Chance-Constrained Path Planning with Continuous Time Safety Guarantees

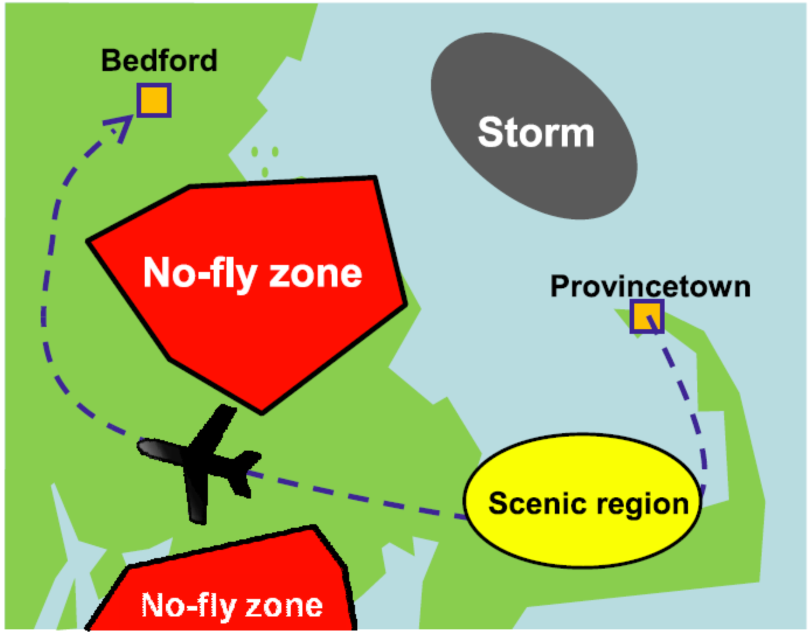
Kaito Ariu, Cheng Fang, Marcio Arantes, Claudio Toledo, and Brian Williams

Outline

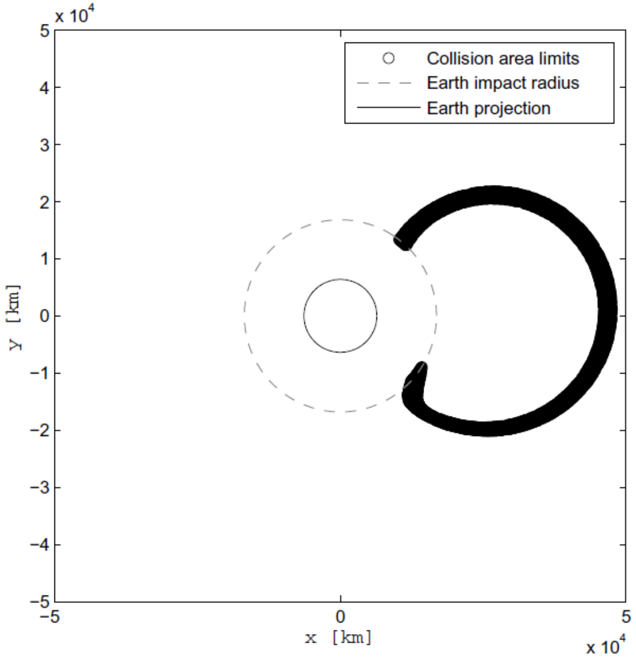
- Background (pSulu)
- Safety of trajectory mean
- Reflection Principle for trajectory safety
- Results
- Summary

Background

- Path or trajectory planning – obstacles as nonconvex constraints



Multiple goals directed traj. planning[1]



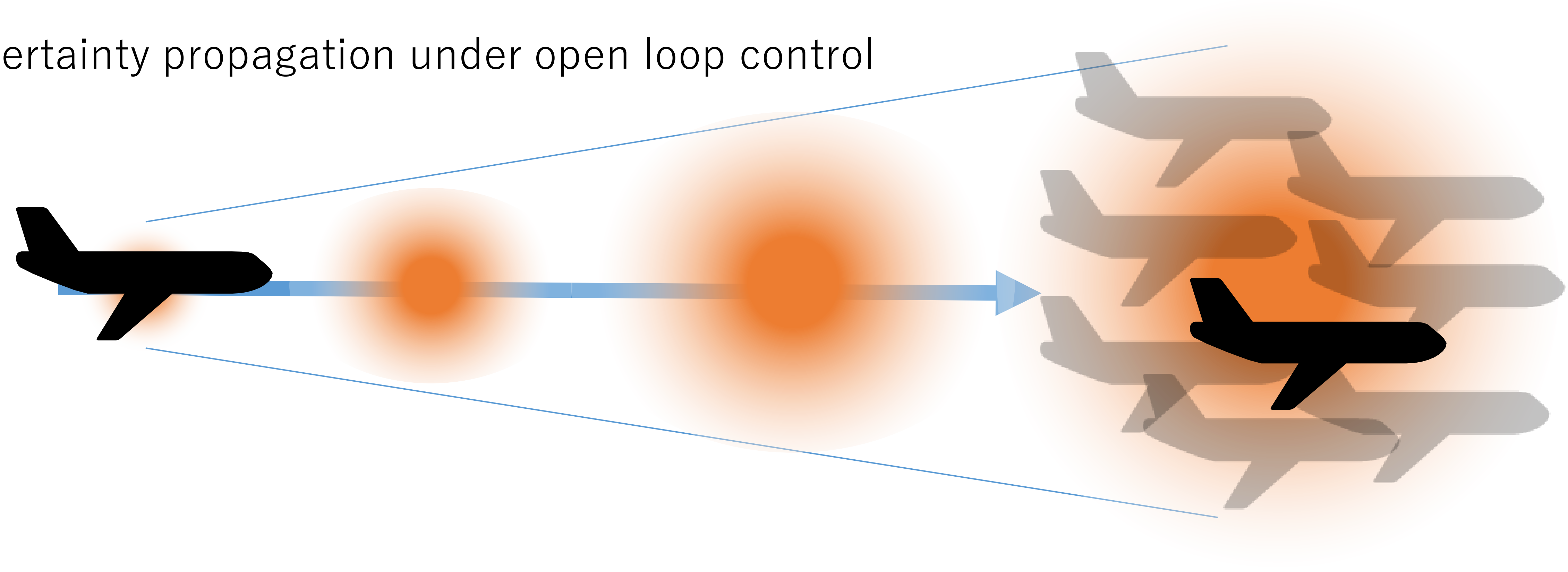
Traj. Planning on a B-plane. Black area has collision probability.
A spacecraft should avoid the area.[2]

[1] Ono, Masahiro, Brian C. Williams, and Lars Blackmore. "Probabilistic planning for continuous dynamic systems under bounded risk." *Journal of Artificial Intelligence Research* 46 (2013): 511-577.
[2] Sarli, Bruno Victorino, Kaito Ariu, and Hajime Yano. "PROCYON's probability analysis of accidental impact on Mars." *Advances in Space Research* 57.9 (2016): 2003-2012.

