Cooperative Control of Mobile Sensors in Dynamic Environments

Shankarachary Ragi

Department of Electrical and Computer Engineering, Colorado State University

Spring 2013
Mobile Sensor

- Senses the environment
- Communicates with other mobile sensors or a base station
- Runs computations
- Changes its state (the state of a mobile sensor may include, e.g., its sensing direction, location, and waveform)

(a) UAV with on-board sensors
(b) PTZ camera
Cooperative Control

- Defense and civilian applications—surveillance, tracking, wild-fire suppression
- Focus—developing planning methods to
  - control multiple unmanned aerial vehicles (UAVs) for tracking multiple targets
  - control multiple autonomous amphibious vehicles (AAVs) for flood rescue support
  - control directional sensors for maximizing information gain
- Frameworks:
  - Centralized system—POMDP
  - Decentralized system—Dec-POMDP
Centralized UAV Guidance for Tracking Multiple Targets


Problem Description

- Targets move in 2-D
- UAVs fly at a constant altitude
- UAVs with onboard sensors send measurement frames to notional fusion node
- Fusion node:
  - Constructs tracks from measurements
  - Plans future motion of UAVs
  - Sends motion commands back to UAVs
- Objective: minimize mean-squared error of target estimates
Non-Myopic Dynamic Control

- The problem is inherently *dynamic*
  - Must exploit feedback
  - Poor control actions—regret at a future time
- Need non-myopic approach!
- POMDP: *Partially Observable Markov Decision Process*
state: \( x_k = (s_k, \chi_k, \xi_k, P_k) \)

- sensor state \( s_k \)
- target state \( \chi_k \)
- track state \( (\xi_k, P_k) \)
POMDP Formulation

state: \( x_k = (s_k, \chi_k, \xi_k, P_k) \)

measurement: 
\[
\begin{align*}
    z^\chi_k &= h(\chi_k, s_k) + w_k, \quad w_k \sim \mathcal{N}(0, R_k(\chi_k, s_k)) \\
    z^s_k &= s_k, \quad z^\xi_k = \xi_k, \quad z^P_k = P_k
\end{align*}
\]
POMDP Formulation

state: \( x_k = (s_k, \chi_k, \xi_k, P_k) \)

measurement: \( z_k^\chi = h(\chi_k, s_k) + w_k, \quad w_k \sim \mathcal{N}(0, R_k(\chi_k, s_k)) \)

action: \( u_k = (a_k, \phi_k) \)

- forward acceleration \( a_k \) and bank angle \( \phi_k \)
  
  \( a_k \in [a_{\text{min}}, a_{\text{max}}]^{N_{\text{sens}}} \quad \phi_k \in [\phi_{\text{min}}, \phi_{\text{max}}]^{N_{\text{sens}}} \)
POMDP Formulation

state: \( x_k = (s_k, \chi_k, \xi_k, P_k) \)

measurement: \( z_k^\chi = h(\chi_k, s_k) + w_k, \quad w_k \sim \mathcal{N}(0, R_k(\chi_k, s_k)) \)

action: \( u_k = (a_k, \phi_k) \)

state transition: \( x_{k+1} = T(x_k, u_k, v_k, w_{k+1}), \quad v_k \sim \mathcal{N}(0, Q_k) \)

\[
\begin{align*}
    s_{k+1} &= \psi(s_k, u_k) \\
    \chi_{k+1} &= f(\chi_k) + v_k \\
    (\xi_{k+1}, P_{k+1}) &= \text{Kalman filter update of } (\xi_k, P_k) \\
    &\quad \text{based on observed } z_{k+1}
\end{align*}
\]
POMDP Formulation

state: \( x_k = (s_k, \chi_k, \xi_k, P_k) \)

measurement: \( z^\chi_k = h(\chi_k, s_k) + w_k, \quad w_k \sim \mathcal{N}(0, R_k(\chi_k, s_k)) \)

action: \( u_k = (a_k, \phi_k) \)

state transition: \( x_{k+1} = T(x_k, u_k, v_k, w_{k+1}), \quad v_k \sim \mathcal{N}(0, Q_k) \)

belief state: \( b_k = (b^s_k, b^\chi_k, b^\xi_k, b^P_k) \)

