DARPA under the Space and Naval Warfare Systems Center, San Diego, Contract Number N66001-01-C-8076. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA or the Space and Naval Warfare Systems Center, San Diego.

it depends on the direction of motion, we further qualify it as a *weighted anisotropic shortest-path* optimization.

To formalize the problem, we denote by  $\mathcal{P}$  the set of all unit-speed paths in  $\mathcal{R}$  from  $x_i$  to  $x_f$  that are continuous and piecewise twice continuously differentiable, i.e., the set of continuous functions  $\rho : [0, T] \rightarrow \mathcal{R}$ ,  $T > 0$  for which (i)  $\rho(0) = x_i$  and  $\rho(T) = x_f$ ; (ii)  $\dot{\rho}$  and  $\ddot{\rho}$  exist on  $[0, T]$ , except for a finite number of points; and (iii)  $\|\dot{\rho}\| = 1$  wherever this derivative exists. The problem under consideration can be formalized as follows.

**Problem 1 (Weighted anisotropic shortest-path).** Compute a path  $\rho^* \in \mathcal{P}$  such that<sup>1</sup>

$$J[\rho^*] = \min_{\rho \in \mathcal{P}} J[\rho],$$

where  $J : \mathcal{P} \rightarrow [0, \infty)$  denotes the functional defined by

$$J[\rho] := \int_0^T \ell(\rho(t), \dot{\rho}(t)) dt,$$

for each  $\rho : [0, T] \rightarrow \mathcal{R}$  in  $\mathcal{P}$ . □

We assume throughout the paper that the cost-weighting  $\ell : \mathcal{R} \times \mathbb{R}^n \rightarrow [0, \infty)$  is continuously differentiable.

The solution to this problem has numerous applications that range from mobile robotics to path planning on topographical maps. The specific application pursued here is the computation of paths for groups of Unmanned Air Vehicles (UAVs) that minimize the risk of being destroyed by ground defenses.

The computation of shortest paths has a long history and is, in fact, the most basic problem in Calculus of Variations (cf., e.g., [8]). Assuming for simplicity that  $\mathcal{R} := \mathbb{R}^n$ , the optimization formulated above is equivalent to the optimal control problem of finding a terminal time  $T \geq 0$  and a control  $v : [0, T] \rightarrow \mathcal{V}$ , with  $\mathcal{V} := \{z \in \mathbb{R}^n : \|z\| = 1\}$  that minimizes

$$J := \int_0^T \ell(x(t), v(t)) dt,$$

subject to the dynamics  $\dot{x} = v$  and initial and terminal conditions  $x(0) = x_i$ ,  $x(T) = x_f$ . The solution to this problem can be found using the Hamilton-Jacobi-Bellman (HJB) equation: Assuming that there exists a continuously differentiable solution  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  to the HJB equation

$$0 = \min_{v \in \mathcal{V}} H(x, v, \nabla_x V(x)), \quad \forall x \in \mathbb{R}^n,$$

with boundary condition  $V(x_f) = 0$ , where the Hamiltonian  $H$  is defined by

$$H(x, v, p) = \ell(x, v) + \langle p, v \rangle, \quad \forall x, v, p \in \mathbb{R}^n,$$

the optimal control  $v^* : [0, T^*] \rightarrow \mathcal{V}$  is given by

$$v^*(t) = \arg \min_{v \in \mathcal{V}} H(x^*(t), v, \nabla_x V(x^*(t))),$$

---

<sup>1</sup>Without further assumptions, the minimum may not be achieved for a path in  $\mathcal{P}$ , but we will ignore this for now.

where  $x^*$  denotes the optimal trajectory defined by  $\dot{x}^* = v^*$  and  $x^*(0) = x_i$ . However, there are several problems with this approach: It generally fails when the HJB equation (or a relaxation of it) has no continuously differentiable solutions. Moreover, even when an appropriate solution exists (perhaps only a viscosity solution), it is generally computationally very difficult to find it.

Several methods have been proposed to overcome this difficulty. These methods typically explore special structures for the cost-weighting  $\ell$  or/and compute paths that are only approximately optimal. We pursue here the latter approach and solve a discretized version of the continuous problem, which provides an approximate solution to our original problem. This approximation yields solutions that are not necessarily optimal but whose cost can be made arbitrarily close to the optimal one. We start by sampling the region  $\mathcal{R}$  to extract a finite number of points  $\mathcal{X}$  and then restricting the search to paths consisting of the concatenation of line segments between points in  $\mathcal{X}$ . The search portion of the algorithms is performed by constructing a finite graph (whose nodes are the points in  $\mathcal{X}$ ), where one then solves a shortest-path problem using standard algorithms.

This type of approach has been proposed before (cf., Section 2) but because of the “curse of dimensionality” its successful application to nontrivial problems depends crucially on the algorithm used to sample the region  $\mathcal{R}$ . This paper addresses precisely this issue. We start by deriving in Section 3 a worst-case bound for the cost penalty introduced by discretization. Inspired by this bound, we then propose in Section 4 a novel efficient algorithm to sample  $\mathcal{R}$  so as to achieve a small cost penalty with a relatively sparse sampling of  $\mathcal{R}$ . The key idea is to sample  $\mathcal{R}$  so that there will be more sample points in regions where the optimal path is more likely to deviate from a straight-line.

We apply the proposed algorithm to the computation of paths for groups of UAVs, which minimize their probability of destruction by ground defenses. This is an inherently three-dimensional, anisotropic shortest-path problem for which previous algorithms do not apply. The results obtained, which are summarized in Section 5, validate the algorithm and show that it can solve the problem with reasonable computational effort. In this section, we also compare the performance of several alternative sampling methods.

## 2 Related work

The solution to shortest path problems is an active area of research in computational geometry [20]. When strong assumptions on the cost-weighting  $\ell$  are imposed, efficient algorithms can be used to compute the shortest path. Often  $\ell(x, v)$  is assumed independent of the velocity  $v$  and takes only two distinct values: a low value corresponding to “free-space” and a high value (in the limit  $+\infty$ ) corresponding to “obstacles.” Hershberger and Suri [9] proposed an algorithms for planar regions and obstacle spaces defined by polygons that runs in worst-case time  $O(n \log n)$  and requires  $O(n \log n)$  space, where  $n$  is the total number of obstacle vertices. Another algorithm that is competitive when the number  $m$  of disconnected obstacles is much smaller than the number of vertices has been proposed by Kapoor et al. [10] and runs in worst-case time  $O(n + m^2 \log n)$  and requires  $O(n)$  in space. The complexity of the problem increases when the cost-weighting  $\ell(x, v)$  is still independent of the velocity  $v$  and piecewise constant over space, but can take more than two values. However, polynomial-time algorithms can still be found that produce  $\epsilon$ -optimal solutions, i.e. paths for which the cost is above the minimum by a factor no larger than  $(1 + \epsilon)$ . E.g., Mitchell and Papadimitriou [15], Aleksandrov et al. [1], and Mata and Mitchell [13] provide  $\epsilon$ -optimal planar algorithms for any  $\epsilon > 0$  that run in worst-case times,  $O(n^8 \log \frac{n}{\epsilon})$ ,  $O(\frac{n}{\epsilon} \log \frac{1}{\epsilon} (\frac{1}{\sqrt{\epsilon}} + \log n))$ , and  $O(\frac{n^3}{\epsilon})$ ,

respectively. In three dimensions the complexity of the problem also increases but polynomial-time algorithms can still be found that produce  $\epsilon$ -optimal solutions. E.g., Aleksandrov et al. [1] provide an  $\epsilon$ -optimal algorithm for any  $\epsilon > 0$  that runs in worst-case time  $O(\frac{n}{\epsilon} \log \frac{1}{\epsilon} (\frac{1}{\sqrt{\epsilon}} + \log n))$ , where  $n$  is the number of convex regions needed to define the piecewise constant cost-weight  $\ell$ . The worst-case time decreases to  $O(\frac{n}{\epsilon^3} \log \frac{1}{\epsilon} \log n)$  when  $\ell$  only takes two very distinct values. The reader is referred to the survey by Mitchell [14] for additional results.

Much less work has been devoted to the anisotropic case, i.e., when  $\ell(x, v)$  depends both on the position  $x$  and velocity  $v$ . Rowe and Ross [19] considered the computation of the minimum-energy path for a ground vehicle moving between two points on a hilly terrain described by a triangular mesh (2 1/2-dimensional case). Rowe and Ross [19] used a simple model for the energy-cost that takes into account the grade of the climb and is therefore velocity dependent. They provided an exact algorithm with worst-time run complexity  $O(n^n)$ . Lanthier et al. [12] later provided an approximate algorithm with polynomial worst-case time.

