LISC

LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

An Information Potential Approach for Tracking and Surveilling Multiple Moving Targets using Mobile Sensor Agents

W. Lu, G. Zhang and S. Ferrari
Laboratory for Intelligent Systems and Control (LISC)
Department of Mechanical Engineering and Materials Science
Duke University

R. Fierro and I. Palunko
Multi-Agent, Robotics, Hybrid, and Embedded Systems Laboratory (MARHES)
Department of Electrical and Computer Engineering, University of New Mexico

SPIE Defense, Security, and Sensing Conference, Orlando (FL)

April 25-29, 2011

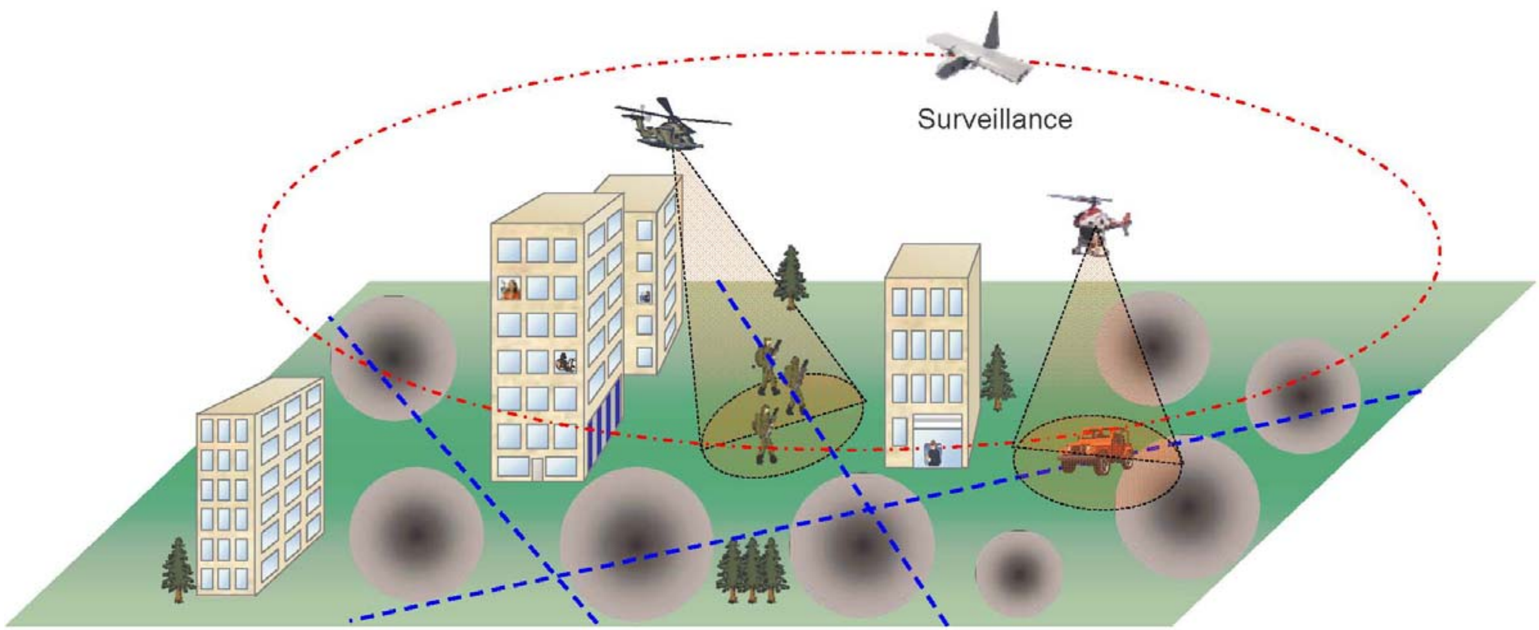
Introduction and Motivation

➢ Research Objectives

-- Online motion planning and control for mobile sensor agents (MSAs)

-- MSA Objectives: target tracking and surveillance; minimize energy consumption (e.g. distance traveled); and, avoid obstacles.

❖ Applications: Modern Surveillance Systems



Modeling of Targets and Mobile Sensors

Modeling of Target Dynamics

- Target state: $\mathbf{X}(t_k) = [\mathbf{x}(t_k) \quad \theta(t_k)]^T, \quad \mathbf{x}(t_k) = [x(t_k) \quad y(t_k)]^T$

- Markov motion process:

$$\dot{\mathbf{x}}(t) = v(t_k)[\cos\theta(t_k)\ \sin\theta(t_k)]^T, \quad \text{for} \quad t_k < t < t_{k+1}, \quad \text{where:}$$