\( b^s_k(s) = \delta(s - s_k) \)

\( b^\chi_k \) updated with \( z_k \) using Bayes theorem

\( b^\xi_k(\xi) = \delta(\xi - \xi_k) \)

\( b^P_k(P) = \delta(P - P_k) \)
POMDP Formulation

state: \( x_k = (s_k, \chi_k, \xi_k, P_k) \)

measurement: \( z^\chi_k = h(\chi_k, s_k) + w_k, \quad w_k \sim \mathcal{N}(0, R_k(\chi_k, s_k)) \)

action: \( u_k = (a_k, \phi_k) \)

state transition: \( x_{k+1} = T(x_k, u_k, v_k, w_{k+1}), \quad v_k \sim \mathcal{N}(0, Q_k) \)

belief state: \( b_k = (b^s_k, b^\chi_k, b^\xi_k, b^P_k) \)

cost: \( C(x_k, u_k) = \mathbb{E}_{v_k, w_{k+1}} \left[ \|\chi_{k+1} - \xi_{k+1}\|^2 \right| x_k, u_k \)
Objective is to minimize

\[ J_H = E \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

Objective can be posed over belief states

\[ J_H = E \left[ \sum_{k=0}^{H-1} c(b_k, u_k) \mid b_0 \right] \]

Solution is a belief state feedback policy: \( u_k = \pi_k(b_k) \)

Bellman’s principle:

\[
J_H^*(b_0) = \min_a \left\{ c(b_0, a) + E[J_{H-1}^*(b_1) \mid b_0, a] \right\} \\
= \min_a Q_H(b_0, a) \quad ("Q\text{-value}")
\]

\[ \pi_0^*(b_0) = \arg\min_a Q_H(b_0, a) \]

Hard to compute \( Q \)-value exactly!
POMDP Solution

- Objective is to minimize

\[ J_H = E \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

- Objective can be posed over belief states

\[ J_H = E \left[ \sum_{k=0}^{H-1} c(b_k, u_k) \mid b_0 \right] \]

- Solution is a belief state feedback policy: \( u_k = \pi_k(b_k) \)

Bellman's principle:

\[ J^*_H(b_0) = \min_a \left\{ c(b_0, a) + E[J^*_{H-1}(b_1) \mid b_0, a] \right\} \]

\[ = \min_a Q_H(b_0, a) \quad \text{("Q-value")} \]

\[ \pi^*_0(b_0) = \arg\min_a Q_H(b_0, a) \]

- Hard to compute Q-value exactly!
POMDP Solution

- Objective is to minimize

\[ J_H = E \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

- Objective can be posed over belief states

\[ J_H = E \left[ \sum_{k=0}^{H-1} c(b_k, u_k) \bigg| b_0 \right] \]

- Solution is a belief state feedback policy: \( u_k = \pi_k(b_k) \)

Bellman’s principle:

\[ J_H^*(b_0) = \min_a \left\{ c(b_0, a) + E[J_{H-1}^*(b_1) \big| b_0, a] \right\} \]

\[ = \min_a Q_H(b_0, a) \quad \text{("Q-value")} \]

\[ \pi_0^*(b_0) = \arg\min_a Q_H(b_0, a) \]

- Hard to compute Q-value exactly!
Objective is to minimize

\[ J_H = E \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

Objective can be posed over belief states

\[ J_H = E \left[ \sum_{k=0}^{H-1} c(b_k, u_k) \right] \]

Solution is a belief state feedback policy: \( u_k = \pi_k(b_k) \)

Bellman’s principle:

\[ J^*_H(b_0) = \min_a \left\{ c(b_0, a) + E[J^*_{H-1}(b_1) | b_0, a] \right\} \]

\[ = \min_a Q_H(b_0, a) \quad ("Q\text{-value"}) \]

\[ \pi^*_0(b_0) = \arg\min_a Q_H(b_0, a) \]