Background

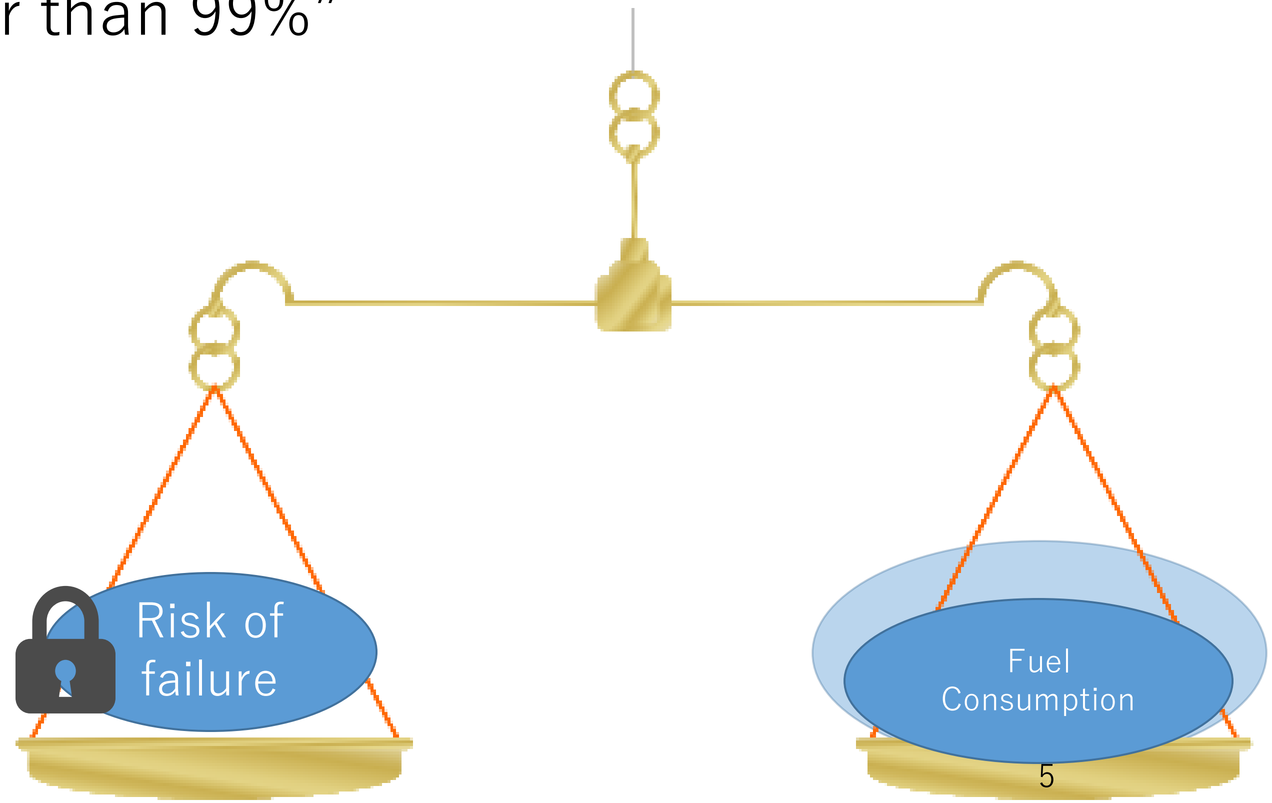
- Under Gaussian stochastic disturbances:

  Uncertainty propagation under open loop control



Key idea: chance-constraints

- Provide probabilistic guarantee: “acceptable losses”
- Optimise given risk bound:
  - “Minimise fuel consumption, s.t. probability of reaching goal safely is greater than 99%”