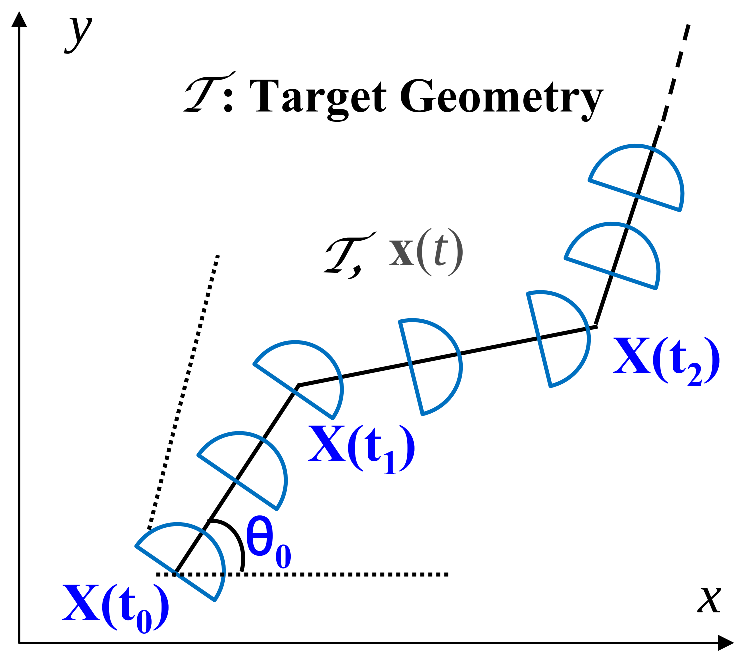
State transition at discontinuities:

$$\mathbf{X}(t_{k+1}) = \mathbf{X}(t_k) + \begin{bmatrix} \cos\theta(t_k)\tau v \\ \sin\theta(t_k)\tau v \\ N(\mu, \sigma^2) \end{bmatrix}$$

Target transition probabilities,

$$\Pr(\mathbf{x}(t_k), v(t_k), \theta(t_k)), \ k = 1, 2 \ldots,$$

are computed from sensor measurements.

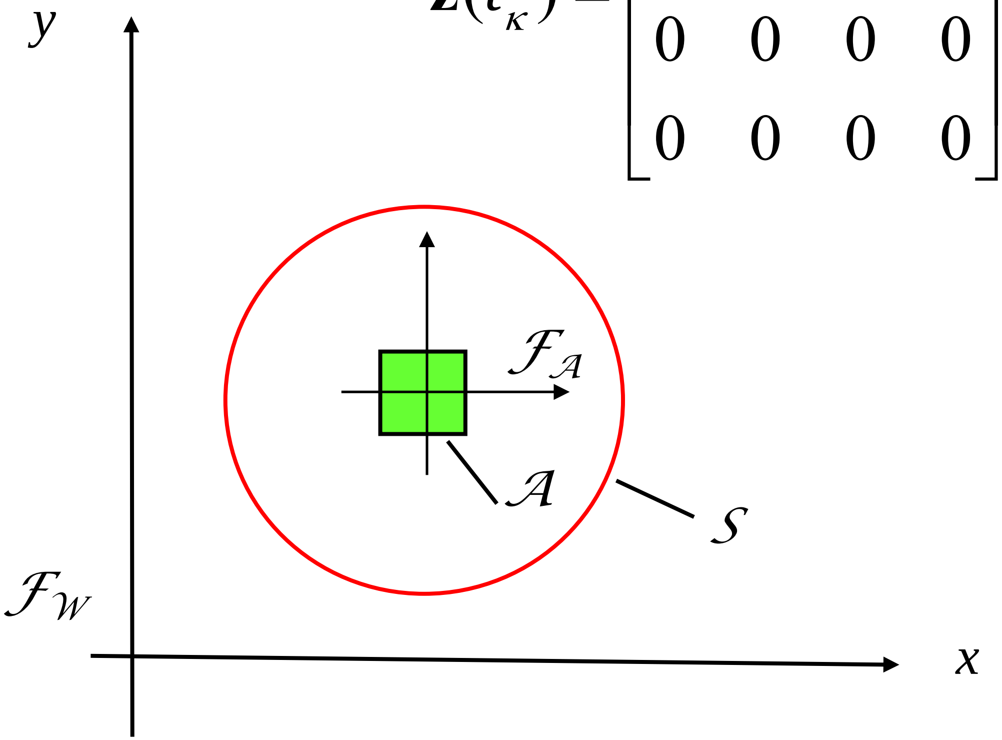


Model of Mobile Sensor Agent (MSA)

The sensor is characterized by a field-of-view (FOV) with geometry $\mathcal{S}$, and by a platform with geometry $\mathcal{A}$.
When the $\mathcal{S} \cap \mathcal{T} \neq \emptyset$, measurements are obtained according to equation:

$$\mathbf{z}(t_\kappa) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(t_\kappa) + \boldsymbol{\upsilon}(t_\kappa)$$

white-noise error



Geometries:
Sensor FOV, $\mathcal{S}$
Sensor Platform, $\mathcal{A}$

Frames of reference:
Body frame, $\mathcal{F}_\mathcal{A}$
Inertial frame, $\mathcal{F}_W$

Model of MSA's Dynamics

The sensor position and velocity in $\mathcal{F}_{\mathcal{W}}$ are described by the vector,

$$\mathbf{y}(t_\kappa) = [x(t_\kappa)\ y(t_\kappa)\ \dot{x}(t_\kappa)\ \dot{y}(t_\kappa)]^T$$

and the platform dynamics are assumed to be LTI, and discretized w.r.t. time:

$$\mathbf{y}(t_\kappa) = \begin{bmatrix} 1 & 0 & \tau & 0 \\ 0 & 1 & 0 & \tau \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{y}(t_{\kappa-1}) + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \tau & 0 \\ 0 & \tau \end{bmatrix} \mathbf{u}(t_{\kappa-1})$$

Where, $\mathbf{u}(t_\kappa)$ is the sensor's platform feedback control input vector.