The numerical solution to stochastic shortest-path problems has also been addressed in the control literature, where most of the focus has been on approximations to the HJB formulation. The reader is referred to [11] for a detailed discussion on this topic. Tsitsiklis [22] proposes two efficient algorithms to solve an isotropic shortest path problem by solving a discretized HJB equation. By restricting their attention to isotropic cost-weights and sampling  $\mathcal{R}$  using a regular rectangular grid, the author constructs an algorithm whose complexity is linear on the number of sample points. An even more efficient algorithm for the same optimization is proposed on [6]. However, none of these algorithms seems to generalize to the anisotropic case [22].

A fundamentally different approximation to the HJB formulation was proposed in [21] for the anisotropic case when  $\ell(x, v)$  only takes two distinct values:  $+\infty$  on a finite collection of non-intersecting ‘‘obstacles’’ and a fixed constant value elsewhere. The authors compute the solution to the HJB exactly for a single disk-like obstacle and then an approximation of it for the case of multiples obstacles, which is constructed considering the effect of one obstacle at a time. The method can be efficiently implemented and the paths generated are near-optimal and guaranteed to reach the goal. This type of approximation depends critically on the binary nature of  $\ell$ .

### 3 Discretization

To overcome the difficulties that arise in solving the Weighted Anisotropic Shortest-Path Problem 1 exactly, one may finely grid the region  $\mathcal{R}$  and force the path to consist of the concatenation of several straight segments between points in the grid. For a finite number of grid points, this procedure converts the original continuous shortest-path problem into a shortest-path problem on a finite graph.

Given a finite set of points  $\mathcal{X}$  that includes  $x_i$  and  $x_f$ , we say that a path  $\rho : [0, T] \rightarrow \mathcal{R}$  in  $\mathcal{P}$  is *piecewise linear with respect to  $\mathcal{X}$*  if there exists a sequence of points

$$\{x_0 = x_i, x_1, x_2, \dots, x_m = x_f\} \subset \mathcal{X}$$

such that

$$\rho(t) = x_{k-1} + \frac{x_k - x_{k-1}}{\|x_k - x_{k-1}\|} (t - t_{k-1}), \quad \forall t \in [t_{k-1}, t_k], \quad k \in \{1, 2, \dots, m\}, \quad (1)$$

where the  $t_k$  are defined recursively by

$$t_0 := 0, \quad t_k = t_{k-1} + \|x_k - x_{k-1}\|, \quad \forall k \in \{1, 2, \dots, N\},$$

and  $T = t_m$ . The set of all such paths is denoted by  $\mathcal{P}_{\mathcal{X}}$ . The following results show that restricting our search to this type of paths can provide solutions arbitrarily close to the optimal one.

**Theorem 1.** *For any  $A, \epsilon > 0$ , there exists a finite set  $\mathcal{X}$  of  $N_\epsilon$  points in  $\mathcal{R}$  such that for every initial and finite points  $x_i, x_f \in \mathcal{R}$*

$$\inf_{\rho \in \mathcal{P}_{\bar{\mathcal{X}}}} J[\rho] - \epsilon \leq \inf_{\rho \in \mathcal{P}_{\|\ddot{\rho}\| \leq A}} J[\rho] \leq \inf_{\rho \in \mathcal{P}_{\bar{\mathcal{X}}}} J[\rho],$$

where  $\bar{\mathcal{X}} := \mathcal{X} \cup \{x_i, x_f\}$  and  $\mathcal{P}_{\|\ddot{\rho}\| \leq A}$  denotes the set of twice continuously differentiable paths in  $\mathcal{P}$  with second derivative bounded by  $A$ .

The proof of Theorem 1 is constructive and will inspire the construction of sampling algorithms that attempt to minimize the worst-case cost penalty  $\epsilon$  introduced by discretization. The following two technical results are needed to prove Theorem 1. The proofs of these results can be found in the Appendix.

**Lemma 1.** *Given any positive constants  $A, \epsilon_x, \epsilon_v, \epsilon_\delta$  (without loss of generality assuming that  $\epsilon_x \leq \epsilon_\delta$ ), there exists a finite subset  $\mathcal{X}$  of  $\mathcal{R}$  such that for every twice continuously differentiable path  $\rho : [0, T] \rightarrow \mathcal{R} \in \mathcal{P}$  with second derivative bounded by  $A$ , one can find*

1. a sequence of times  $\{\tau_0, \tau_1, \dots, \tau_N\} \subset [0, T]$ , with  $\tau_0 := 0 \leq \tau_{k-1} < \tau_k \leq \tau_N := T$ , and
2. a sequence of points  $\{x_0, x_1, \dots, x_N\} \in \mathcal{X}$

such that

$$\|x_k - x_{k-1}\| \leq \epsilon_\delta, \quad \forall k \in \{1, \dots, N\}, \quad (2)$$

$$\|\rho(t) - x_{k-1}\| \leq \epsilon_\delta, \quad \forall t \in [\tau_{k-1}, \tau_k], \quad k \in \{1, \dots, N\}, \quad (3)$$

$$\|\rho(\tau_k) - x_k\| \leq \epsilon_x, \quad \forall k \in \{0, 1, \dots, N\}, \quad (4)$$

$$\left\| \dot{\rho}(\tau) - \frac{x_k - x_{k-1}}{\|\tau - \tau_{k-1}\|} \right\| \leq \epsilon_v, \quad \forall \tau \in (\tau_{k-1}, \tau_k), \quad k \in \{1, \dots, N\}, \quad (5)$$

$$\sum_{k=1}^N \|x_k - x_{k-1}\| \leq \frac{1 + \epsilon_v}{2 + \epsilon_v} 2T. \quad (6)$$

Moreover, as  $\epsilon_x \downarrow 0$ , the integer  $N$  remains uniformly bounded.

**Lemma 2.** *Given two paths  $\rho_i : [0, T_i] \rightarrow \mathcal{R}$  in  $\mathcal{P}$  and two intervals  $(t_i, \tau_i) \subset [0, T_i]$ ,  $i \in \{1, 2\}$  on which the corresponding  $\rho_i$  are twice continuously differentiable,*

$$\begin{aligned} & \int_{t_2}^{\tau_2} \ell(\rho_2(t), \dot{\rho}_2(t)) dt - \int_{t_1}^{\tau_1} \ell(\rho_1(t), \dot{\rho}_1(t)) dt \\ & \leq |\ell(\rho_2(t_2), \dot{\rho}_2(t_2^+)) - \ell(\rho_1(t_1), \dot{\rho}_1(t_1^+))| \Delta + |\ell(\rho_2(\tau_2), \dot{\rho}_2(\tau_2^-)) - \ell(\rho_1(\tau_1), \dot{\rho}_1(\tau_1^-))| \Delta \\ & \quad + (\tau_2 - t_2 - 2\Delta)h + (g_{x,1} + g_{x,2} + a_1g_{v,1} + a_2g_{v,2})\Delta^2 \quad (7) \end{aligned}$$

where<sup>2</sup>

$$\Delta := \frac{1}{2} \min \{ \|\rho_1(\tau_1) - \rho_1(t_1)\|, \|\rho_2(\tau_2) - \rho_2(t_2)\| \}, \quad h := \sup_{s \in (t_2 + \Delta, \tau_2 - \Delta)} \ell(\rho_2(s), \dot{\rho}_2(s))$$

$$g_{(\cdot), i} := \sup_{s \in (t_i, \tau_i)} \|\nabla_{(\cdot)} \ell(\rho_i(s), \dot{\rho}_i(s))\|, \quad a_i := \sup_{s \in (t_i, \tau_i)} \|\ddot{\rho}_i(s)\|, \quad i \in \{1, 2\}, \quad (\cdot) = x \text{ or } v.$$

The inequality (7) still holds if we replace the term  $(g_{x,1} + g_{x,2} + a_1 g_{v,1} + a_2 g_{v,2})\Delta^2$  by  $(g_{x,1} + g_{x,2})\Delta^2 + 2(g_{v,1} + g_{v,2})\Delta$ .