Problem definition

$$\min_{U} \sum_{k=1}^{T} |u_k|$$

s.t. $x_{k+1} = Ax_k + Bu_k + w_k$

$$u_{min} \leq u_k \leq u_{max}$$

$$w_k \sim N(0, \Sigma_w)$$

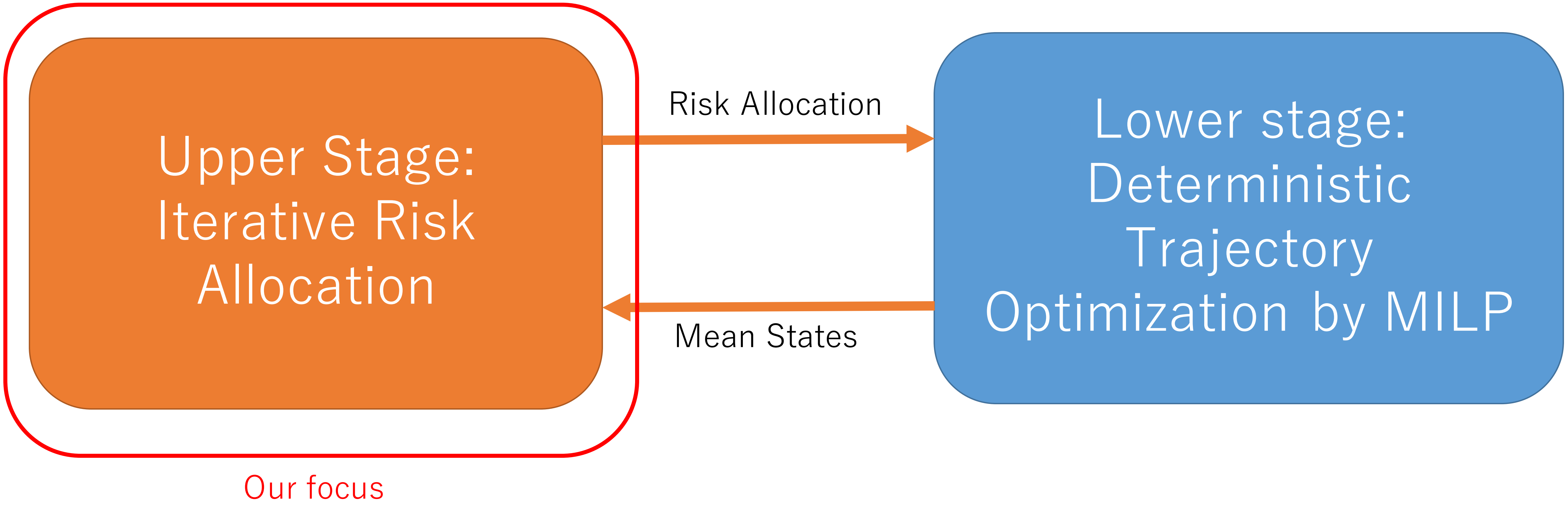
$$x_0 \sim N\left(\bar{x}_0, \Sigma_{x,0}\right)$$

$$P\left(\wedge_{k=0}^{T} \wedge_{i=1}^{N} \vee_{j=1}^{M_i} h_k^{i,j} x_k \leq g_k^i\right) \geq 1 - \Delta$$

Prior work

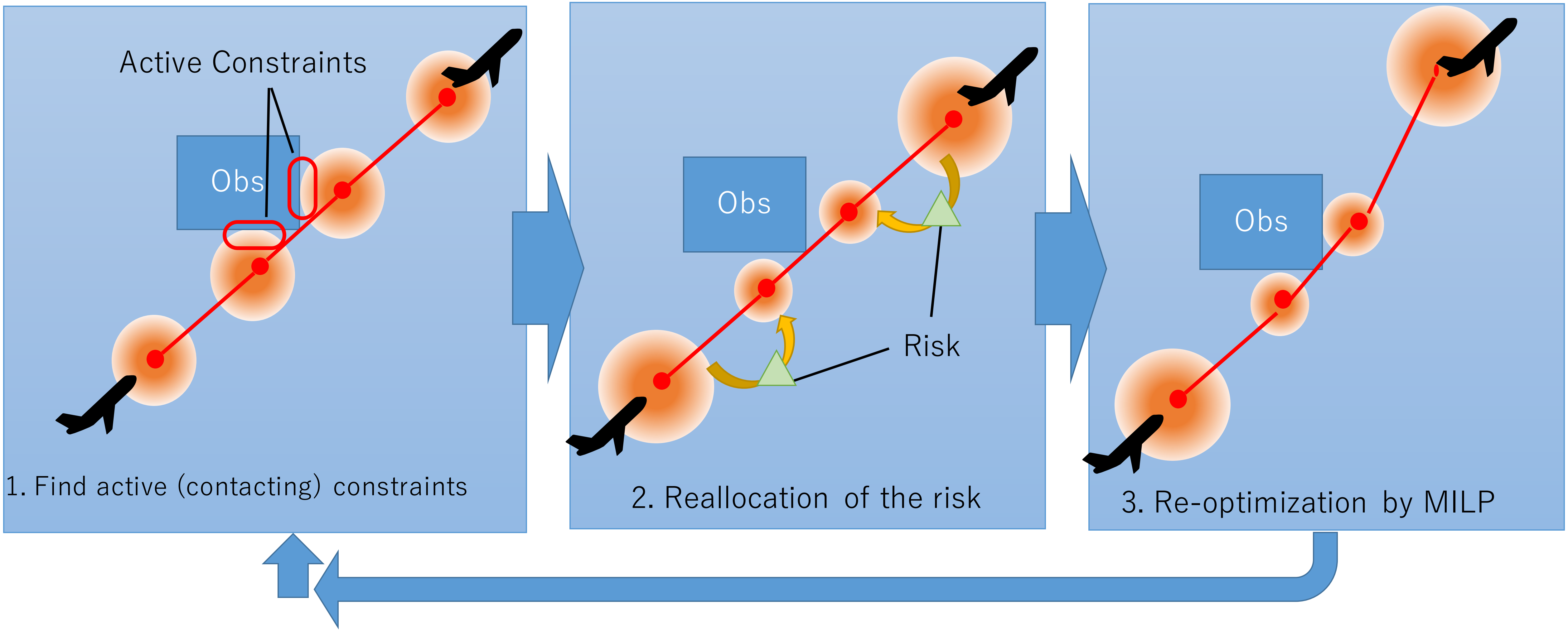
- pSulu – chance-constrained path planner
- Key insight:
  - Union bound: $P(A \cup B) \leq P(A) + P(B)$
  - Risk as resource
  - Fixed risk => MILP
  - Iterative risk allocation (IRA): redistribute risk for better solutions

Bi-stage optimization: IRA and MILP



Iteratively solving the risk allocation problem and the deterministic trajectory optimization problem, a near optimal trajectory can be produced

