Discrete Approximations to Continuous Shortest-Path:

Application to Minimum-Risk Path Planning for Groups of Uninhabited Air Vehicles (UAVs)

Jongrae Kim João Hespanha

University of California
Santa Barbara

Research supported by DARPA/IXO MICA program

Outline

1. Motivation—minimum-risk path planning
2. Discretization approach to shortest-path
3. Sampling methods
4. Back to minimum-risk path planning…

Anisotropic shortest-path problem

Compute path ρ from initial position x_i
to final position x_f

minimizing the integral cost

cost-weighting function

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

position

velocity vector

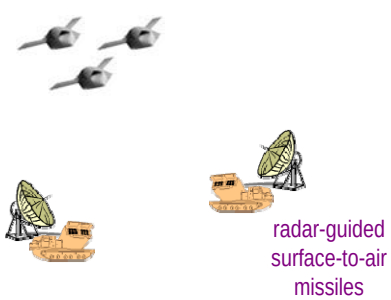
as opposed to isotropic shortest path

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

speed

Motivating problems

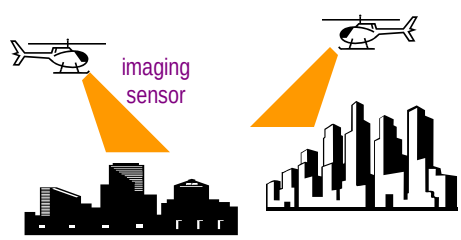
Minimum-risk path planning
for fixed-wing UAVs



- probability of radar acquisition depends on UAVs attitude
- "safe-space" is very much nonconvex

focus of the paper

Path-planning for UAV surveillance
in complex 3D environments



- region covered by sensor depends on UAVs attitude
- free-space is very much nonconvex

Radar acquisition

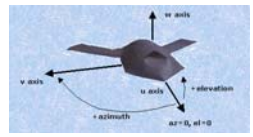
probability of UAV being acquired by radar on an elementary time interval dt:

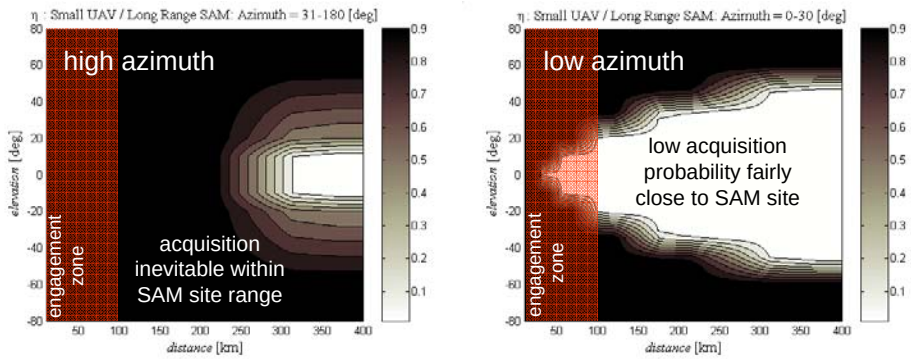
acquisition density function

$$\eta(x, \dot{x}, m)dt$$

position/ velocity vector

radar position





η changes significantly with the attitude of the UAV with respect to the radar

data from DARPA/MICA challenge problem v1.1

Survivability ("risk")

k SAM sites

probability of UAV being acquired by a radar on a path ρ with duration T

$$p^{\mathrm{acq}}[\rho] = 1 - e^{-\int_0^T \ell^{\mathrm{acq}}(\rho(t), \dot{\rho}(t))dt} \qquad \ell^{\mathrm{acq}}(x, v) := \sum_{i=1}^{k} \eta_i(x, v, m_i)$$

probability of UAV being destroyed by a radar on a path ρ with duration T

$$p^{\mathrm{kill}}[\rho] = p^{\mathrm{letha}}\left(1 - e^{-\int_0^T \ell^{\mathrm{acq}}(\rho(t), \dot{\rho}(t))dt}\right)$$

conditional probability of kill, given that UAV was acquired by radar

Survivability ("risk")