Sensor Objectives:
Target Tracking and Surveillance

Target Tracking: Particle Filter Method

The particle filter is a recursive method to estimate a probability density function (PDF), e.g. $f(\theta(t_\kappa) \mid \mathbf{z}(t_{0\to\kappa}))$, based on sequential Monte Carlo simulations.

- Each iteration has three steps
  - Sampling particles from an importance density function (IPD): $q(\theta)$
  - Update the weight for each particle using Bayes' rule:

$$w_p \propto \frac{f(\theta(t_\kappa) \mid \mathbf{z}(t_{0\to\kappa}))}{q(\theta_p)},$$

$$f(\theta_p \mid \mathbf{z}(t_{0\to\kappa})) \propto f(\mathbf{z}(t_\kappa) \mid \theta_p, \mathbf{z}(t_{0\to\kappa-1})) \times f(\theta_p \mid \mathbf{z}(t_{0\to\kappa-1}))$$

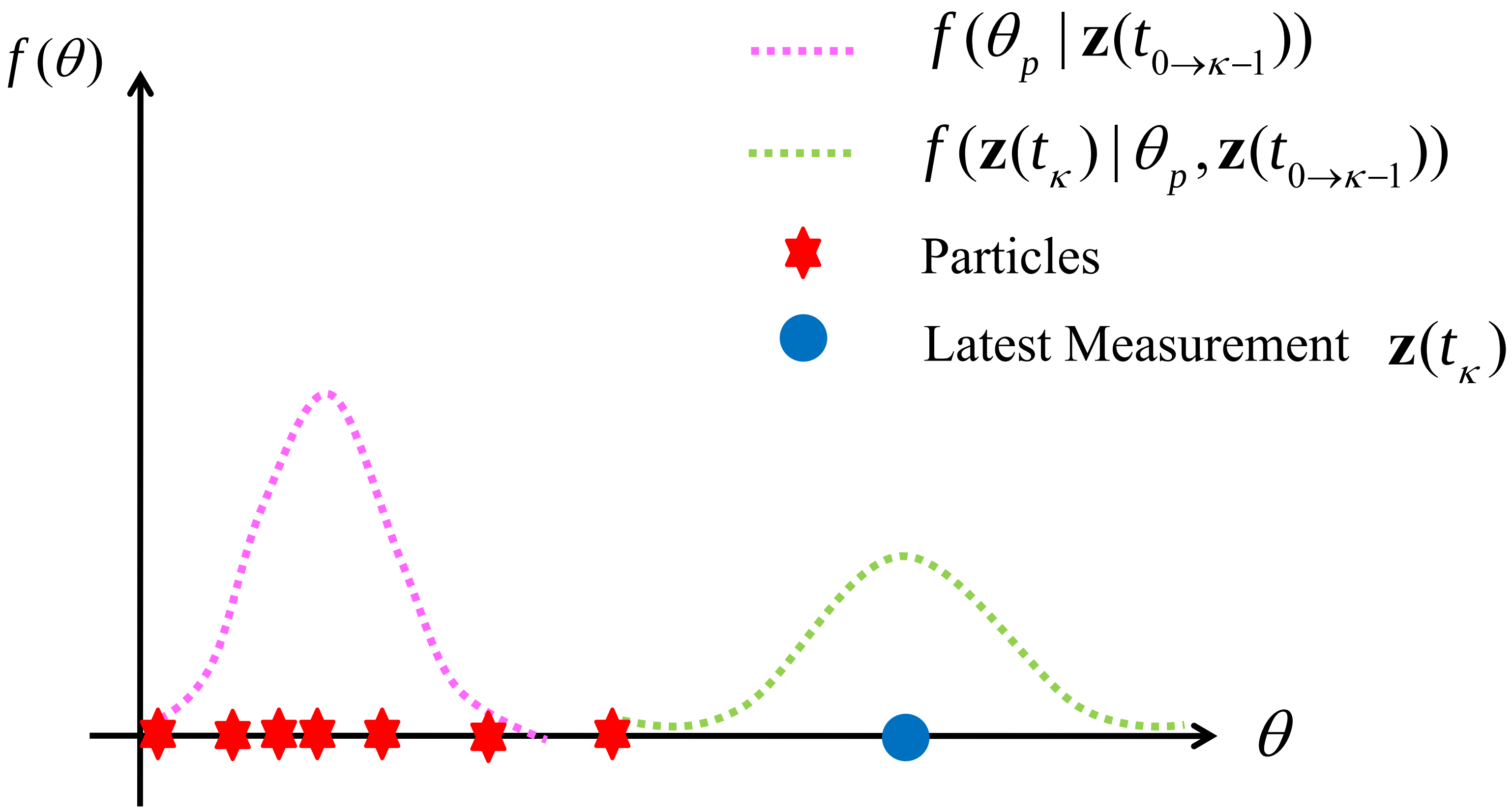
  - Re-sampling if effective size N_e is smaller than $\frac{N}{2}$

$$N_e = \frac{1}{\sum_{p=1}^{N} w_p^2}$$

- PDF representation: $\sum_{p=1}^{N} w_p \delta(\theta_p), \quad \sum_{p=1}^{N} w_p = 1, \quad w_p > 0$