*Proof of Theorem 1.* Let  $\mathcal{X}$  be the set whose existence is guaranteed by Lemma 1, for parameters  $\epsilon_x, \epsilon_v, \epsilon_\delta$  to be defined shortly and let  $\rho^* : [0, T^*] \rightarrow \mathcal{R}$  be a path in  $\mathcal{P}$  for which

$$J[\rho^*] \leq \inf_{\rho \in \mathcal{P}} J[\rho] + \delta,$$

where  $\delta$  is a constant that can be made arbitrarily small ( $\delta$  is only needed when the infimum is not a minimum). From Lemma 1, we know that there exists a sequence of times  $\{\tau_0, \tau_1, \dots, \tau_N\} \subset [0, T]$ , with  $\tau_0 := 0 \leq \tau_{k-1} < \tau_k \leq \tau_N := T$ , and a sequence of points  $\{x_0, x_1, \dots, x_N\} \in \mathcal{X}$  such that  $\rho^*$  and  $\dot{\rho}^*$  exist on any interval  $(\tau_{k-1}, \tau_k)$ ,  $k \in \{1, \dots, N\}$  and (4)–(6) hold for the path  $\rho^*$ . We can then use these points to construct a path  $\rho : [0, T] \rightarrow \mathcal{R}$  in  $\mathcal{P}_{\bar{\mathcal{X}}}$  defined as in (1).

We proceed to compare the costs associated with  $\rho^*$  and  $\rho$ . To this effect, we expand

$$J[\rho] - J[\rho^*] = \sum_{k=1}^N \int_{t_{k-1}}^{t_k} \ell(\rho(s), \dot{\rho}(s)) ds - \int_{\tau_{k-1}}^{\tau_k} \ell(\rho^*(s), \dot{\rho}^*(s)) ds$$

Applying Lemma 2 to each pair of interval  $(\tau_{k-1}, \tau_k)$  and  $(t_{k-1}, t_k)$ ,  $k \in \{1, \dots, N\}$ , we conclude that

$$J[\rho] - J[\rho^*] \leq \sum_{k=1}^N |\ell(x_{k-1}, \dot{\rho}(t_{k-1}^+)) - \ell(\rho^*(\tau_{k-1}), \dot{\rho}^*(\tau_{k-1}^+))| \Delta_k$$

$$+ |\ell(x_k, \dot{\rho}(t_k^-)) - \ell(\rho^*(\tau_k), \dot{\rho}^*(\tau_k^-))| \Delta_k + (t_k - t_{k-1} - 2\Delta_k) h_k + 2(g_{x,k} + A g_{v,k}) \Delta_k^2, \quad (8)$$

where

$$\Delta_k := \frac{1}{2} \min \{ \|x_k - x_{k-1}\|, \|\rho^*(\tau_k) - \rho^*(\tau_{k-1})\| \}, \quad h_k := \sup_{\substack{x \in \mathcal{R}, v \in \mathcal{V} \\ \|x - x_{k-1}\| \leq \epsilon_\delta}} \ell(x, v), \quad (9)$$

$$g_{x,k} := \sup_{\substack{x \in \mathcal{R}, v \in \mathcal{V} \\ \|x - x_{k-1}\| \leq \epsilon_\delta}} \|\nabla_x \ell(x, v)\|, \quad g_{v,k} := \sup_{\substack{x \in \mathcal{R}, v \in \mathcal{V} \\ \|x - x_{k-1}\| \leq \epsilon_\delta}} \|\nabla_v \ell(x, v)\|. \quad (10)$$

From (6), we further conclude that

$$\sum_{k=1}^N \Delta_k \leq \frac{1}{2} \sum_{k=1}^N \|x_k - x_{k-1}\| \leq \frac{1 + \epsilon_v}{2 + \epsilon_v} T. \quad (11)$$

---

<sup>2</sup>We denote by  $\nabla_x \ell$  and  $\nabla_v \ell$  the gradient of  $\ell(x, v)$  with respect to  $x$  and  $v$ , respectively.

In case  $\|x_k - x_{k-1}\| \leq \|\rho^*(\tau_k) - \rho^*(\tau_{k-1})\|$ , we have that  $2\Delta_k = \|x_k - x_{k-1}\| = t_k - t_{k-1}$  and therefore  $t_k - t_{k-1} - 2\Delta_k = 0$ . On the other hand, if  $\|x_k - x_{k-1}\| > \|\rho^*(\tau_k) - \rho^*(\tau_{k-1})\|$ , then  $2\Delta_k = \|\rho^*(\tau_k) - \rho^*(\tau_{k-1})\|$  and therefore

$$t_k - t_{k-1} = \|x_k - x_{k-1}\| \leq \|x_k - \rho^*(\tau_k)\| + \|\rho^*(\tau_k) - \rho^*(\tau_{k-1})\| + \|x_{k-1} - \rho^*(\tau_{k-1})\| \leq 2(\epsilon_x + \Delta_k).$$

In either case, it is true that

$$t_k - t_{k-1} - 2\Delta_k \leq 2\epsilon_x. \quad (12)$$

Using the Mean Value Theorem, (4), and (5) we also conclude that

$$\begin{aligned} |\ell(x_{k-1}, \dot{\rho}(\tau_{k-1}^+)) - \ell(\rho^*(\tau_{k-1}), \dot{\rho}^*(\tau_{k-1}^+))| &\leq g_{x,k} \|x_{k-1} - \rho^*(\tau_{k-1})\| + g_{v,k} \|\dot{\rho}(\tau_{k-1}^+) - \dot{\rho}^*(\tau_{k-1}^+)\| \\ &\leq g_{x,k} \epsilon_x + g_{v,k} \epsilon_v, \end{aligned} \quad (13)$$

$$\begin{aligned} |\ell(x_k, \dot{\rho}(\tau_k^-)) - \ell(\rho^*(\tau_k), \dot{\rho}^*(\tau_k^-))| &\leq g_{x,k} \|x_k - \rho^*(\tau_k)\| + g_{v,k} \|\dot{\rho}(\tau_k^-) - \dot{\rho}^*(\tau_k^-)\| \\ &\leq g_{x,k} \epsilon_x + g_{v,k} \epsilon_v. \end{aligned} \quad (14)$$

We can now use the bounds provided by (11), (12), and (13)–(14) in (8) to obtain

$$J[\rho] - J[\rho^*] \leq 2 \sum_{k=1}^N \left( (g_{x,k} \epsilon_x + g_{v,k} \epsilon_v + (g_{x,k} + Ag_{v,k}) \Delta_k) \Delta_k + \epsilon_x h_k \right) \leq \frac{2 + 2\epsilon_v}{2 + \epsilon_v} gT + 2N\epsilon_x h, \quad (15)$$

where

$$g := \sup_k (g_{x,k} \epsilon_x + g_{v,k} \epsilon_v + (g_{x,k} + Ag_{v,k}) \epsilon_\delta), \quad h := \sup_k h_k.$$

This finishes the proof, since the right-hand-side of (15) can be made arbitrarily small by selecting  $\epsilon_x$ ,  $\epsilon_\delta$ , and  $\epsilon_v$  sufficiently small. Note that Lemma 2 guarantees that  $N$  will not grow unbounded as  $\epsilon_x$  is decreased.  $\blacksquare$

## 4 Non-uniform sampling

In principle, solving the optimization over paths in  $\mathcal{P}_{\bar{\mathcal{X}}}$  is a simple problem that can be solved using standard tools to determine shortest-paths on finite graphs. The main difficulty with this approach is that the required number of points  $N_\epsilon$  may grow very fast as  $\epsilon$  decreases. However, this can be minimized by carefully selecting the location of the points.

The idea behind the proof of Theorem 1 is that we can construct a piecewise linear approximation  $\rho$  to an optimal path  $\rho^*$  by approximately sampling  $\rho^*$  so that the following constraints are satisfied:

1. the distance between consecutive sample points  $x_k$  should not exceed a given constant  $\epsilon_\delta$ ;
2. each sample point  $x_k$  should be in an  $\epsilon_x$ -ball of the corresponding point  $\rho^*(\tau_k)$  in the path;
3. the difference between the derivatives of  $\rho^*$  and its approximation  $\rho$  should not exceed  $\epsilon_v$ .