k SAM sites

probability of UAV being acquired by a radar on a path ρ with duration T

$$p^{\mathrm{acq}}[\rho] = 1 - e^{-\int_0^T \ell^{\mathrm{acq}}(\rho(t),\dot{\rho}(t))dt} \qquad \ell^{\mathrm{acq}}(x,v) := \sum_{i=1}^{k} \eta_i(x,v,m_i)$$

probability of UAV being destroyed by a radar on a path ρ with duration T

$$p^{\mathrm{kill}}[\rho] = p^{\mathrm{letha}}\left(1 - e^{-\int_0^T \ell^{\mathrm{acq}}(\rho(t),\dot{\rho}(t))dt}\right)$$

conditional probabilit
that UAV w

$$\log\left(\frac{p^{\mathrm{leth}}}{p^{\mathrm{leth}} - p^{\mathrm{kill}}[\rho]}\right) = \int_0^T \ell^{\mathrm{acq}}(\rho(t),\dot{\rho}(t))dt$$

small p^{kill} $\Leftrightarrow$ small $\int_0^T \ell^{\mathrm{acq}}(\cdot)dt$

Anisotropic shortest-path problem

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

over all paths ρ with (non-fixed) duration $T > 0$ such that

$$\rho(0) = x_i \quad \text{initial position}$$
$$\rho(T) = x_f \quad \text{final position}$$
$$\|\dot{\rho}\| \leq \text{maximum velocity of slowest UCAV}$$

Anisotropic shortest-path problem

UCSB

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

over all paths ρ with (non-fixed) duration $T > 0$ such that

$$\rho(0) = x_i \quad \text{initial position}$$
$$\rho(T) = x_f \quad \text{final position}$$
$$||\dot{\rho}|| \leq \text{maximum velocity of slowest UCAV}$$

We would obtain similar problem formulations if the previous setup was generalized to:

1. *Unknown radar positions* (need to take expected value with respect to the distribution of the radars, analytically or using randomized methods)
2. *Groups of UAVs* flying together
3. *Multi-criteria* optimization (e.g., considering constraints on path duration and fuel consumption)

Anisotropic shortest-path problem

UCSB

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

over all paths ρ with (non-fixed) duration $T > 0$ such that

$$\rho(0) = x_i \quad \text{initial position}$$
$$\rho(T) = x_f \quad \text{final position}$$
$$||\dot{\rho}|| \leq \text{maximum velocity of slowest UCAV}$$

Variational approach

Assuming that there exists a solution to the Hamilton-Jacobi-Bellman PDE

$$\min_{v \in \mathcal{V}} H(x, v, \nabla_x V(x)) = 0 \qquad V(x_f) = 0$$

Hamiltonian $\quad H(x, v, p) = \ell(x, v) + \langle p, v \rangle$

the optimal path is given by

$$\rho(t) = x_i + \int_0^t \arg\min_{v \in \mathcal{V}} H(\rho(\tau), v, \nabla_x V(\rho(\tau)))d\tau$$

Anisotropic shortest-path problem

UCSB

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

over all paths ρ with (non-fixed) d

$$\rho(0) = x_i \quad \text{initial}$$
$$\rho(T) = x_f \quad \text{final}$$
$$||\dot{\rho}|| \leq \text{maximum}$$

But…
1. often HJB has no solution &
2. finding the solution to a 3D PDE is computationally very difficult

Variational approach

Assuming that there exists a solution to the Hamilton-Jacobi-Bellman PDE

$$\min_{v \in \mathcal{V}} H(x, v, \nabla_x V(x)) = 0 \qquad V(x_f) = 0$$

Hamiltonian $\quad H(x, v, p) = \ell(x, v) + \langle p, v \rangle$

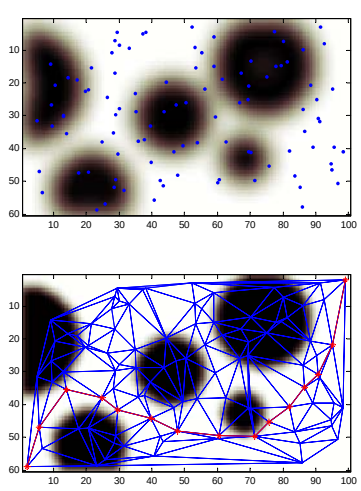
the optimal path is given by

$$\rho(t) = x_i + \int_0^t \arg\min_{v \in \mathcal{V}} H(\rho(\tau), v, \nabla_x V(\rho(\tau)))d\tau$$