Hard to compute \( Q\)-value exactly!
Nominal Belief-State Optimization (NBO)

- Define nominal belief state sequence \((\hat{b}_1, \ldots, \hat{b}_{H-1})\)

\[
b_{k+1} = \Phi(b_k, u_k, v_k, w_{k+1})
\]

\[
\Rightarrow \hat{b}_{k+1} = \Phi(\hat{b}_k, u_k, 0, 0)
\]

\[
\hat{b}_0 = b_0
\]
Nominal Belief-State Optimization (NBO)

1. Define nominal belief state sequence \( (\hat{b}_1, \ldots, \hat{b}_{H-1}) \)

2. Replace expectation over random future belief states

\[
J_H(b_0) = \mathbb{E}_{b_1, \ldots, b_H} \left[ \sum_k c(b_k, u_k) \left| b_0 \right. \right]
\]

with the sample given by nominal belief state sequence

\[
J_H(b_0) \approx \sum_k c(\hat{b}_k, u_k)
\]
Nominal Belief-State Optimization (NBO)

1. Define nominal belief state sequence \((\hat{b}_1, \ldots, \hat{b}_{H-1})\)

2. Replace expectation over random future belief states

\[
J_H(b_0) = \text{E}_{b_1, \ldots, b_H} \left[ \sum_k c(b_k, u_k) \bigg| b_0 \right]
\]

with the sample given by nominal belief state sequence

\[
J_H(b_0) \approx \sum_k c(\hat{b}_k, u_k)
\]

3. Objective function from NBO:

\[
J_{\text{NBO}} = \sum_{k=0}^{H-1} \sum_{i=1}^{N_{\text{targets}}} \text{Tr} \; \hat{P}_{k+1}^i
\]
Nominal Belief-State Optimization (NBO)

1. Define nominal belief state sequence \((\hat{b}_1, \ldots, \hat{b}_{H-1})\)
2. Replace expectation over random future belief states

\[
J_H(b_0) = \mathbb{E}_{b_1, \ldots, b_H} \left[ \sum_k c(b_k, u_k) \bigg| b_0 \right]
\]

with the sample given by nominal belief state sequence

\[
J_H(b_0) \approx \sum_k c(\hat{b}_k, u_k)
\]

3. Objective function from NBO:

\[
J_{\text{NBO}} = \sum_{k=0}^{H-1} \sum_{i=1}^{N_{\text{targets}}} \text{Tr} \hat{P}_{k+1}^i
\]
Three UAVs Tracking Two Targets
Stationary Target: Performance Bounds

Proposition

\[
\sum_{i=1}^{N_{\text{targs}}} \sum_{k=0}^{H-1} \frac{M^2}{\text{Tr} \left[ (P^i_0)^{-1} \right]} + (k + 1)C \leq J^*_H(b_0) \leq H \sum_{i=1}^{N_{\text{targs}}} \text{Tr}P^i_0,
\]

where \( M \) and \( C \) are constants
Performance comparison: 1 UAV and 1 Stationary Target
Enhancements

- Wind compensation
- Collision avoidance
- Track swap avoidance
- Tracking evasive targets
- Threat evasion
Decentralized UAV Guidance for Tracking Multiple Targets
Problem Specification

Targets
Target Specification

**Targets**

**Unmanned aerial vehicles** (also called agents)

- the locations and velocities of every UAV in the system are available at each UAV
Problem Specification

Targets

Unmanned aerial vehicles (also called agents)

Spatially varying measurement errors
Problem Specification

Targets

**Unmanned aerial vehicles** (also called agents)

**Spatially varying measurement errors**

Communication

- each UAV can transmit and receive information to/from other UAVs
Problem Specification

Targets

Unmanned aerial vehicles (also called agents)