By forcing  $\epsilon_\delta$ ,  $\epsilon_x$ , and  $\epsilon_v$  to be sufficiently small, one can get piecewise linear paths whose cost is arbitrarily close to  $\rho^*$ . It turns out that if one examines closely the proof of Theorem 1, in particular equation (15), one conclude that  $\epsilon_\delta$  can actually be large and that it suffices that  $\epsilon_x$  and

$$g_{x,k}\epsilon_x + g_{v,k}\epsilon_v + (g_{x,k} + Ag_{v,k})\|x_k - x_{k-1}\| \quad (16)$$

be small for every  $k$ , where the constants  $g_{x,k}$  and  $g_{v,k}$  are defined in (10). This shows that one can actually allow  $\|x_k - x_{k-1}\|$  to be large in some regions, provided that  $g_{x,k} + g_{v,k}A$  be sufficiently small on those regions. To take advantage of this, one should not sample  $\mathcal{R}$  uniformly. The honeycomb-like sampling shown in Figure 1 can be used to keep (16) small with sparse sampling. Indeed, by keeping the diameter of the cells inversely proportional to  $g_{x,k} + g_{v,k}A$ , one makes sure that it is possible to pick the  $x_k$  appropriately spaced, while minimize the overall number of sample points. Note that to be able to keep  $x_k$  in a small  $\epsilon_x$ -ball of  $\rho^*(\tau_k)$ , the edges of the cells should be finely sampled. This type of spacing can be efficiently obtained using the following procedure:

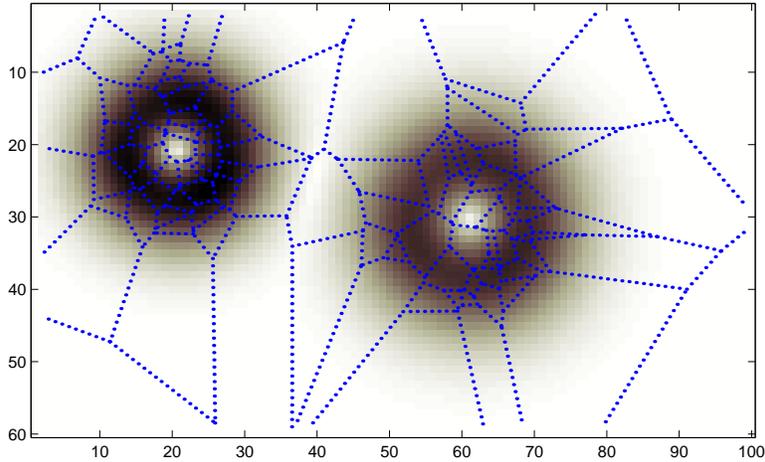


Figure 1: Honeycomb sampling

**Algorithm 1 (Honeycomb sampling).**

1. Extract randomly  $K$  points  $\mathcal{Z} := \{z_k \in \mathcal{R} : k = 1, 2, \dots, K\} \subset \mathbb{R}^n$  are, with a spacial probability density over  $\mathcal{R}$  proportional to

$$\left( \sup_{v \in \mathcal{V}} \|\nabla_x \ell(x, v)\| + A \|\nabla_v \ell(x, v)\| \right)^n.$$

The distance between points in a particular area is then roughly inversely proportional to

$$K^{\frac{1}{n}} \sup_{v \in \mathcal{V}} \|\nabla_x \ell(x, v)\| + A \|\nabla_v \ell(x, v)\|. \quad (17)$$

2. Compute the Voronoi diagram for the points in  $\mathcal{Z}$ . The size of the resulting cells in a particular area is roughly proportional to the distance between the points in that area, which is inversely proportional to (17).

3. Construct  $\mathcal{X}$  by sampling the edges of the Voronoi diagram sufficiently finely so that it is possible to chose points  $x_k$  in an  $\epsilon_x$ -ball of any point where an optimal path would cross the cell boundary.

This algorithm was used to produce the sampling in Figure 1. In this figure, the background color represents the magnitude of  $\sup_{v \in \mathcal{V}} \|\nabla_x \ell(x, v)\| + A \|\nabla_v \ell(x, v)\|$ , with dark representing large values. Increasing  $K$ , decreases the term  $(g_{x,k} + Ag_{v,k})\|x_k - x_{k-1}\|$  in (16), whereas increasing the density of sampling over the edges of the Voronoi diagram decreases  $\epsilon_x$ . It should be noted that this type of sampling does *not* correspond to a finite-elements approximation to the continuous HJB equation.

We considered several other alternative sampling algorithms and contrasted them with Honeycomb sampling. The following two methods are the simplest and do not explore the structure of the cost weight  $\ell$ :

**Algorithm 2 (Regular-grid sampling).** Construct  $\mathcal{X}$  by overlaying on  $\mathcal{R}$  a regular rectangular grid.

**Algorithm 3 (Randomized uniform sampling).** Construct  $\mathcal{X}$  by randomly extracting  $N$  points, with a uniform spacial probability density over  $\mathcal{R}$ .

The following method is also inspired by (16) but simply attempts to minimize the term  $(g_{x,k} + Ag_{v,k})\|x_k - x_{k-1}\|$ .

**Algorithm 4 (Randomized gradient-based sampling).** Construct  $\mathcal{X}$  by randomly extracting  $N$  points, with a spacial probability density over  $\mathcal{R}$  proportional to

$$\left( \sup_{v \in \mathcal{V}} \|\nabla_x \ell(x, v)\| + A \|\nabla_v \ell(x, v)\| \right)^n.$$

In the next section we compare the performance of these four sampling algorithms.

## 5 Minimum-risk path planning for groups of UAVs

Consider a packet of  $m$  UAVs flying in a region  $\mathcal{R} \subset \mathbb{R}^3$  under the threat of  $k$  Surface-to-Air Missile (SAM) sites located at positions  $z_1, z_2, \dots, z_k \in \mathbb{R}^3$ . In this context, one would like to compute a path  $\rho : [0, T] \rightarrow \mathcal{R}$  for the packet of UAVs that starts at an initial position  $x_i$  and ends at a final position  $x_f$ , maximizing the probability that the UAVs will survive the journey. In general, different UAVs may have different defensive/stealth capabilities and therefore their probabilities of survival are distinct. Because of this, minimal-risk path planning is a multi-criteria optimization problem. We will pursue here Pareto-optimal paths, i.e., paths for which the probability of any single UAV surviving cannot be improved without decreasing the survivability of another UAV in the packet.

Denoting by  $p_j^{\text{survive}}[\rho]$  the probability that the  $j$ th UAV safely reaches the destination, Pareto-optimal (maximal) paths can be obtained as the solution to single-criteria optimization problems of the form

$$\max_{\rho} \sum_{j=1}^m \lambda_j p_j^{\text{survive}}[\rho],$$

where the  $\lambda_j$  denote positive constants [5]. Assuming that velocities are normalized so the maximum speed of the packet is equal to one, the optimization should be performed as  $\rho$  ranges over the set  $\mathcal{P}$  considered in the previous sections. Note that in general, risk is minimized for the maximum speed so there is no reason to consider paths with speeds smaller than the maximum. We ignore here all constraints posed by the aircraft dynamics, other than its maximum speed. The path generated by this optimization would serve as a reference trajectory for the formation in algorithms such as the ones proposed, e.g., in [18, 7, 17]. One could also take fuel consumption and path length as additional criteria for Pareto-optimality. Although we do not pursue this here, it would be straightforward.

## 5.1 Risk Model

We assume that the probability of the  $j$ th UAV being hit by the  $i$ th SAM in an elementary interval of time  $dt$  is given by

$$\eta_{ij}(x, \dot{x}, z_i)dt,$$

where  $x$  and  $\dot{x}$  denote the position and velocity of the packet of UAVs and the function  $\eta_{ij}$  is called the *risk density for the  $j$ th UAV with respect to the  $i$ th SAM*. When the SAMs operate independently, the probability of surviving all the SAMs is given by

$$\prod_{i=1}^n (1 - \eta_{ij}(x, \dot{x}, z_i)dt).$$

The main reason why risk density functions depend on the position of the SAMs and the position and velocity of the UAVs, is that the radar signature of the UAVs is a function of their distance and flight angle with respect to the SAMs. We will expand on this below.

Suppose now that the packet of UAVs fly over a path  $\rho : [0, T] \rightarrow \mathcal{R}$ . Assuming that the probability of being hit over disjoint path elements is independent, the probability  $p_j^{\text{survive}}[\rho]$  of the  $j$ th UAV surviving the whole path  $\rho$  is given by the limit as  $dt \rightarrow 0$  of

$$\prod_{k=0}^{T/dt} \prod_{i=1}^n \left(1 - \eta_{ij}(\rho(kdt), \dot{\rho}(kdt), z_i)dt\right).$$

Taking log and making  $dt \rightarrow 0$ , we obtain<sup>3</sup>

$$\begin{aligned} \log p_j^{\text{survive}}[\rho] &= \lim_{dt \rightarrow 0} \sum_{k=0}^{T/dt} \sum_{i=1}^n \log \left(1 - \eta_{ij}(\rho(kdt), \dot{\rho}(kdt), z_i)dt\right) \\ &= - \lim_{dt \rightarrow 0} \sum_{k=0}^{T/dt} \sum_{i=1}^n \eta_{ij}(\rho(kdt), \dot{\rho}(kdt), z_i)dt \\ &= - \sum_{i=1}^n \int_0^T \eta_{ij}(\rho(t), \dot{\rho}(t), z_i)dt. \end{aligned}$$

---

<sup>3</sup>Here, we used the fact that  $\log(1 - \epsilon) \approx -\epsilon$  for  $\epsilon \ll 1$ .