Numerical methods based on the maximum principle also run into difficulties because of local minima

Discretization approach

UCSB



50 nodes, 300 edges
cost-matrix comp. = .17 sec
optimization = .03 sec

- ✓ extract a finite set $\mathcal{X}$ of points
- ✓ restrict the search to piecewise linear paths between points in $\mathcal{X}$

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

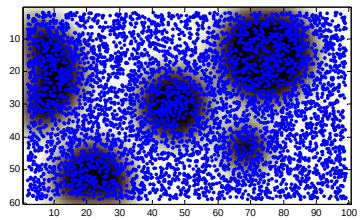
with

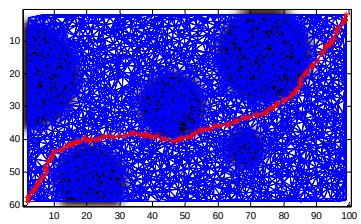
$$0 = t_0 < t_1 < \cdots < t_m = T$$
$$x_k \in \mathcal{X} \quad \forall k \in \{0, 1, \ldots, m\}$$

☺ original continuous problem was converted into an optimization on a finite graph

☹ at the expense of a higher cost

Discretization approach





5000 nodes, 30000 edges
cost-matrix computation = 16 sec
optimization = 13 sec

- ✓ extract a finite set $\mathcal{X}$ of points
- ✓ restrict the search to piecewise linear paths between points in $\mathcal{X}$

$$\rho(t) = x_{k-1} + \frac{x_k - x_{k-1}}{||x_k - x_{k-1}||}(t - t_{k-1})$$

$$\forall t \in [t_{k-1}, t_k],\ k \in \{1, 2, \ldots, m\}$$

with

$$0 = t_0 < t_1 < \cdots < t_m = T$$

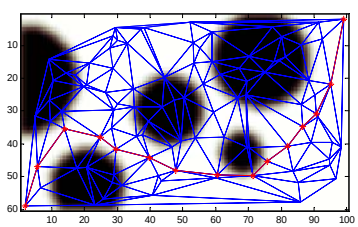
$$x_k \in \mathcal{X} \quad \forall k \in \{0, 1, \ldots, m\}$$

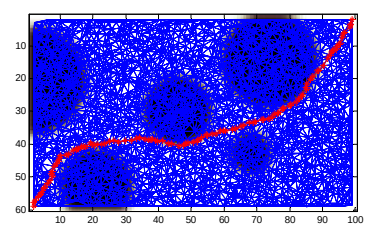
☺ original continuous problem was converted into an optimization on a finite graph

☹ at the expense of a higher cost

☺ the cost penalty can be mode arbitrarily small

Suboptimality





Theorem: For every ϵ, $A > 0$ there exists a finite set $\mathcal{X}$ of N_ϵ points such that

$$\min_{\rho \in \mathcal{P}_{\mathcal{X}}} J[\rho] - \min_{\rho \in \mathcal{P}_{||\ddot{\rho}|| \le A}} J[\rho] \le \epsilon$$

optimal cost over piecewise linear paths

optimal cost over all paths with acceleration bounded by A

but less cost penalty requires more points

Optimized sampling

Theorem: For every ϵ, $A > 0$ there exists a finite set $\mathcal{X}$ of N_ϵ points such that