Spatially varying measurement errors

Communication

Tracker
Problem Specification

Targets
Unmanned aerial vehicles (also called agents)
Spatially varying measurement errors
Communication
Tracker

Objective
- optimize UAV controls, in a decentralized setting, such that a composite cost metric, a weighted sum of the mean-squared error and the cost of communication, is minimized
Communication Between the Agents

- Agents communicate over an underlying network
- *Communication link*—the end-to-end path through the network between a pair of agents
- *Link usage cost*—proportional to the distance between the two associated agents
- Further assumptions
  - an agent can communicate with at most one other agent at any decision epoch
  - an agent can send one of the $L$ locally generated target observations in the past $L$ time-steps to a chosen destination (agent)
Key Components of Dec-POMDP

agents: set of agents $\mathcal{I} = \{1, \ldots, N\}$
Key Components of Dec-POMDP

agents: set of agents $\mathcal{I} = \{1, \ldots, N\}$

state: $x_k = (s_k, \chi_k, T_k)$
  - UAV state $s_k$
  - target state $\chi_k$
  - track state $T_k = (T_k^1, \ldots, T_k^N)$, where $T_k^i = (\xi_k^i, P_k^i)$ is the state of the tracking algorithm at agent $i$
Key Components of Dec-POMDP

- **agents**: set of agents $\mathcal{I} = \{1, \ldots, N\}$
- **state**: $x_k = (s_k, \chi_k, T_k)$
- **measurement**: $z_k = (z_1^k, \ldots, z_N^k)$, where $z_i^k$ is the local observation at agent $i$
Key Components of Dec-POMDP

agents: set of agents $\mathcal{I} = \{1, \ldots, N\}$

state: $x_k = (s_k, \chi_k, T_k)$

measurement: $z_k = (z_k^1, \ldots, z_k^N)$, where $z_k^i$ is the local observation at agent $i$

action: $u_k = (u_k^1, \ldots, u_k^N)$, where $u_k^i$ is the action-vector at agent $i$

$u_k^i = (a_k^i, g_k^i, l_k^i)$, where $a_k^i$ is the kinematic control command, $g_k^i$ is the ID of agent ($\neq i$) to whom $i$ will send its locally generated target observation at time $l_k^i \in \{k, \ldots, k - L\}$ in the past
Key Components of Dec-POMDP

agents: set of agents \( \mathcal{I} = \{1, \ldots, N\} \)

state: \( x_k = (s_k, \chi_k, T_k) \)

measurement: \( z_k = (z^1_k, \ldots, z^N_k) \), where \( z^i_k \) is the local observation at agent \( i \)

action: \( u_k = (u^1_k, \ldots, u^N_k) \), where \( u^i_k \) is the action-vector at agent \( i \)

state transition: \( x_{k+1} \sim p_k(\cdot | x_k, u_k) \)

\[
\begin{align*}
s_{k+1} &= \psi(s_k, u_k) \\
\chi_{k+1} &= F\chi_k + e_k
\end{align*}
\]

The track states at each agent \( i \) are updated according to the Kalman filter equations given the local observations and the information received from other agents.
Key Components of Dec-POMDP

agents: set of agents $\mathcal{I} = \{1, \ldots, N\}$

state: $x_k = (s_k, \chi_k, T_k)$

measurement: $z_k = (z^1_k, \ldots, z^N_k)$, where $z^i_k$ is the local observation at agent $i$

action: $u_k = (u^1_k, \ldots, u^N_k)$, where $u^i_k$ is the action-vector at agent $i$

state transition: $x_{k+1} \sim p_k(\cdot | x_k, u_k)$

cost: $C(x_k, u_k) = \text{weighted mean of the mean-squared tracking error and the cost of communication}$
Objective and Optimal Policy

- Minimize

\[
J_H = \mathbb{E}\left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right]
\]

- The optimal policy for the centralized system is

\[
\pi^*(b_0) = \arg\min_u Q(b_0, u),
\]

where

\[
Q(b_0, u) = c(b_0, u) + \mathbb{E}[J_H^*(b_1) \mid b_0, u],
\]