We therefore conclude that, under the previous risk model, we can express  $p_{\text{survive}}[\rho]$  as

$$p_j^{\text{survive}}[\rho] = e^{-\int_0^T \ell_j(\rho(t), \dot{\rho}(t)) dt},$$

where

$$\ell_j(x, v) := \sum_{i=1}^n \eta_{ij}(x, v, z_i). \quad (18)$$

This model is consistent with the expectation that if one remains under the danger for an increasingly long amount of time, the probability of survival eventually converges to zero.

Since the function  $s \mapsto e^{-s}$  is monotone decreasing, paths that are Pareto-optimal (maximal) with respect to the rewards  $p_j^{\text{survive}}[\rho]$  are also Pareto-optimal (minimal) with respect to the costs

$$J_j[\rho] := \int_0^T \ell_j(\rho(t), \dot{\rho}(t)) dt. \quad (19)$$

We can therefore find these paths by solving the Weighted Anisotropic Shortest-Path Problem 1 considered in Sections 3–4.

## 5.2 Numerical results

In this section, we analyze the performance of the algorithms proposed to solve Problem 1 in the context of the minimum-risk path planning problem formulated above. We utilize a cost of the form (19), computed from a realistic risk density function.

The risk density functions  $\eta_{ij}$  depend on the distance and attitude of the UAVs with respect to the SAM sites. This is because SAMs are usually guided by a tracking radar that locks on the target UAV and guides the SAM until interception. Therefore,  $\eta_{ij}$  depends significantly on the aircraft’s Radar Cross Section (RCS), which is a measure of its ability to reflect radar signals in the direction of the radar receiver. RCSs for typical aircraft can be found in [16]. However, one should keep in mind that UAVs (such as Boeing’s X-45A [4]) are smaller than a normal fighter aircraft and have stealth capabilities. This makes them more interesting from the perspective of minimum-risk path planning.

The RCS of an aircraft is a function of the azimuth angle  $\alpha$  and the elevation angle  $\theta$  of the line-of-sight vector from the UAV to the SAM’s tracking radar, expressed in the body frame of the UAV (cf. Figure 2). Since these angles are defined in the body frame, the RCS—and consequently the risk density functions  $\eta_{ij}$ —depend on the attitude of the aircraft and therefore on the direction of motion. This leads to anisotropic cost-weightings  $\ell_j$  given by (18). The results summarized below use risk density functions consistent with RCSs taken from a challenge problem set forth by Boeing for the DARPA program Mixed Initiative Control for Automa-teams (MICA) [2, 3]. The resulting cost-weightings are strongly anisotropic, with low costs when the line-of-sight vector from the UAVs to the SAM’s tracking radar corresponds to low elevation and azimuth angles. The positions and characteristics of the SAM sites were also taken from the same challenge problem.

In this study, we consider two scenarios that are representative of typical minimal-risk path planning problems. These scenarios are shown in Figure 3. In this figure, each cross ( $\times$ ) represents the location of one SAM site. Each site is surrounded by a circle that indicates the maximum

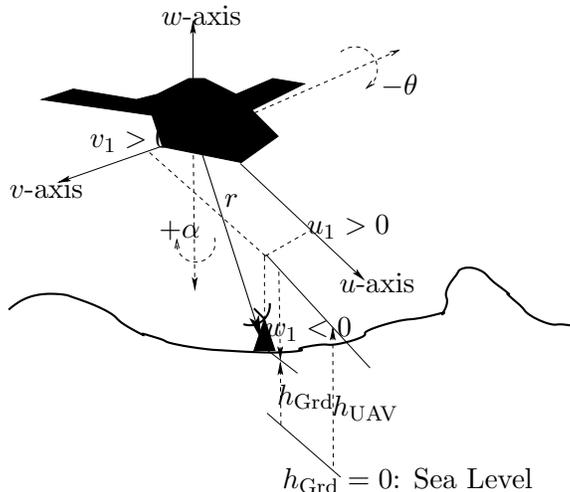


Figure 2: Azimuth ( $\alpha$ ) and Elevation ( $\theta$ ) angles for the line-of-sight vector from the UAV to a SAM's tracking radar.

range of the SAM. However, one should emphasize that it is possible for the UAVs to fly well inside the SAMs' ranges without being threatened, provided that their RCSs are kept low by appropriate choice of position and attitude. In the first scenario there are five medium range SAMs (MS1-MS5) and two long range SAMs (LS1-LS2) in the region, whereas in the second one there are only three medium range SAMs (MS1-MS3).

In the first scenario, the group of UAVs are required to fly from the star ( $\star$ ) location at the bottom of Figure 3a to the location of the long range SAM LS2. In the second scenario, the UAVs are required to fly from the star ( $\star$ ) location in the top of Figure 3b to a position in the bottom left of the same figure, while passing through a way-point over the medium range SAM MS1. The main distinguishing feature of the two scenarios is that in the former most of the path can be flown over safe areas, whereas in the latter the group of UAVs are forced to spend a significant portion of the path over a potentially unsafe area. In both figures, small dots represent the sample points used to discretize the problem, whereas solid lines denote the corresponding optimal paths. For illustration, we used randomized uniform sampling in Figure 3a and honeycomb sampling in Figure 3b.

Figure 4 show a comparison of the costs obtained for each scenario for different sampling algorithms and different sampling densities. In the vertical axis we plotted the minimum cost obtained and in the horizontal axis the time it took to compute the corresponding optimal path. The algorithms were implemented in MATLAB and ran on a Dell Dimension 4500 workstation (Pentium 4 processor, 2GHz clock, 768Mb of RAM). We compare the four sampling algorithms described in Section 4: honeycomb, regular-grid, randomized uniform, and randomized gradient-based. For each algorithms we considered several sampling densities, leading to distinct execution times and distinct costs.

Figure 4a shows the results obtained in scenario 1. As can be seen in Figure 3a, for the given configuration of SAMs most of the optimal path is inside the no-risk region, except for the final segment entering the fire-range of the Long Range SAM LS2. In this case, most sampling methods perform well with the exception of regular-grid sampling. This problem has been noted by several authors and is often referred to as digitization bias (e.g., in Tsitsiklis [22]) and it is essentially

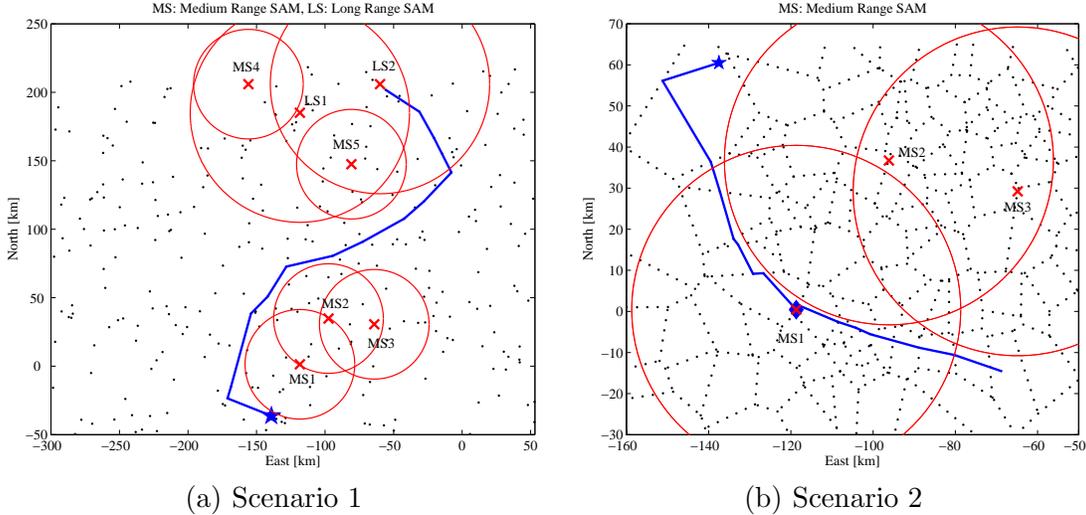


Figure 3: Typical minimum-risk paths for the two scenarios considered. Randomized uniform sampling was used to compute the path on the left and honeycomb sampling for the path on the right.

caused by an implicit discretization of the control space. In practice, Scenario 1 turns out not to be very challenging for most methods because the UAVs can avoid the threat areas and therefore the anisotropic nature of the problem vanishes.