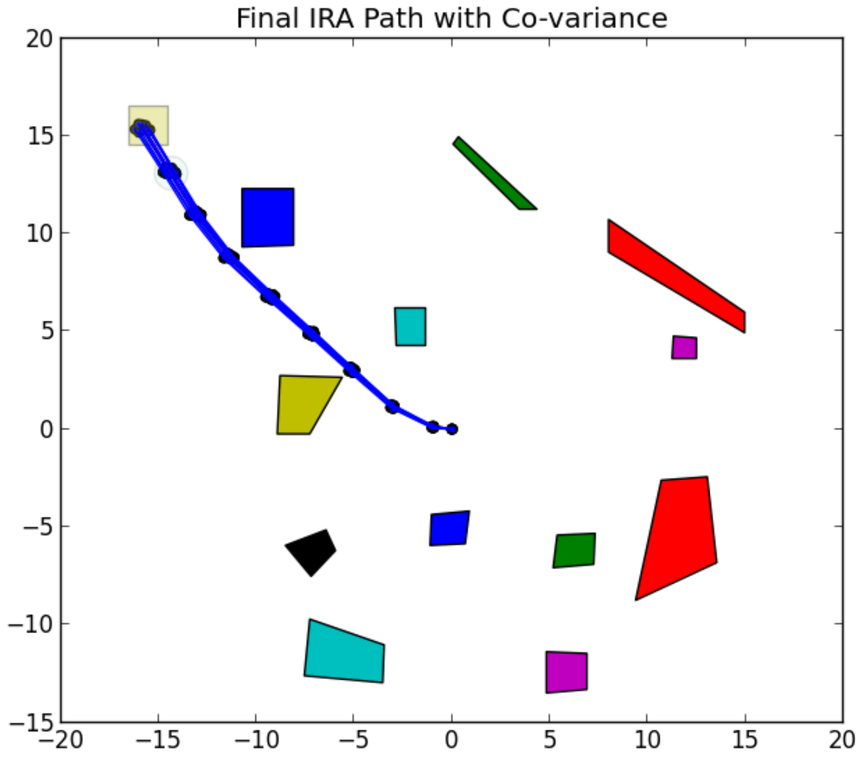
IRA: Iterative Risk Allocation

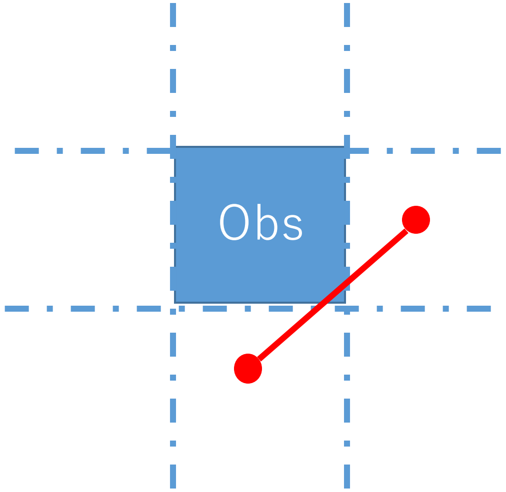


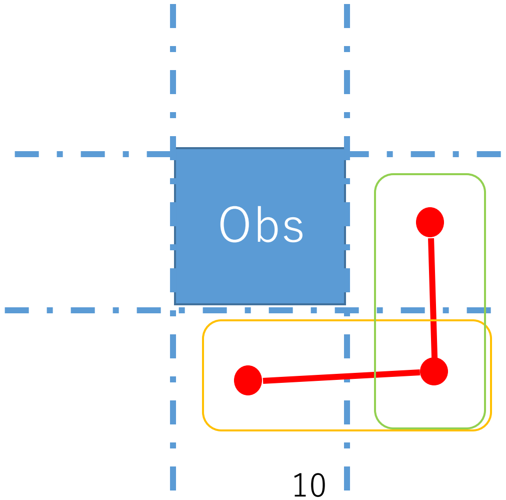
: Safety margin at each state. The risk is calculated for each state point

Safety of trajectory mean

- Recent encoding: for each obstacle, require two consecutive time steps to share an active boundary
  - Require consecutive mean points to be on the same side of obstacle
- Required assumption: Given consecutive time steps x_t, x_{t+1}, the mean state at time $x_{t+\alpha} = (1-\alpha)x_t + \alpha x_{t+1}$ for all $\alpha \in [0,1]$



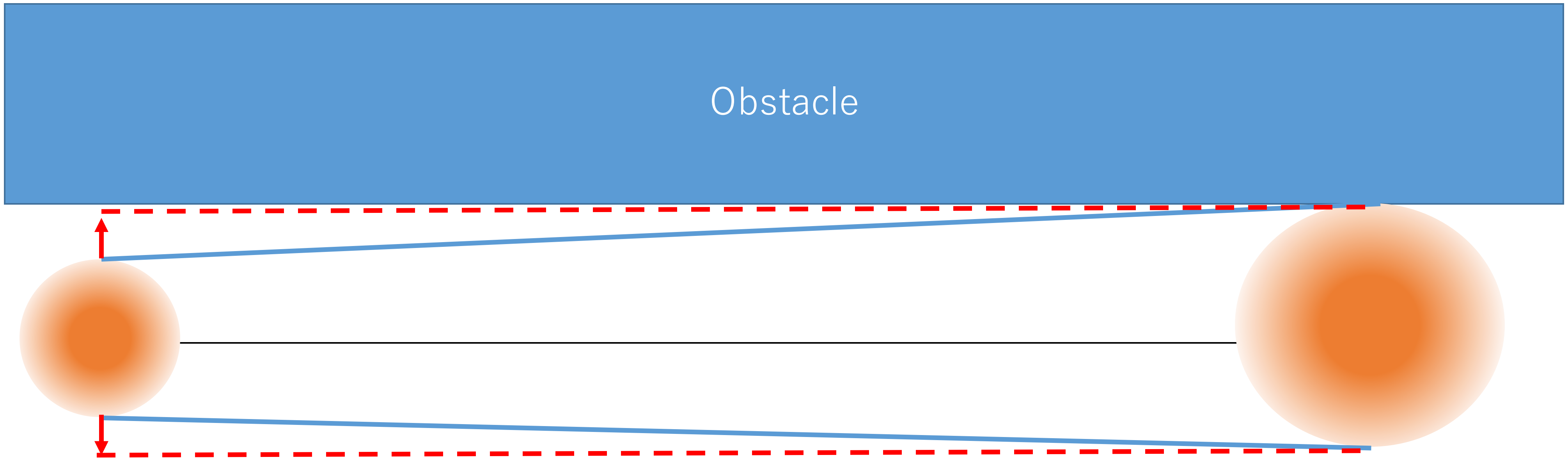




Outline

- Background (pSulu)
- Safety of trajectory mean
- Reflection Principle for trajectory safety
- Results
- Summary