Target Tracking: Particle Filter Update

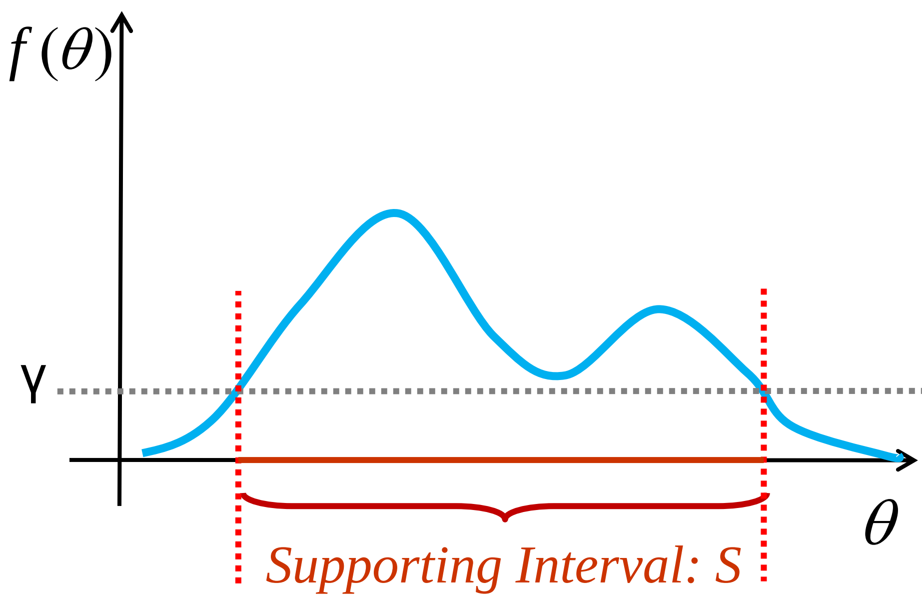
Particle filter method applied to target heading (θ) estimation:



Importance Probability Density Function

Definition of important probability density (IPD) function:

$$f_s(\theta) = \begin{cases} \frac{1}{L} & \text{if } \theta \in S \\ 0 & \text{else} \end{cases} \qquad L = \int_R g(\theta)d\theta \qquad g(\theta) = \begin{cases} 1 & \text{if } \theta \in S \\ 0 & \text{else} \end{cases}$$