Figure 4b shows the same comparison for scenario 2. As can be seen in Figure 3b, now the optimal path must go through the danger area and the performance of the several algorithms differs significantly. Not surprisingly, regular-grid sampling remains non-competitive. The randomized uniform and randomized gradient-based sampling methods exhibit similar performances. These methods start producing reasonable paths for low sampling densities (taking less than 10 seconds). However, their performance does improve significantly for sampling densities that lead to optimization problems that can be solved in less than one minute. On the other hand, honeycomb sampling consistently leads to lower costs for sampling densities that lead to optimization problems solvable in more than 20 seconds. The “outliers” point corresponding to a cost of 320 and computation time of 35 seconds, resulted from a particularly unfavorable configuration of cells. This type of situation is unavoidable with randomized methods. Figure 5 shows typical paths for the randomized uniform, randomized gradient-based, and honeycomb sampling methods obtained for a large number of points. The plots on this figure confirm that honeycomb sampling produces lower cost paths with smaller sampling density.

Table 1 shows the mean value and the standard deviation of the costs obtained with each sampling method. The average costs from honeycomb sampling are about 8 to 37% smaller than the ones from other sampling methods.

## 6 Conclusion

In this paper we propose an algorithm for the computation of weighted anisotropic shortest paths, which reduces the continuous problem to an optimization over a finite graph that can be efficiently

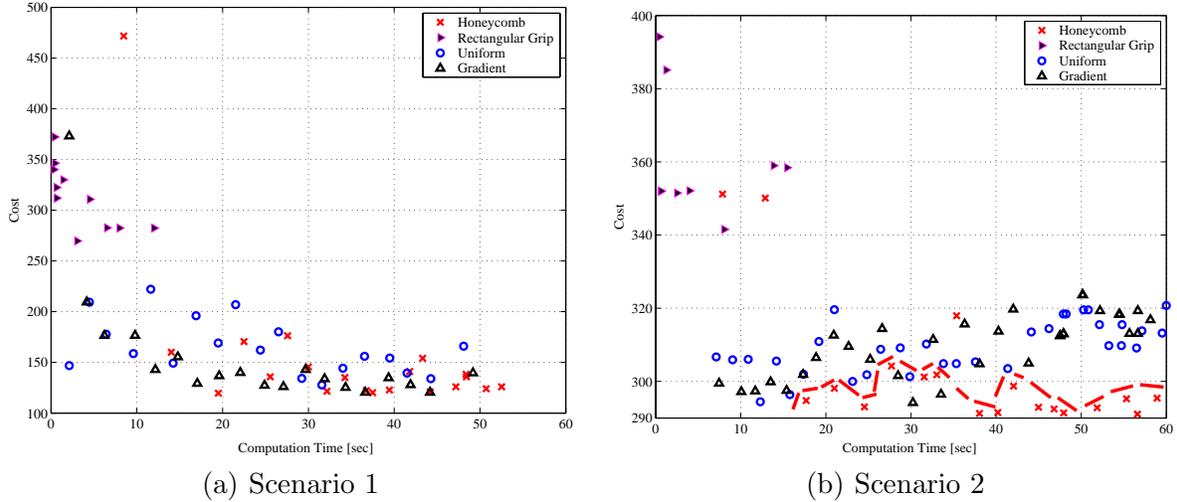


Figure 4: Cost Comparisons

Table 1: The Mean and the Standard Deviation of the Costs (Scenario 2)

	Honeycomb	Rectangular Grip	Uniform	Gradient
Mean	301.04	338.87	309.48	310.49
Standard Deviation	17.76	70.89	6.91	8.28

solved. The algorithm proposed restricts the search to paths formed by the concatenation of straight-line segments between points from a suitably chosen discretization of the continuous region. To maximize efficiency, the discretization should not be uniform. We propose a novel “honeycomb” sampling algorithm that minimizes the cost penalty introduced.

This methodology is applied to the computation of paths for groups of Unmanned Air Vehicles (UAVs) that minimize the risk of being destroyed by ground defenses. We show that this problem can be formulated as a shortest path optimization and that a novel honeycomb sampling method can efficiently produce low-risk paths with less computation time than other methods, for the same or better costs.

Future work will include the computation of non-conservative bounds on the cost penalty introduced by discretization with honeycomb and other sampling algorithms. As well the develop of combined sampling and graph optimization algorithms to avoid oversampling when this does not introduce significant cost benefit.

## Appendix

In what follows, given a path  $\rho : [0, T] \rightarrow \mathcal{R}$  in  $\mathcal{P}$  and a time  $t \in [0, T]$ , we use the notation  $\rho(t^+)$  and  $\rho(t^-)$  to denote the limits of  $\rho(s)$  as  $s \rightarrow t$  from above and below, respectively.

*Proof of Lemma 1.* Let  $\mathcal{X}$  be a set that is “sufficiently dense” on  $\mathcal{R}$  so that for every  $r \in \mathcal{R}$ , there is a point  $x \in \mathcal{X}$  for which  $\|r - x\| \leq \epsilon$ . We will specify shortly the value for the constant  $\epsilon$ .

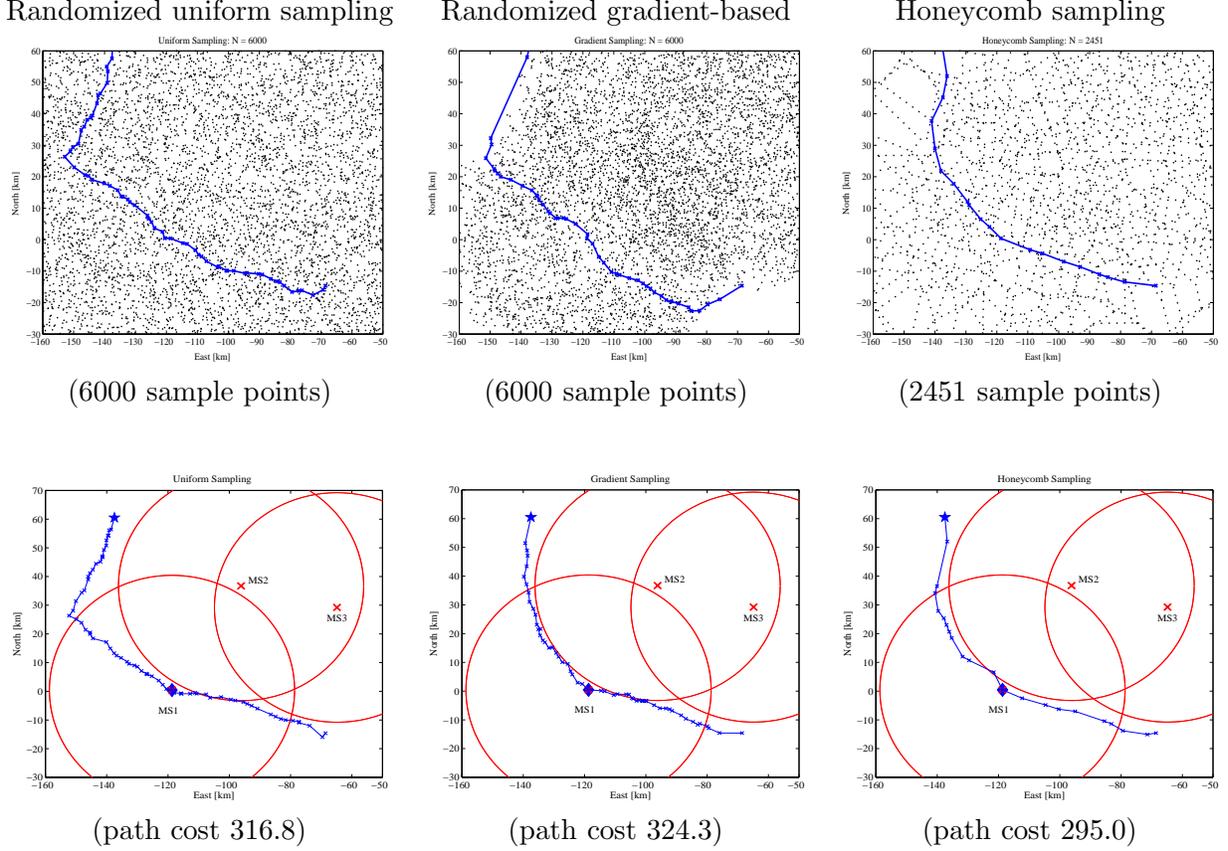


Figure 5: Typical sampling (top) and the corresponding optimal paths (bottom) for Scenario 2. Three sampling methods are compared: randomized uniform (left), randomized gradient-based (center), and honeycomb sampling (right).