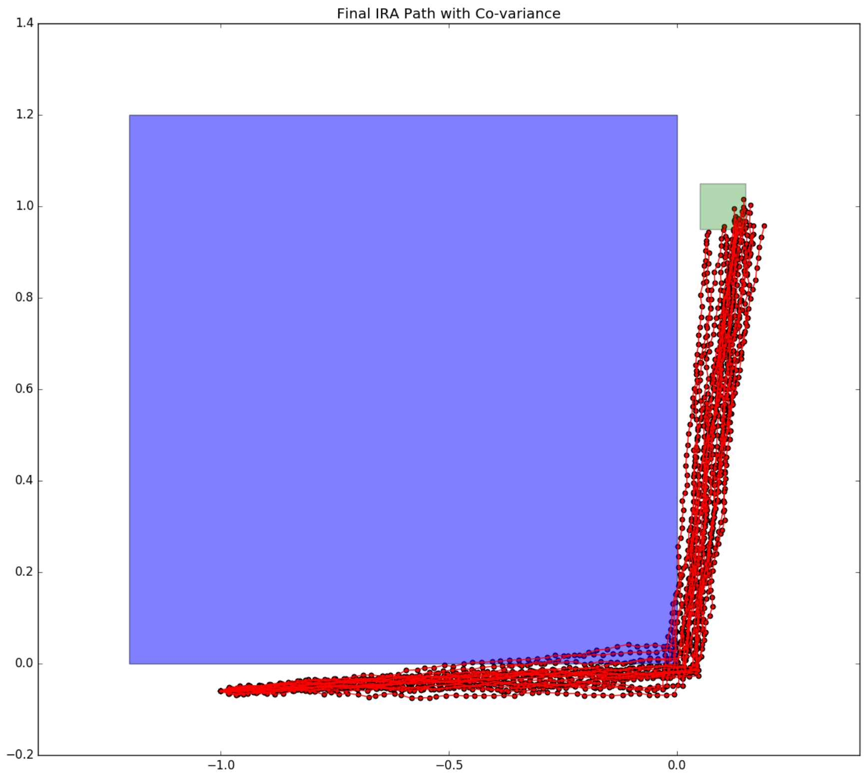
Problem of the encoding



- Prior encoding provides two guarantees:
  - Probabilistic guarantee of safety at discrete time points (same as original pSulu)
  - Guarantee that the mean trajectory is obstacle free
- Question:
  - Is this equivalent to guaranteeing continuous trajectory safety?

Numerical check

- Example problem:
  - Vehicle must round a corner and arrive at the goal area
  - Impulse velocity control
  - 3 time steps of 1s each
  - 20% risk bound
- Solution from pSulu with mean safety
- Simulation:
  - Simulated with 0.02s intervals
  - Noise scaled according to time
  - Of 10000 samples, 3491 collided with the obstacle



Intuition for numerical result

- The traversal in between time steps is important
- Even if noise is added according to the time step size used, there is a greater chance of collision
  - There are more time steps for the vehicle state
  - Hence there are more chances for a boundary crossing
- Give the above, can we still give guarantees for continuous time?

Outline

- Background (pSulu)
- Safety of trajectory mean
- Reflection Principle for trajectory safety
- Results
- Summary

Some observations

- We really wanted to plan for continuous time
- From the original formulation of pSulu problem
  - Noise is additive, Gaussian
  - Consider expected position and actual position as functions of time $\bar{x}(t)$ and $x(t)$
  - Then, deviation from the expected state is

$$\tilde{x}(t) = x(t) - \bar{x}\,(t)$$

Brownian motion

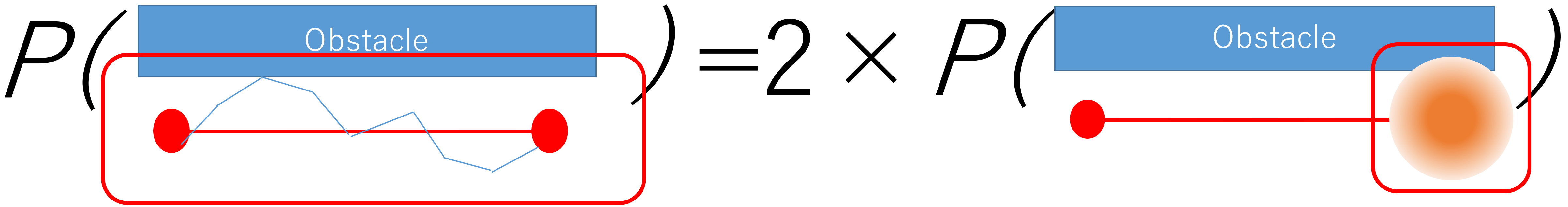
- From the model used in the original pSulu
  - Property 1: $\tilde{x}(t)$ has independent increments
  - Property 2: $K\left(\tilde{x}(t) - \tilde{x}(s)\right) \sim N(0, t - s)$ for some constant K (intuition: this is because the noise is a bunch of additive Gaussians)
- We add the following assumptions
  - Property 3: $\tilde{x}(0) = 0$ (this can be relaxed)
  - $\tilde{x}(t)$ is almost surely continuous (this is to allow for continuous time)
- Then, taking all of the above, $\tilde{x}(t)$ satisfies all the requirements for it to be a Brownian motion.
- Hence $h\tilde{x}(t)$ is a Brownian motion for vector h

The Reflection Principle

- For any Brownian motion we can apply the Reflection Principle

Reflection principle:
For the Brownian motion, $P\left(\max_{0\leq s\leq T} w(s) \geq a\right) = 2P(w(T) \geq a)$

Intuitive description:



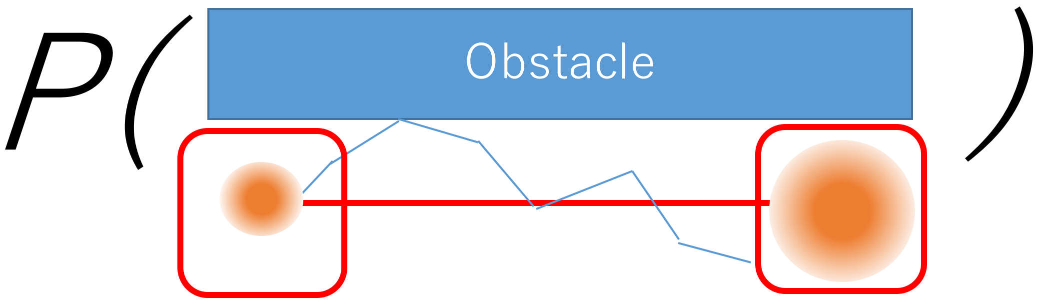
Collision probability during the traversal

Twice of the collision probability at the end point

Implementation of the reflection principle

- Current risk allocation in IRA:

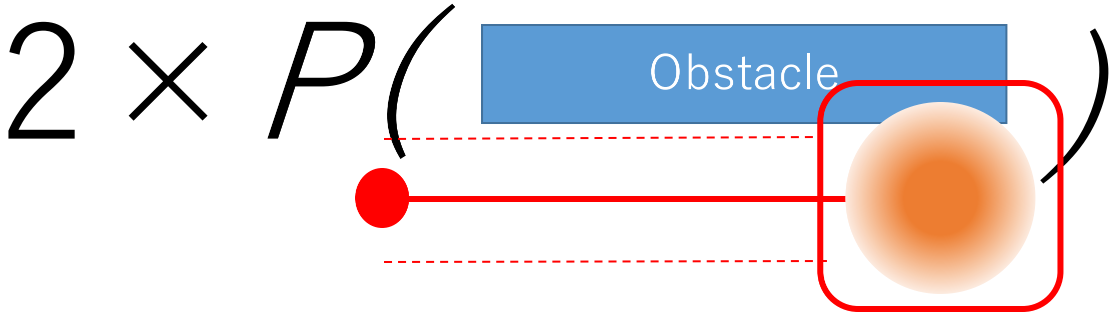
```
for each time step:
   for each side of the obstacle:
          allocate risk
          (based on the covariance of each time)
   end
end
```



Using the corresponding covariance variable
Ensures the probability for each step

- Modified risk allocation in IRA:

```
for each time segment:
   for each side of the obstacle:
          allocate risk
          (based on covariance at end time step)
   end
end
```



Twice of the collision probability at the end point
Ensures the probability for the entire path segment

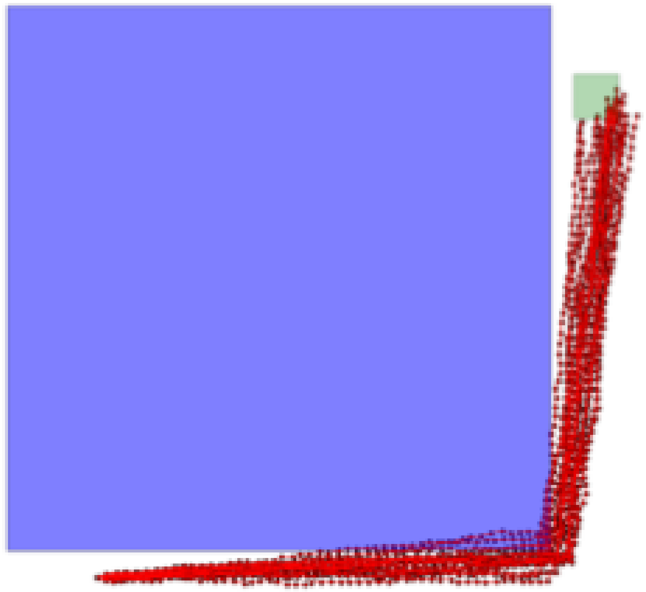
Outline

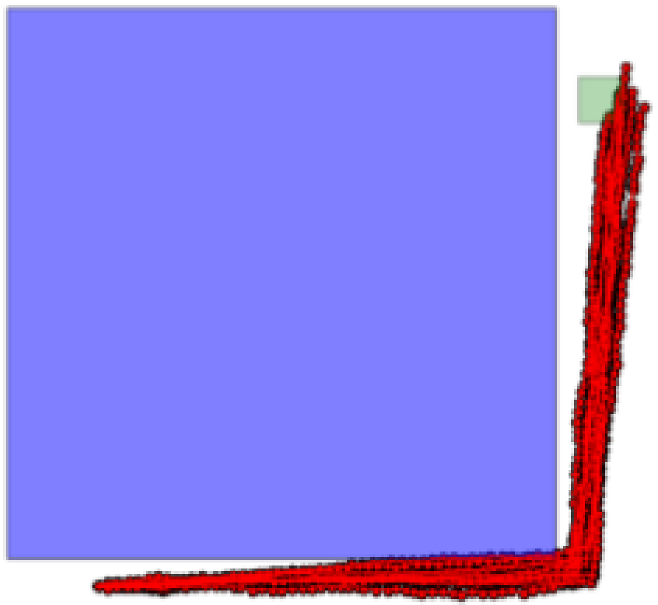
- Background (pSulu)
- Safety of trajectory mean
- Reflection Principle for trajectory safety
- Results
- Summary

Results

For the specified 20% risk:

Prior encoding



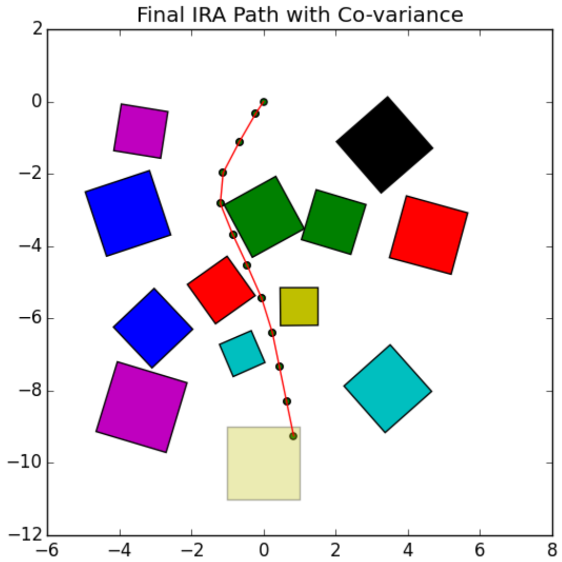


With the reflection principle

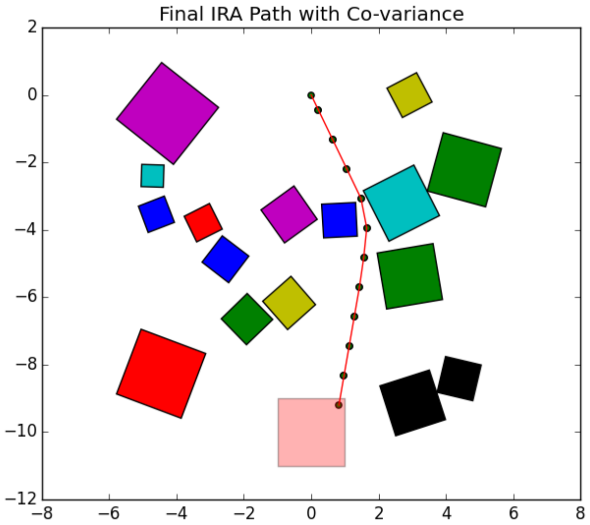
	Sim time	**Collision time**	(Nominal) Obj fun
Reflection Principle encoding	10000	621	3.011812
Discrete time encoding	10000	3491	2.906687

Results

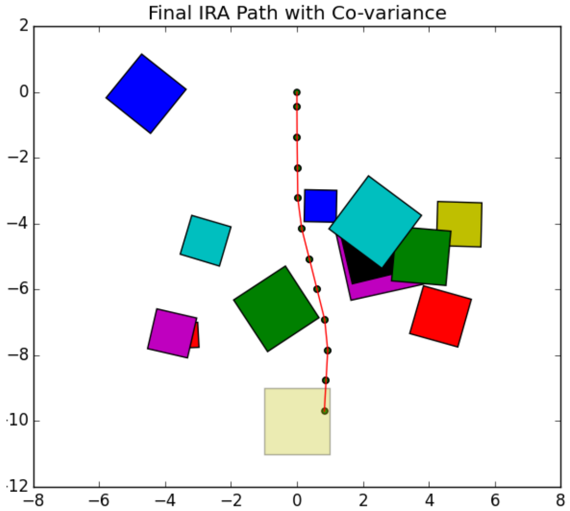
- We compared the objective function and computation times between previous algorithm and our algorithm for 4 (type of maps) × 50 (number of sample maps) = 200 maps.



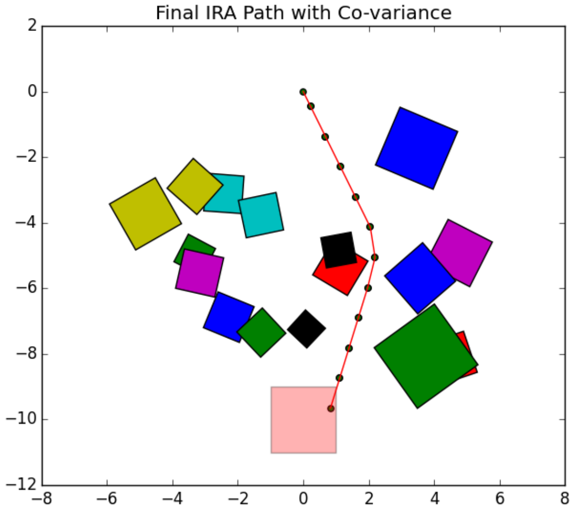
12 Obstacles × 50 maps



16 Obstacles × 50 maps



12 Obstacles w/ wrapping × 50 maps



16Obstacles w/ wrapping × 50 maps

Results

	Avg. soln. time (new:old)	Avg. obj. (new:old)	No solution maps (new)	No solution maps (old)
Map 12	0.620447432	1.020924996	0	0
Map W12	0.585377489	1.006711871	0	0
Map 16	1.726021078	1.064344226	2	0
Map W16	0.674433433	1.01014466	2	0

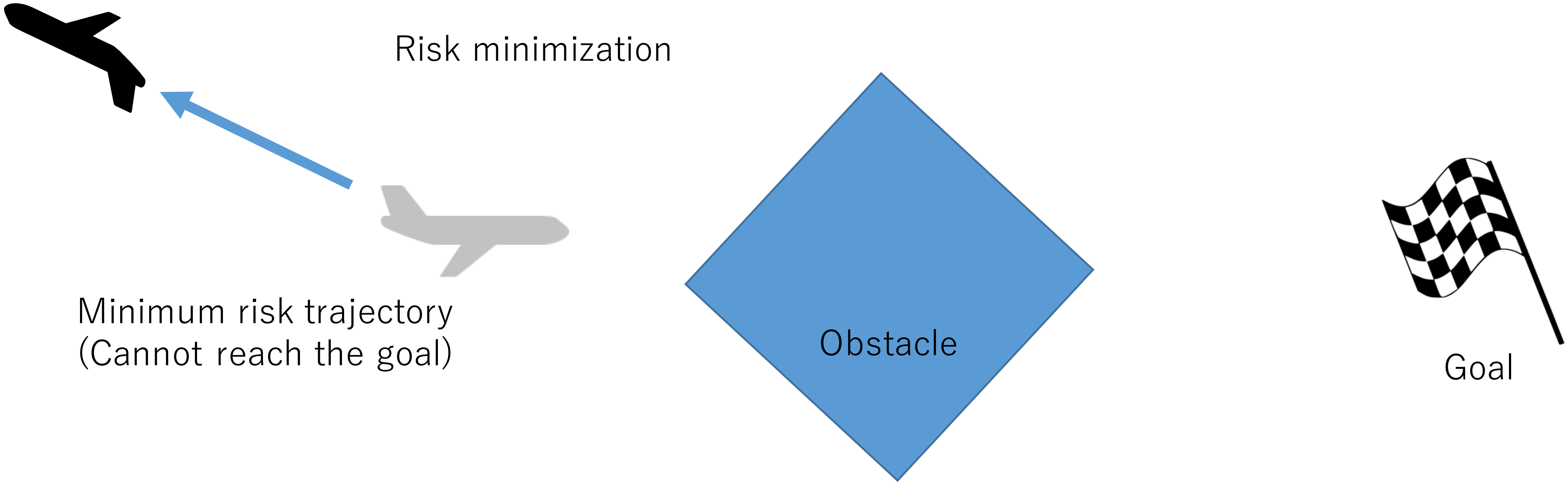
Summary

- Prior work guaranteed safety of trajectory mean and discrete time steps
- Problem actually involves a Brownian process – safety in continuous time not guaranteed
- Use Reflection principle to provide guarantees of trajectory safety
- New solution: correct, faster, not significantly worse in terms of utility
- Future work: look at nonlinear dynamics
  - No longer Brownian motion, but what concentration inequalities can we use?