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

One can show that the cost penalty ϵ is proportional to

$$g_{x,k}\epsilon_x + g_{v,k}\epsilon_v + \frac{g_{x,k} + g_{v,k}A}{2}\|x_k - x_{k-1}\|$$

norm of the cost gradient around the points

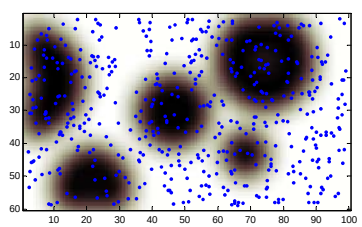
distance between consecutive points

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

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

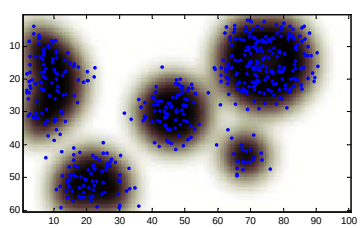
one can under-sample where gradient is low and over-sample where gradient is high

Sampling methods



Uniform sampling

extract points randomly over the whole region, with uniform probability



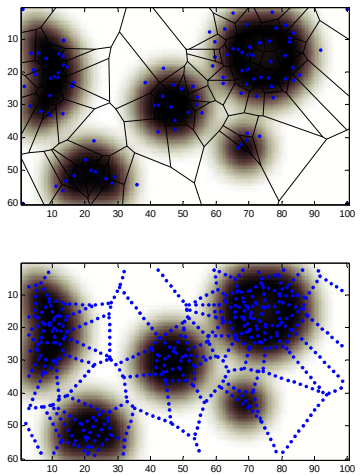
background color ≡ gradient-norm

Gradient sampling

extract points randomly over the whole region, with probability proportional to the norm of the gradient

sampling over a regular grid is much worse for anisotropic costs and will not be reported here (not enough directions!)

Optimized sampling



background color ≡ gradient-norm

Honeycomb sampling

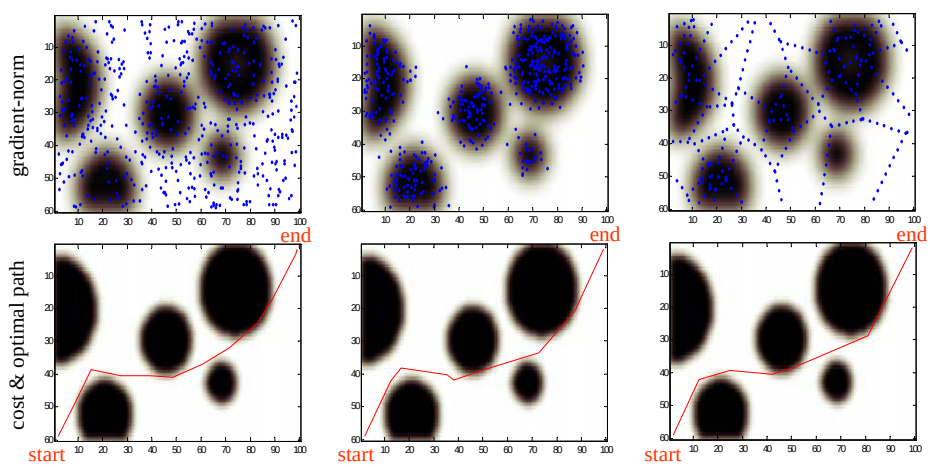
1. extract points randomly over the whole region with probability proportional to the norm of the gradient

$$g_{x,k}\epsilon_x + g_{v,k}\epsilon_v + \frac{g_{x,k} + g_{v,k}A}{2}\|x_k - x_{k-1}\|$$

2. compute the Voronoi diagram for these points
3. uniformly sample the edges of the Voronoi diagram

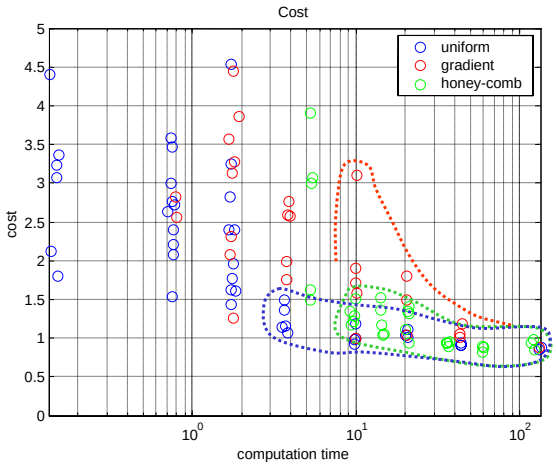
$$g_{x,k}\epsilon_x + g_{v,k}\epsilon_v + \frac{g_{x,k} + g_{v,k}A}{2}\|x_k - x_{k-1}\|$$