We construct the sequences  $\{\tau_k\}$ ,  $\{x_k\}$  recursively as follows. In this construction, we use three parameters  $\alpha \in (0, 1]$  and  $\gamma_{\max} \geq \gamma_{\min} > 0$  that will be specified later. The construction will guarantee (by an induction argument) that

$$\|x_k - \rho(\tau_k)\| \leq \alpha \epsilon_x \quad \forall k \in \{0, 1, \dots, N\}. \quad (20)$$

Since  $\alpha \leq 1$ , this shows that (4) holds. We start by initializing  $\tau_0 = 0$  and by picking  $x_0 \in \mathcal{X}$  such that (20) holds for  $k = 0$ . This is always possible, provided that  $\alpha \epsilon_x \geq \epsilon$ . Then, for each  $k \geq 1$ , pick  $\tau_k$  to be the any time instant in

$$[\tau_{k-1} + \gamma_{\min}, \tau_{k-1} + \gamma_{\max}] \cap [0, T]$$

for which  $\rho(t)$  is in the ball of radius  $\epsilon_\delta - \alpha \epsilon_x$  centered around  $x_{k-1}$ , for all  $t \in [\tau_{k-1}, \tau_k]$ , which guarantees by construction that (3) holds. Such a time always exists because  $\rho$  is continuous and is inside the ball in the interval  $[\tau_{k-1}, \tau_{k-1} + \gamma_{\min}]$ , provided that  $\gamma_{\min} \leq \epsilon_\delta - 2\alpha \epsilon_x$ . Indeed, for  $t \in [\tau_{k-1}, \tau_{k-1} + \gamma_{\min}]$ ,

$$\begin{aligned} \|\rho(t) - x_{k-1}\| &\leq \|\rho(t) - \rho(\tau_{k-1}) + \rho(\tau_{k-1}) - x_{k-1}\| \\ &\leq \|\rho(t) - \rho(\tau_{k-1})\| + \|\rho(\tau_{k-1}) - x_{k-1}\| \end{aligned}$$

$$\leq \gamma_{\min} + \alpha\epsilon_x \leq \epsilon_\delta - \alpha\epsilon_x.$$

Here, we used the fact that  $\|\dot{\rho}\| \leq 1$  on  $[\tau_{k-1}, \tau_{k-1} + \gamma_{\min}]$ . We can then pick  $x_k$  to be any point in a ball of radius  $\alpha\epsilon_x \geq \epsilon$  centered at  $\rho(\tau_k)$ , which guarantees that the induction step needed to prove (20) holds. Moreover,

$$\|x_k - x_{k-1}\| \leq \|x_k - \rho(\tau_k) + \rho(\tau_k) - x_{k-1}\| \leq \alpha\epsilon_x + \|\rho(\tau_k) - x_{k-1}\| \leq \epsilon_\delta.$$

which shows that (2) holds. To prove (5), we define  $\delta_k := \dot{\rho}(\tau) - \frac{x_k - x_{k-1}}{\tau_k - \tau_{k-1}}$ ,  $\tau \in (\tau_{k-1}, \tau_k)$ . Since

$$x_k - x_{k-1} = (\tau_k - \tau_{k-1})(\dot{\rho}(\tau) - \delta_k), \quad (21)$$

we conclude that

$$\begin{aligned} \left\| \dot{\rho}(\tau) - \frac{x_k - x_{k-1}}{\tau_k - \tau_{k-1}} \right\| &= \left\| \dot{\rho}(\tau) - \frac{\dot{\rho}(\tau) - \delta_k}{\|\dot{\rho}(\tau) - \delta_k\|} \right\| \\ &= \left\| \left(1 - \frac{1}{\|\dot{\rho}(\tau) - \delta_k\|}\right) \dot{\rho}(\tau) + \frac{\delta_k}{\|\dot{\rho}(\tau) - \delta_k\|} \right\| \\ &\leq \frac{|\|\dot{\rho}(\tau) - \delta_k\| - 1|}{\|\dot{\rho}(\tau) - \delta_k\|} + \frac{\|\delta_k\|}{\|\dot{\rho}(\tau) - \delta_k\|} \leq \frac{2\|\delta_k\|}{1 - \|\delta_k\|}. \end{aligned} \quad (22)$$

Thus (5) will hold, provided that

$$\frac{2\|\delta_k\|}{1 - \|\delta_k\|} \leq \epsilon_v \quad \Leftrightarrow \quad \|\delta_k\| \leq \frac{\epsilon_v}{2 + \epsilon_v}.$$

Using the Mean Value Theorem, we conclude that

$$\begin{aligned} \|\delta_k\| &= \left\| \dot{\rho}(\tau) - \frac{x_k - \rho(\tau_k) - x_{k-1} + \rho(\tau_{k-1}) + \rho(\tau_k) - \rho(\tau_{k-1})}{\tau_k - \tau_{k-1}} \right\| \\ &\leq \frac{\|x_k - \rho(\tau_k)\| + \|x_{k-1} - \rho(\tau_{k-1})\|}{\tau_k - \tau_{k-1}} + \left\| \dot{\rho}(\tau) - \frac{\rho(\tau_k) - \rho(\tau_{k-1})}{\tau_k - \tau_{k-1}} \right\| \\ &\leq \frac{2\alpha\epsilon_x}{\tau_k - \tau_{k-1}} + \frac{A}{2}(\tau_k - \tau_{k-1}) \leq \frac{2\alpha\epsilon_x}{\gamma_{\min}} + \frac{A\gamma_{\max}}{2}. \end{aligned}$$

By selecting  $\gamma_{\max}$  and  $\alpha$  sufficiently small, and keeping in mind that one also needs  $\gamma_{\min} \leq \epsilon_\delta - 2\alpha\epsilon_x^4$ , it is always possible to make sure that

$$\frac{2\alpha\epsilon_x}{\gamma_{\min}} + \frac{A\gamma_{\max}}{2} \leq \frac{\epsilon_v}{2 + \epsilon_v} \quad (23)$$

and therefore that (5) holds. The price to pay is that a small value for  $\alpha$  will necessarily require small  $\epsilon \leq \alpha\epsilon_x$  and therefore a fine gridding of  $\mathcal{R}$ . Finally, from (21) and the fact that  $\|\dot{\rho}(\tau)\| \leq 1$ , we conclude that

$$\|x_k - x_{k-1}\| \leq (\tau_k - \tau_{k-1})(1 + \|\delta_k\|) \leq \frac{2(1 + \epsilon_v)}{2 + \epsilon_v}(\tau_k - \tau_{k-1}),$$

from which (6) follows. Finally, note that as one decreases  $\epsilon_x$ , with need a finer gridding of  $\mathcal{R}$  but  $\gamma_{\min}$  need not decrease and therefore  $N$  remains bounded.  $\blacksquare$

---

<sup>4</sup>The best bound would be obtained for  $\gamma_{\max} = \gamma_{\min} = 2\sqrt{\frac{\alpha\epsilon_x}{A}}$  and would lead to  $\|\delta_k\| \leq 2\sqrt{\alpha\epsilon_x A}$ , which means that we would need  $2\sqrt{\alpha\epsilon_x A} \leq \frac{\epsilon_v}{2 + \epsilon_v}$ .