- For the decentralized case, our approach is

\[
\pi^i(b^i_k) = \arg\min_u Q(b^i_k, u), \quad i = 1, \ldots, N
\]

where \(b^i_k\) is the local belief-state at \(i\)

- The objective function from NBO at agent \(i\) is

\[
J_{NBO}(b^i_0) = \sum_{k=0}^{H-1} \left( \text{Tr} \hat{P}^i_{k+1} + \beta \hat{d}^i_k g^i_k \right)
\]
Objective and Optimal Policy

- Minimize

\[ J_H = E \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

- The optimal policy for the centralized system is

\[ \pi^*(b_0) = \arg \min_u Q(b_0, u), \]

where

\[ Q(b_0, u) = c(b_0, u) + E[J^*_H(b_1) \mid b_0, u] \]

- For the decentralized case, our approach is

\[ \pi^i(b^i_k) = \arg \min_u Q(b^i_k, u), \quad i = 1, \ldots, N \]

where \( b^i_k \) is the local belief-state at \( i \)

- The objective function from NBO at agent \( i \) is

\[ J_{NBO}(b^i_0) = \sum_{k=0}^{H-1} \left( \text{Tr} \hat{P}^i_{k+1} + \beta \hat{d}^i_{k}, g^i_k \right) \]
Objective and Optimal Policy

- Minimize

\[ J_H = \mathbb{E} \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

- The optimal policy for the centralized system is

\[ \pi^*(b_0) = \arg \min_u Q(b_0, u), \]

where

\[ Q(b_0, u) = c(b_0, u) + \mathbb{E}[J^*_H(b_1) | b_0, u], \]

- For the decentralized case, our approach is

\[ \pi^i(b^i_k) = \arg \min_u Q(b^i_k, u), \quad i = 1, \ldots, N \]

where \( b^i_k \) is the local belief-state at \( i \)

- The objective function from NBO at agent \( i \) is

\[ J_{NBO}(b^i_0) = \sum_{k=0}^{H-1} \left( \text{Tr} \hat{P}^i_{k+1} + \beta \hat{d}^i_{k+1}, g^i_k \right) \]
Objective and Optimal Policy

- Minimize

\[ J_H = \mathbb{E} \left[ \sum_{k=0}^{H-1} C(x_k, u_k) \right] \]

- The optimal policy for the centralized system is

\[ \pi^*(b_0) = \arg \min_u Q(b_0, u), \]

where

\[ Q(b_0, u) = c(b_0, u) + \mathbb{E}[J^*_H(b_1) \mid b_0, u], \]

- For the decentralized case, our approach is

\[ \pi^i(b^i_k) = \arg \min_u Q(b^i_k, u), \quad i = 1, \ldots, N \]

where \( b^i_k \) is the local belief-state at \( i \)

- The objective function from NBO at agent \( i \) is

\[ J_{NBO}(b^i_0) = \sum_{k=0}^{H-1} \left( \text{Tr} \hat{\mathbf{P}}^i_{k+1} + \beta \hat{d}^i_k \hat{g}^i_k \right) \]
Two UAVs Tracking Two Targets; $\beta = 1$
Two UAVs Tracking Two Targets; $\beta = 100$
Average Target Location Error

Empirical CDF

- $\beta = 1$
- $\beta = 50$
- $\beta = 100$

Cumulative frequency vs. aggregate target-location error
Average Communication Cost

Empirical CDF

- $\beta = 1$
- $\beta = 50$
- $\beta = 100$

Cumulative frequency

aggregate communication cost

$\beta$ — a tuning parameter
Autonomous Amphibious Vehicle (AAV) Guidance for Flood Rescue Support
Publications


Flood Scenario

AAV Victim/Target River

\[ d_{ref} \]

Victim/Target
POMDP Formulation

state: \( x_k = (s_k, d^\text{ref}_k, \chi_k, \xi_{\text{riv}}^k, P_{\text{riv}}^k, \xi_{\text{targ}}^k, P_{\text{targ}}^k) \)