Appendix

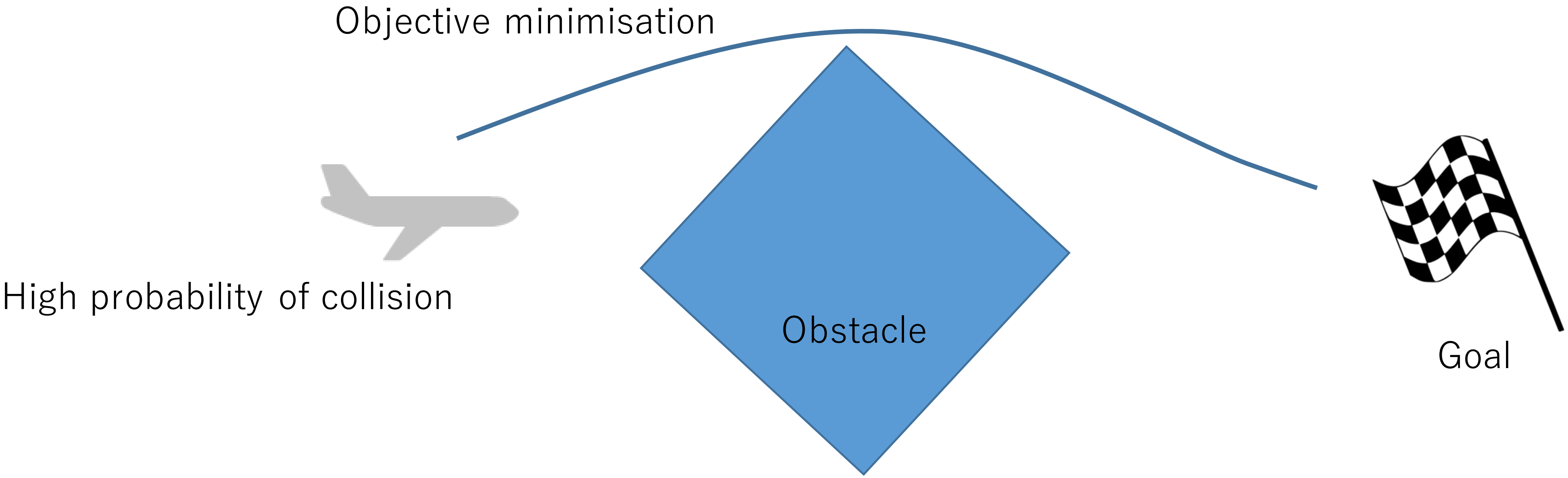
Background

- Trade off between risk and objective function (eg distance)



Background

- Trade off between risk and objective function (eg distance)

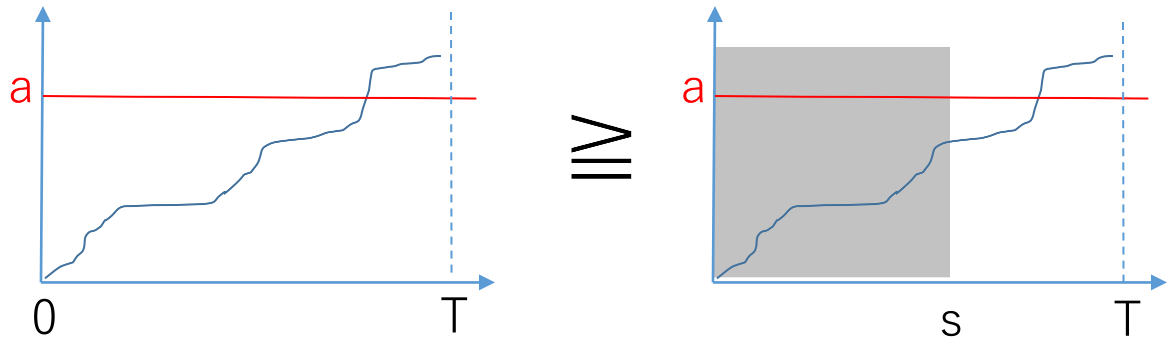


Why chance-constraints in general?

- Alternative approach: min expected loss
- Problematic when:
  - Difficult to characterise objective function (loss of science data during science surveys, cascading delays in airport scheduling)
  - Infinite penalty for loss (unique vehicles)

Relaxing Property 3

- Consider Brownian Motion $w(s)$, $w(0) = 0$ by definition
- We know that $P\left(\max_{0\le s\le T} w(s) \ge a\right) = 2P(w(T) \ge a)$
- $$P\left(\max_{S_0\le s\le T} w(s) \ge a\right) \le P\left(\max_{0\le s\le T} w(s) \ge a\right)$$
  $$= 2P(x(T) \ge a)$$
- This gives a conservative approximation and is what we use for segments



Why risk allocate over time step?

- From Reflection Principle
  - Only need to consider covariance at the last time step
  - We tried to take away risk allocation to time segments
    - Motivation:
      - We would then no longer break up the risk over so many steps, maybe less conservatism
      - Risk allocation still there – over the most relevant corners
    - Result: More conservative than allocating to time segments
      - Although we're collapsing some of the risk allocations together, we are still working with a changing mean in position over time – allocation over time segments makes sense