**Proposition 1.** Given a path  $\rho : [0, T] \rightarrow \mathcal{R}$  in  $\mathcal{P}$  and an interval  $(t, \tau) \subset [0, T]$  on which  $\rho$  is twice continuously differentiable,

$$|\ell(\rho(t), \dot{\rho}(t^+)) - \ell(\rho(\tau), \dot{\rho}(\tau^-))| \leq g_x(\tau - t) + a g_v(\tau - t) \quad (24)$$

and

$$|\ell(\rho(t), \dot{\rho}(t^+)) - \ell(\rho(\tau), \dot{\rho}(\tau^-))| \leq g_x(\tau - t) + 2g_v \quad (25)$$

where

$$g_x := \sup_{s \in (t, \tau)} \|\nabla_x \ell(\rho(s), \dot{\rho}(s))\|, \quad g_v := \sup_{s \in (t, \tau)} \|\nabla_v \ell(\rho(s), \dot{\rho}(s))\|, \quad a := \sup_{s \in (t, \tau)} \|\ddot{\rho}(s)\|$$

*Proof of Proposition 1.* Since  $\ell$  is continuously differentiable on  $(t, \tau)$ , by the Mean Value Theorem we conclude that

$$|\ell(\rho(t), \dot{\rho}(t^+)) - \ell(\rho(\tau), \dot{\rho}(\tau^-))| = g_x \|\rho(t) - \rho(\tau)\| + g_v \|\dot{\rho}(t^+) - \dot{\rho}(\tau^-)\|.$$

Since  $\|\dot{\rho}\| = 1$  on  $(t, \tau)$ , we conclude that  $\|\dot{\rho}(t^+) - \dot{\rho}(\tau^-)\| \leq 2$  and, because of the Mean Value Theorem, that  $\|\rho(t) - \rho(\tau)\| \leq \tau - t$ , leading to (25). On the other hand, again by the Mean Value Theorem  $\|\dot{\rho}(t^+) - \dot{\rho}(\tau^-)\| \leq a(\tau - t)$  and therefore (24) also holds.  $\blacksquare$

*Proof of Lemma 2.* For simplicity of notation let us define

$$\ell_i(t) := \begin{cases} \ell(\rho_i(t), \dot{\rho}_i(t)) & t \in (t_i, \tau_i) \\ \ell(\rho_i(t), \dot{\rho}_i(t^+)) & t = t_i \\ \ell(\rho_i(t), \dot{\rho}_i(t^-)) & t = \tau_i \end{cases} \quad i \in \{1, 2\}.$$

From the fact that  $\dot{\rho}_i = 1$ , the Mean-Value Theorem and the definition and  $\Delta$ , we conclude that

$$\tau_i - t_i \geq \|\rho_i(\tau_i) - \rho_i(t_i)\| \geq 2\Delta.$$

We can therefore split each integral in (7) in three parts as follows.

$$\begin{aligned} & \int_{t_2}^{\tau_2} \ell_2(t) dt - \int_{t_1}^{\tau_1} \ell_1(t) dt \\ &= \int_{t_2}^{t_2+\Delta} \ell_2(t) dt - \int_{t_1}^{t_1+\Delta} \ell_1(t) dt + \int_{\tau_2-\Delta}^{\tau_2} \ell_2(t) dt - \int_{\tau_1-\Delta}^{\tau_1} \ell_1(t) dt + \int_{t_2+\Delta}^{\tau_2-\Delta} \ell_2(t) dt - \int_{t_1+\Delta}^{\tau_1-\Delta} \ell_1(t) dt \\ &= \int_{t_2}^{t_2+\Delta} \ell_2(t) - \ell_1(t + t_1 - t_2) dt + \int_{\tau_2-\Delta}^{\tau_2} \ell_2(t) - \ell_1(t + \tau_1 - \tau_2) dt + \int_{t_2+\Delta}^{\tau_2-\Delta} \ell_2(t) dt - \int_{t_1+\Delta}^{\tau_1-\Delta} \ell_1(t) dt. \end{aligned} \quad (26)$$

Using the triangular inequality together with Proposition 1, we conclude that, for every  $t \in (t_2, t_2 + \Delta]$ ,

$$\begin{aligned} |\ell_2(t) - \ell_1(t + t_1 - t_2)| &\leq |\ell_2(t_2) - \ell_1(t_1)| + |\ell_2(t) - \ell_2(t_2)| + |\ell_1(t + t_1 - t_2) - \ell_1(t_1)| \\ &\leq |\ell_2(t_2) - \ell_1(t_1)| + (g_{x,1} + g_{x,2} + a_1 g_{v,1} + a_2 g_{v,2})(t - t_2). \end{aligned} \quad (27)$$

Similarly, we can also conclude that, for every  $t \in [\tau_2 - \Delta, \tau_2)$

$$|\ell_2(t) - \ell(\rho_1(t + \tau_1 - \tau_2))| \leq |\ell_2(\tau_2) - \ell_1(\tau_1)| + (g_{x,1} + g_{x,2} + a_1g_{v,1} + a_2g_{v,2})(\tau_2 - t). \quad (28)$$

From (26) (with the last term dropped to get an upper bound) together with (27), and (28), we obtain (7). Using (25) instead of (24) in the application of Proposition 1, would lead to the last term of (7) equal to  $(g_{x,1} + g_{x,2})\Delta^2 + 2(g_{v,1} + g_{v,2})\Delta$ . ■

## References

- [1] L. Aleksandrov, A. Maheshwari, and J.-R. Sack. Approximation algorithms for geometric shortest path problems. In *Proc. of the thirty-second annual ACM symposium on Theory of computing*, pages 286–295. ACM Press, 2000. ISBN 1-58113-184-4.
- [2] Boeing. Challenge problem #1.1, Feb. 2003. DARPA’s program “Mixed Initiative Control for Automa-teams.”
- [3] *User Guide for the Open Experimental Platform (OEP), version 1.1, revision 11*. Boeing, Mar. 2003.
- [4] Boeing Phantom Works. Unmanned combat air vehicle (X-45). Available online at <http://www.boeing.com/phantom/ucav.html>, Mar. 2003.
- [5] S. P. Boyd and C. H. Barratt. *Linear Controller Design: Limits of Performance*. Prentice-Hall, New Jersey, 1991.
- [6] L. C. Polymenakos, D. P. Bertsekas, and J. N. Tsitsiklis. Implementation of efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automat. Contr.*, 43(2):278–283, Feb. 1998.
- [7] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. In *Proc. of the 15th World Congress of Int. Federation of Automat. Contr.*, July 2002.
- [8] I. M. Gelfand and S. V. Fomin. *Calculus of Variations*. Selected Publications in the Mathematical Sciences. Prentice-Hall, Englewood Cliffs, N. J., 1963.
- [9] J. Hershberger and S. Suri. An optimal algorithm for euclidean shortest paths in the plane. *SIAM J. on Computing*, 28(6):2215–2256, 1999. ISSN 0097-5397.
- [10] S. Kapoor, S. N. Maheshwari, and J. S. B. Mitchell. An efficient algorithm for euclidean shortest paths among polygonal obstacles in the plane. *Discrete Comput. Geometry*, 18:377–383, 1997.
- [11] H. J. Kushner and P. G. Dupuis. *Numerical Methods for Stochastic Control Problems in Continuous Time*. Springer-Verlag, New York, 1992.
- [12] M. Lanthier, A. Maheshwari, and J.-R. Sack. Shortest anisotropic paths on terrains. In J. Wiedemann, P. van Emde Boas, and M. Nielsen, editors, *Proc. 26th Int. Colloquium Automata, Languages and Programming (ICALP’99)*, volume 1644 of *Lecture Notes in Computer Science*, pages 524–533. Springer-Verlag, Berlin, 1999.
- [13] C. S. Mata and J. S. B. Mitchell. A new algorithm for computing shortest paths in weighted planar subdivisions (extended abstract). In *Proc. of the thirteenth annual symposium on Computational geometry*, pages 264–273. ACM Press, 1997. ISBN 0-89791-878-9.

- [14] J. S. B. Mitchell. Geometric shortest paths and network optimization. In Sack and Urrutia [20].
- [15] J. S. B. Mitchell and C. H. Papadimitriou. The weighted region problem: finding shortest paths through a weighted planar subdivision. *J. of the ACM*, 38(1):18–73, 1991. ISSN 0004-5411.
- [16] Naval Air Warfare Center Weapons Division. Electronic warfare and radar systems engineering handbook. Available online at <https://ewhdbks.mugu.navy.mil>, Jan. 2003.
- [17] P. Ogren, E. Fiorelli, and N. E. Leonard. Formations with a mission: Stable coordination of vehicle group maneuvers. In *Proc. of the Int. Symposium on the Mathematical Theory of Networks and Syst.*, Aug. 2002.
- [18] A. Pant, P. Seiler, and K. Hedrick. Mesh stability of look-ahead interconnected systems. *IEEE Trans. on Automat. Contr.*, 47(2):403–407, Feb. 2002.
- [19] N. C. Rowe and R. S. Ross. Optimal grid-free path planning across arbitrarily contoured terrain with anisotropic friction and gravity effects. *IEEE Trans. Robot. Automat.*, 6(5):540–553, Oct. 1990.
- [20] J.-R. Sack and J. Urrutia, editors. *Handbook of Computational Geometry*. Elsevier, Amsterdam, 2000.
- [21] S. Sundar and Z. Shiller. Optimal obstacle avoidance based on the hamilton-jacobi-bellman equation. *IEEE Trans. Robot. Automat.*, 13(2):305–310, 1997.
- [22] J. N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automat. Contr.*, 40(9):1528–1538, Sept. 1995.