- AAV state \( s_k \)
- river state \( d^\text{ref}_k \)
- target state \( \chi_k \)
- track states \( (\xi_{\text{riv}}^k, P_{\text{riv}}^k, \xi_{\text{targ}}^k, P_{\text{targ}}^k) \)
POMDP Formulation

state: \( x_k = (s_k, d^\text{ref}_k, \chi_k, \xi_{riv}^k, P_{riv}^k, \xi_{targ}^k, P_{targ}^k) \)

measurement: sensor and the track states are fully observable; noisy observations of river and target states are available
state: \( x_k = (s_k, d_{k}^{\text{ref}}, \chi_k, \xi_{\text{riv}}^k, P_{\text{riv}}^k, \xi_{\text{targ}}^k, P_{\text{targ}}^k) \)

measurement: sensor and the track states are fully observable; noisy observations of river and target states are available

action: \( u_k = (g_k, a_k) \)

- \( g_k \) represents a one-to-one correspondence between AAVs and targets
- \( a_k \) includes forward accelerations and steering angles for each AAV
POMDP Formulation

**state:** \( x_k = (s_k, d_{k+1}^{\text{ref}}, \chi_k, \xi_k, P_k^{\text{riv}}, \xi_k^{\text{targ}}, P_k^{\text{targ}}) \)

**measurement:** sensor and the track states are fully observable; noisy observations of river and target states are available

**action:** \( u_k = (g_k, a_k) \)

**state transition:**
\[
x_{k+1} = T(x_k, u_k, e_k^{\text{targ}}, e_k^{\text{riv}}, n_{k+1}^{\text{targ}}, n_{k+1}^{\text{riv}}),
\]
\[
e_k^{\text{targ}} \sim \mathcal{N}(0, Q_k^{\text{targ}}), e_k^{\text{riv}} \sim \mathcal{N}(0, Q_k^{\text{riv}})
\]
\[
\chi_{k+1} = F \chi_k + e_k^{\text{targ}}
\]
\[
d_{k+1}^{\text{ref}} = d_k^{\text{ref}} + e_k^{\text{riv}}
\]

AAV state evolves according to the AAV kinematic motion model. Track states are updated according to Kalman filter equations.
POMDP Formulation

state: \( x_k = (s_k, d_{k}^{ref}, \chi_k, \chi_{riv}^k, P_{riv}^k, \chi_{targ}^k, P_{targ}^k) \)

measurement: sensor and the track states are fully observable; noisy observations of river and target states are available

action: \( u_k = (g_k, a_k) \)

state transition: \( x_{k+1} = T(x_k, u_k, e_{k}^{targ}, e_{k}^{riv}, n_{k+1}^{targ}, n_{k+1}^{riv}) \),

\[ e_{k}^{targ} \sim \mathcal{N}(0, Q_{k}^{targ}), \ e_{k}^{riv} \sim \mathcal{N}(0, Q_{k}^{riv}) \]

cost: \( C(x_k, u_k) = \sum_{i=1}^{N} 1 \left\{ \mathbb{E} \left[ \| s_{k+1}^{i,pos} - \chi_{k+1}^{g(i),targ,pos} \| \left| x_k, u_k \right\| > d_{\text{dist-thresh}} \right] \right\} \),
The objective function in terms of the belief states is

\[ J_H(b_0) = \mathbb{E}\left[ \sum_{k=0}^{H-1} c(b_k, u_k) \mid b_0 \right] , \]

NBO approach with distance-based ECTG

\[ J_{\text{NBO}}(b_0) = \sum_{k=0}^{H-1} \sum_{i=1}^{N} 1 \left\{ \left\| \hat{s}_{i,\text{pos}}^{k+1} - \hat{\xi}_{g(i),\text{targ},\text{pos}}^{k+1} \right\| > d_{\text{dist-thresh}} \right\} + J_{H}^{\text{dist-ECTG}}, \]

where

\[ J_{H}^{\text{dist-ECTG}} = \sum_{i=1}^{N} \left\| \hat{s}_{H}^{i,\text{pos}} - \hat{\xi}_{g(i),\text{targ},\text{pos}}^{H} \right\| , \]
Objective Function and Solution