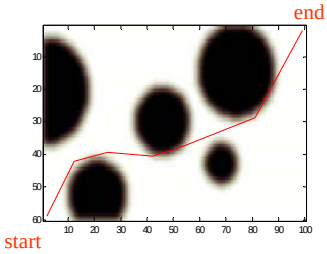
Sampling methods – spots configurations



uniform sampling | gradient-proportional sampling | honeycomb sampling

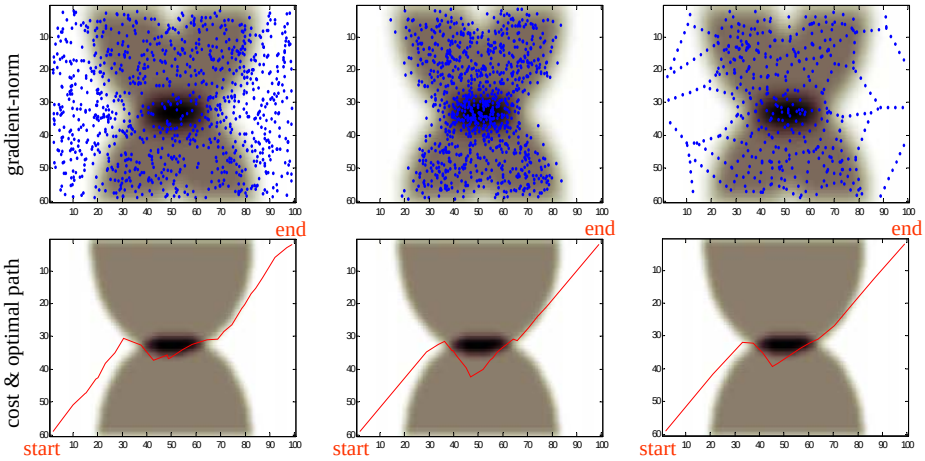
Sampling methods – spots configurations





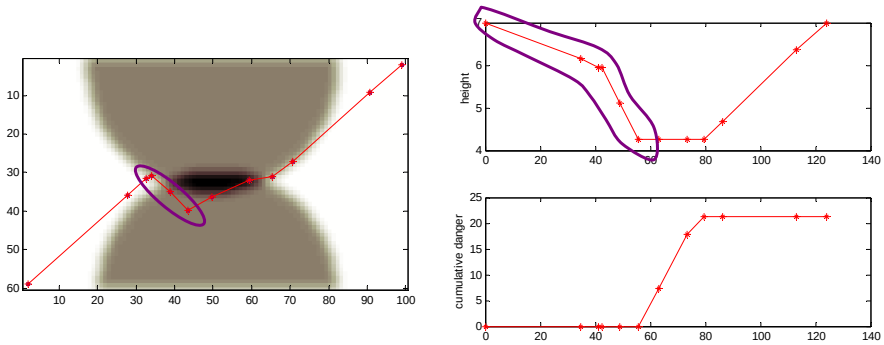
Since the optimal path can avoid danger, gradient-based methods does not help with respect to uniform sampling

Sampling methods – X configurations



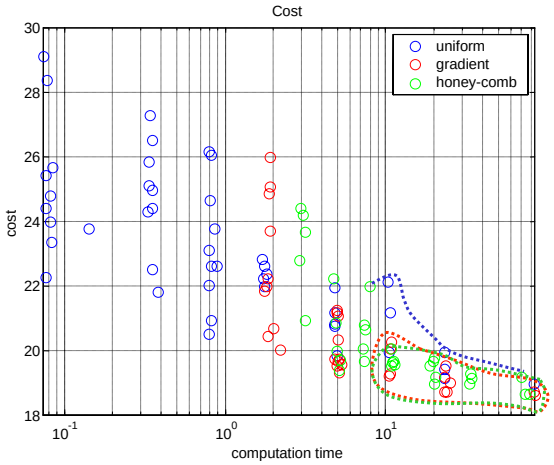
uniform sampling | gradient-proportional sampling | honeycomb sampling