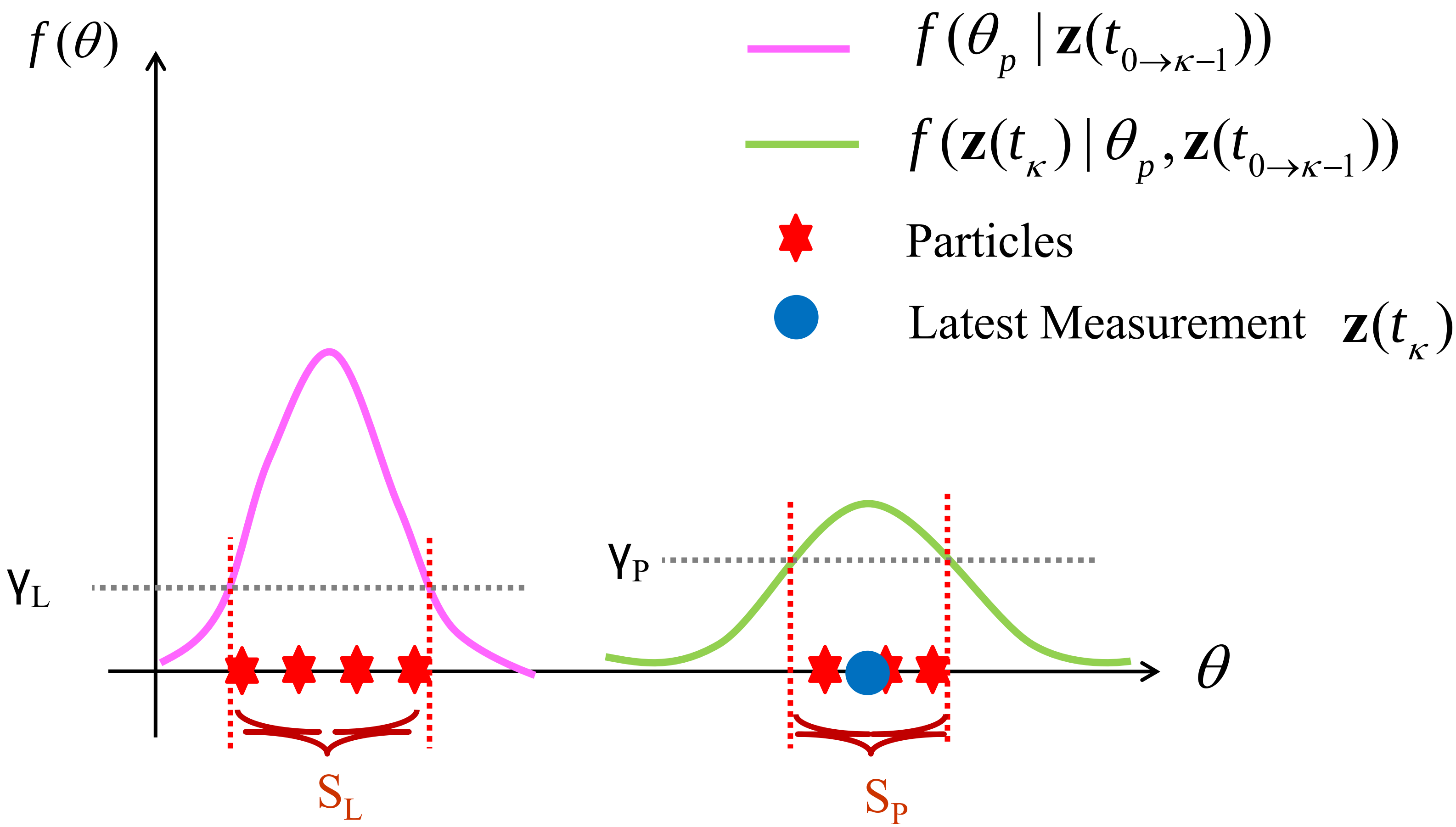


Approximate $f_S(\theta)$ by a finite mixture of Gaussians, $\sum_{i=1}^{m} \pi_i N(\mathbf{u}_i, \sigma_i^2)$, i.e.:

$$S_i : \left\{ \theta \,\middle|\, N\left(\theta \mid \mathbf{u}_i, \sigma_i^2\right) > \frac{\gamma}{\pi_i} \right\}$$

$$S = S_1 \cup S_2 \ldots \cup S_m$$

Particle Filter Update by Proposed IPD



$$S = S_L \cup S_P$$

Gaussian-Mixture Representation

- **Particle filter representation:** $\sum_{p=1}^{N} w_p \delta(\theta_p), \quad \sum_{p=1}^{N} w_p = 1, \quad w_p > 0$

⇩

- **Degeneracy phenomenon:** The variance of particle weight accumulates along iterations

⇩

- **Consequence:** A number of particles have low weights and no contributions in approximating

⇩

- **Re-sampling method:** The particles with high weights are repeated, redundant particles

- **Gaussian Mixture:** $\sum_{i=1}^{m} \pi_i \, \mathrm{N}(u_i, \sigma_i^2)$

Information Potential Field Method

❖ **Classical Potential Field Method for Robot Navigation and Control:**

$$U(\mathbf{q}) = U_{att}(\mathbf{q}) + U_{rep}(\mathbf{q})$$

$$\mathbf{F} \propto -\nabla U(\mathbf{q})$$

❑ **Information Potential Field Method for MSA Navigation:**

$$U(\mathbf{q}) = \begin{cases} \frac{1}{2}\eta_t\left(\frac{1}{\rho(\mathbf{q_t},\mathbf{q})} - \frac{1}{\rho_0}\right)^2 & \text{if } \rho(\mathbf{q_t},\mathbf{q}) \leq \rho_0 \\ 0 & \text{if } \rho_0 < \rho(\mathbf{q_t},\mathbf{q}) < \rho_1 \\ \frac{1}{2}\xi_t(\rho(\mathbf{q}^*,\mathbf{q}) - \rho_1)^2 & \text{if } \rho(\mathbf{q_t},\mathbf{q}) \geq \rho_1 \end{cases}$$

$$\mathbf{q}^* = \mathbf{q}_t + \alpha \begin{bmatrix} \cos\tilde{\theta}^s \\ \sin\tilde{\theta}^s \end{bmatrix} (\rho(\mathbf{q}_t,\mathbf{q}_s) - \rho_1)$$

❑ **Information Potential Field (IPF) - MSA Control Law:**

$$\mathbf{u}(\mathbf{q}) = v\begin{bmatrix} \cos\tilde{\theta}(t_\kappa) \\ \sin\tilde{\theta}(t_\kappa) \end{bmatrix} - \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\mathbf{y}(t_\kappa) - \nabla U(\mathbf{q})$$

Stability of IPF Control Law

❖ Proposed Lyapunov Function for IPF Control:

Wenjie: Please write on this slide the proof and eq. (30)-(31) from the paper.