-The objective function in terms of the belief states is

\[ J_H(b_0) = E\left[ \sum_{k=0}^{H-1} c(b_k, u_k) \bigg| b_0 \right] , \]

-NBO approach with distance-based ECTG

\[ J_{NBO}(b_0) = \sum_{k=0}^{H-1} \sum_{i=1}^{N} 1 \left\{ \| \hat{s}_{i, \text{pos}}^{k+1} - \hat{\xi}_{g(i)\text{,targ, pos}}^{k+1} \| > d_{\text{dist-thresh}} \right\} + J_{H}^{\text{dist-ECTG}} , \]

where

\[ J_{H}^{\text{dist-ECTG}} = \sum_{i=1}^{N} \left\| \hat{s}_{H}^{i, \text{pos}} - \hat{\xi}_{H}^{g(i)\text{,targ, pos}} \right\| , \]
2 AAVs and 2 Targets (or Flood Victims)
Proposed Work
Decentralized UAV Guidance (Extensions)
Extensions

- Dynamic (time-varying) network between the UAVs
- Explicit optimization of communication decisions among the UAVs at network level
- Incorporate communication delays
Directional Sensor Control
Directional Sensor

Limited field-of-view!
Problem Specification

Targets

- static or moving
Problem Specification

Targets

Directional Sensors

- set of possible directions for each sensor $\Theta = \{1, \ldots, K\}$
- control vector $u = (u_1, \ldots, u_M)$, $u_i \in \Theta$, $i = 1, \ldots, M$
Problem Specification

Targets

Directional Sensors

Measurement Errors
- measurement of a target is available only if the target lies within the field-of-view of a sensor
- spatially varying random (Gaussian) measurement errors
Problem Specification

Targets
Directional Sensors
Measurement Errors
Fusion Center
Problem Specification

Targets
Directional Sensors
Measurement Errors
Fusion Center
Objective

find $u$ such that the information gain is maximized

$$E \left[ \sum_{j=1}^{N} - \log \left( \det (P_j) \right) \right] ,$$

a combinatorial optimization problem!
Example Scenario
Proposed Work

- Develop heuristic approaches for static targets
- Extend to dynamic target scenario—POMDP
- Extend to decentralized setting
  - Dec-POMDP formulation
  - Optimize the communication decisions at network level (for a given network)
- Extend to video/camera network applications
Proposed Work

- Develop heuristic approaches for static targets
- Extend to dynamic target scenario—POMDP
- Extend to decentralized setting
  - Dec-POMDP formulation
  - Optimize the communication decisions at network level (for a given network)
- Extend to video/camera network applications
Proposed Work

- Develop heuristic approaches for static targets
- Extend to dynamic target scenario—POMDP
- Extend to decentralized setting
  - Dec-POMDP formulation
  - Optimize the communication decisions at network level (for a given network)
- Extend to video/camera network applications
Proposed Work

- Develop heuristic approaches for static targets
- Extend to dynamic target scenario—POMDP
- Extend to decentralized setting
  - Dec-POMDP formulation
    - Optimize the communication decisions at network level (for a given network)
- Extend to video/camera network applications
Proposed Work

- Develop heuristic approaches for static targets
- Extend to dynamic target scenario—POMDP
- Extend to decentralized setting
  - Dec-POMDP formulation
  - Optimize the communication decisions at network level (for a given network)
- Extend to video/camera network applications
Proposed Work

- Develop heuristic approaches for static targets
- Extend to dynamic target scenario—POMDP
- Extend to decentralized setting
  - Dec-POMDP formulation
  - Optimize the communication decisions at network level (for a given network)
- Extend to video/camera network applications
Questions?