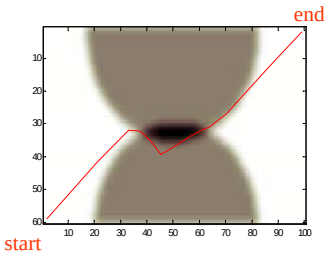
Sampling methods – X configurations



probability of radar acquisition is minimized through azimuth/elevation angles management

Sampling methods – X configurations

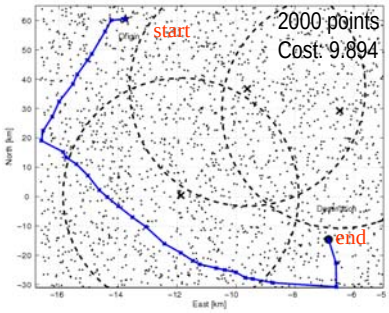




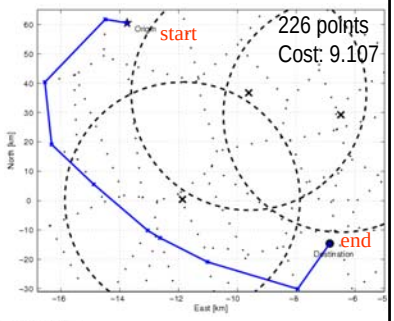
Gradient-based methods improve upon uniform sampling

honeycomb sampling is fast and provides good performance over a wide range of configurations

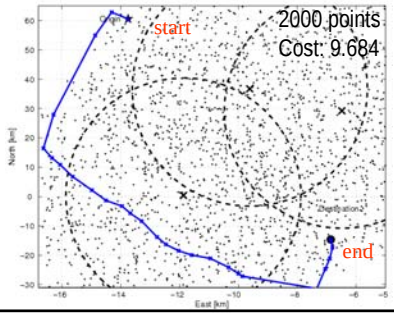
Sampling methods – complex config.



uniform sampling

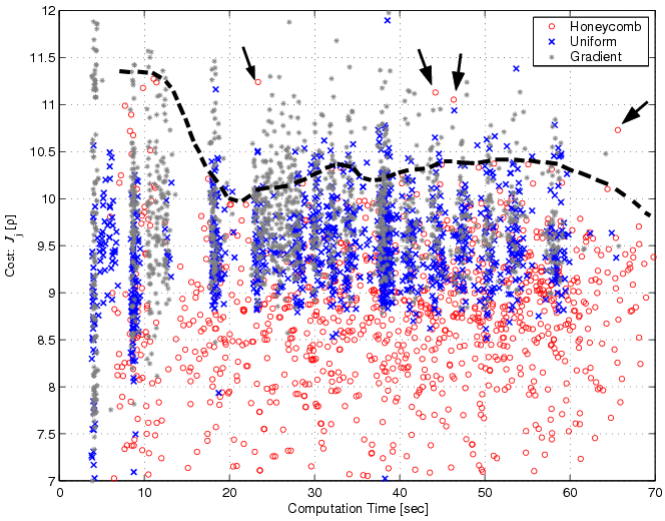


honeycomb sampling



gradient-proportional sampling

Sampling methods – complex config.



honeycomb sampling is fast and provides good performance over a wide range of configurations

Conclusions

- Proposed a method to solve anisotropic shortest path problems based on *discretization* of the continuous state-space
- Proposed a "honeycomb" *sampling algorithm* to minimize the penalty introduced
- Illustrated the potentialities of the algorithm for *minimum-risk path planning* for groups of UAVs— "honeycomb" consistently produces low-risk paths with less computation time!

Future work

- Apply this type of algorithm to other path planning problems (e.g., UAV surveillance in complex 3D environments)
- Support incremental computation to obtain *any-time* optimization algorithms (for real-time implementation)