Simulations and Results

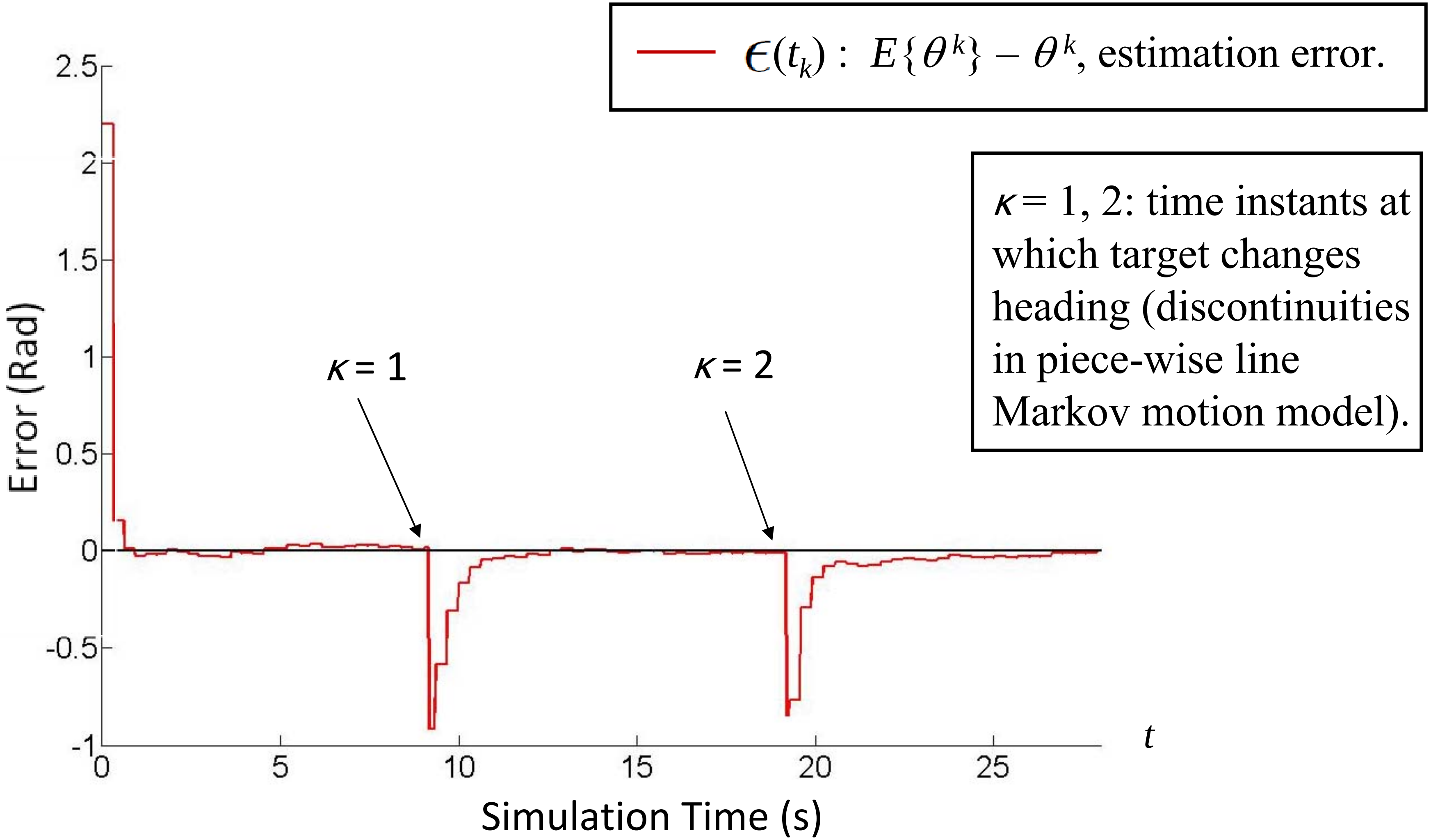
Simulated Scenario # 1

Scenario: The sensor platform and the target are modeled as point masses, the workspace has no obstacles

Simulation parameters:

Parameter	Value
Target speed (v)	2m/s
Workspace size	50m×50m
Heading changing frequency	0.1Hz
Measuring frequency	3.3Hz
Measuring noise	diag (0.4 0.4)

Results: Particle-Filter Target Tracking



$\kappa = 1, 2$: time instants at which target changes heading (discontinuities in piece-wise line Markov motion model).

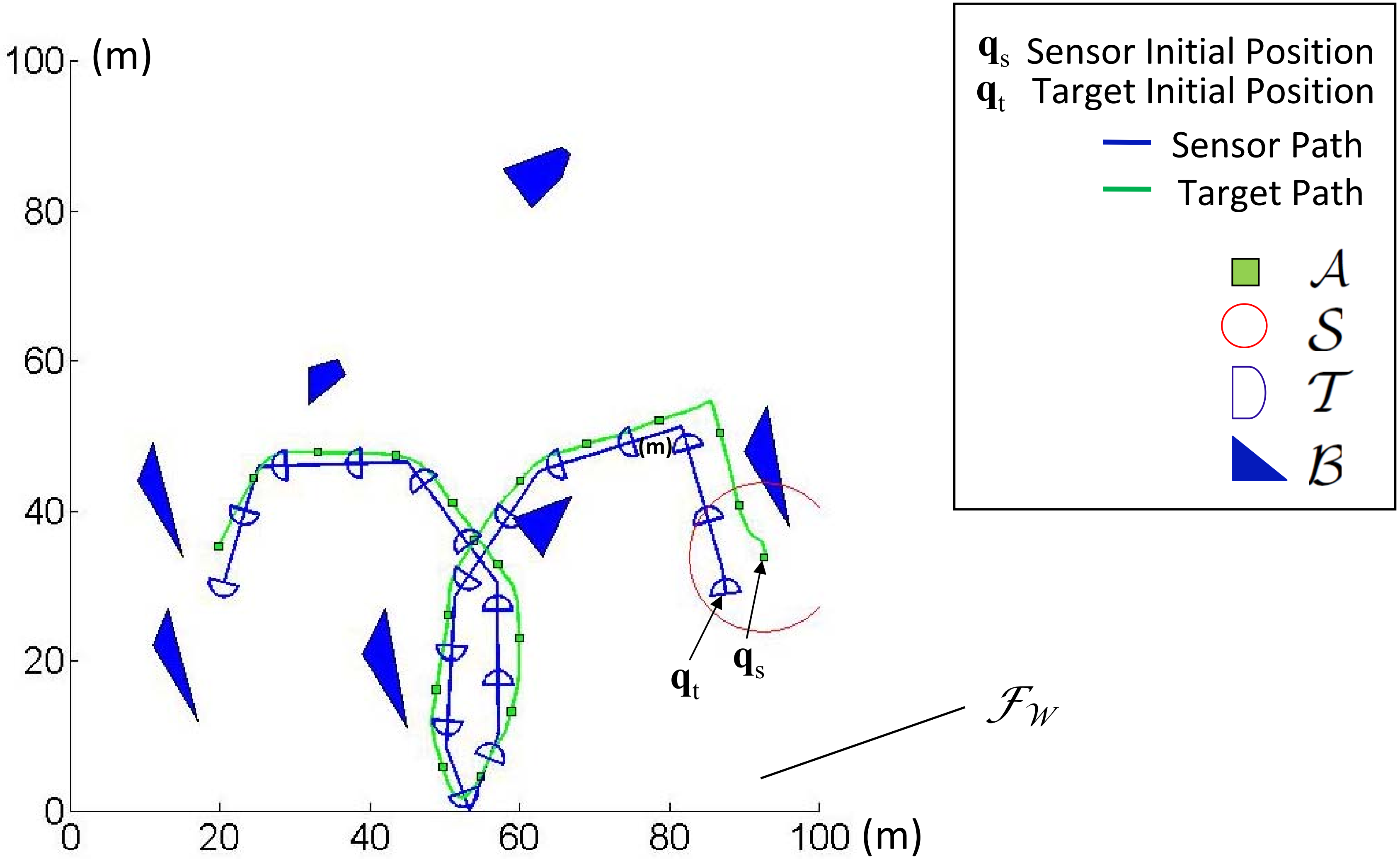
Simulated Scenario # 2

Scenario: The workspace is populated with 7 obstacles; the target and the sensor's platform have finite geometries (bdd subsets of R^2).

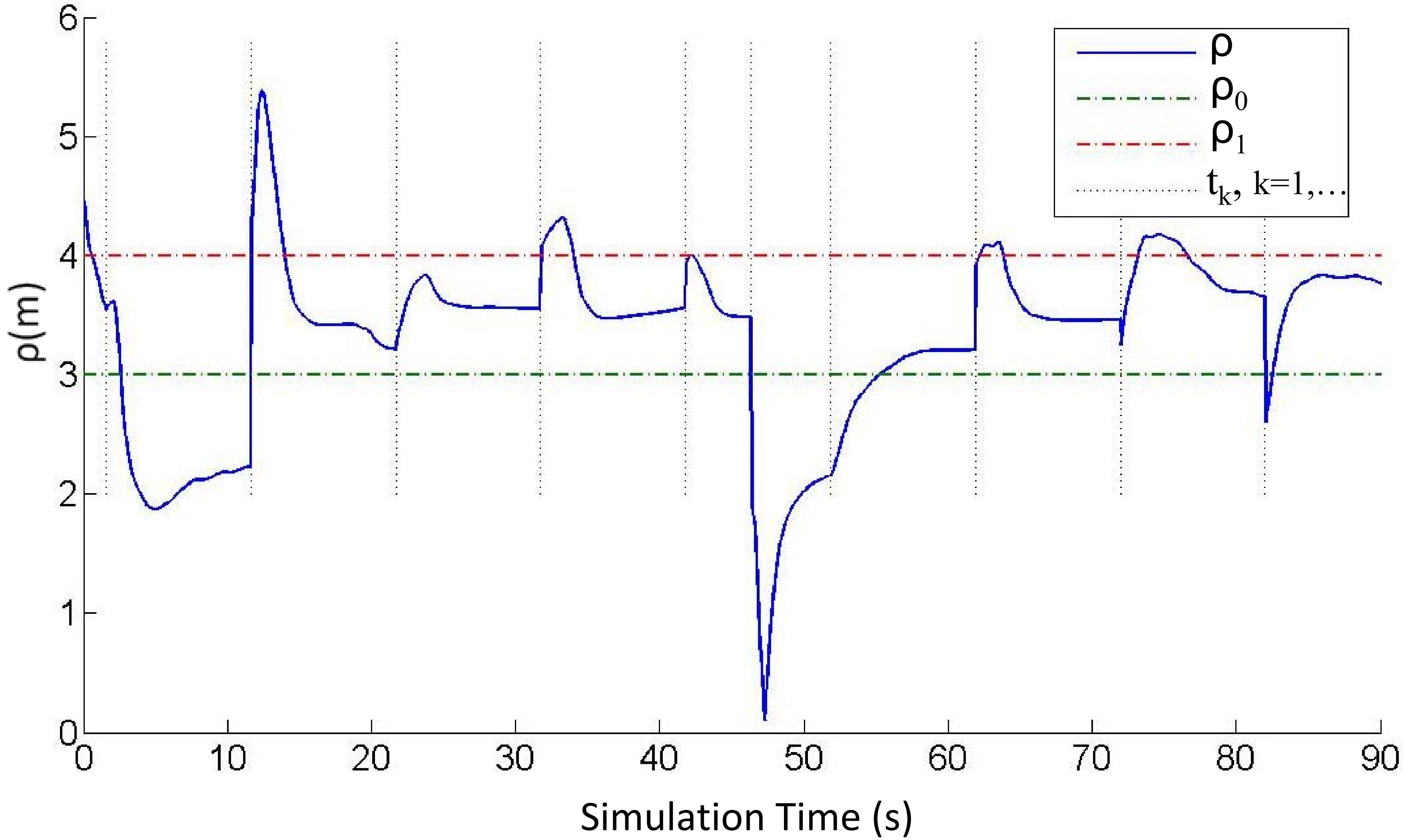
Simulation parameters:

Parameter	Value
Target speed (v)	2m/s
Workspace size	100m×100m
Heading changing frequency	0.1Hz
Measuring frequency	3.3Hz
Measuring noise	diag (0.4 0.4)
ρ_0	3m
ρ_1	4m

Results: Sensor Path Planning



Results: Sensor Surveillance



Summary and Conclusions

Results:

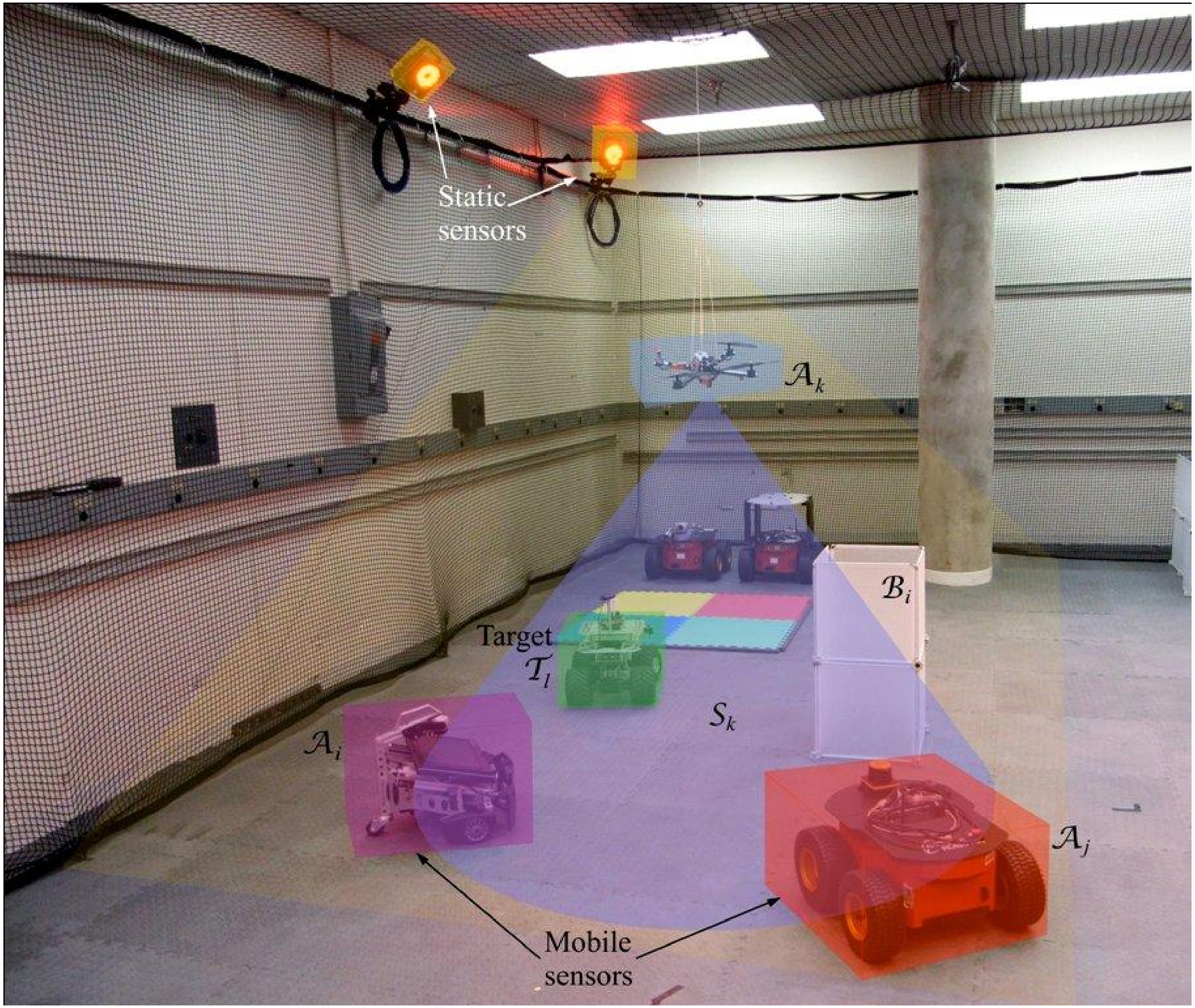
- Integration of geometric target and sensor modeling
- Particle filter method for target tracking
- Information potential field method:
  sensor path planning for tracking and surveillance

Future work:

- Estimate multiple Markov parameters (e.g. speed)
- Adaptive dynamic programming (ADP) approach
- Experimental testing

Acknowledgements: This work is supported by NSF grant ECCS #1028506.

Future Work: MARHES Experimental Testbed



Experimental Testing Concept