LISC

LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Sensor Planning for Multiple Targets Tracking

Hongchuan Wei

Advisor: Silvia Ferrari
Laboratory for Intelligent Systems and Controls

Cornell Robotics In Society Seminar
December 14, 2015

Outline

- Introduction and Motivation
- Problem Formulation
- Methodology: Information Based Sensor Planning
  - Target behavior described by single model
  - Target behavior described by mixture model mobile
  - Sensor dynamics constraints
- Simulation and Results
- Conclusion

LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Introduction and Motivation

Target behaviors learning

- Security surveillance
- Tracking endangered species
- Environmental monitoring

⋮

[1]

[2]

[3]

[1] http://dowley.com/Services/VideoSurveillance/tabid/91/Default.aspx
[2] An Information Value Function For Nonparametric Gaussian Processes
[3] www.h3c.com

Introduction and Motivation

- Goal: control actuated or configurable sensors to actively collect most valuable information
  - Estimator: compute the system state by fusing the data
  - Planner: determine the control by optimizing a function of costs and utilities
  - Actuator: follow the task execution as closely as possible
- Challenge:
  - Nonlinear target dynamics
  - Adaptive to data
  - Define 'value' of information
  - Number of sensors < number of targets

Mathematical Problem Formulation

Target behaviors learning

- Workspace: $\mathcal{W} \subset \mathbb{R}^2$, convex polygon
- Target dynamics: unknown form

$$\dot{\mathbf{x}}_j(t) = \mathbf{f}[\mathbf{x}_j(t)], \;\; j = 1, \ldots, N$$

- Sensor dynamics: known

$$\dot{\mathbf{s}}(t) = \mathbf{g}[\mathbf{s}(t), \mathbf{u}(t)], \;\; \mathbf{u}(t) \in U$$

- Detection model: limited sensor field of view (FOV)

$$P_d = \begin{cases} 0 : \mathbf{x}_j(t) \notin S(t) \\ 1 : \mathbf{x}_j(t) \in S(t) \end{cases}$$

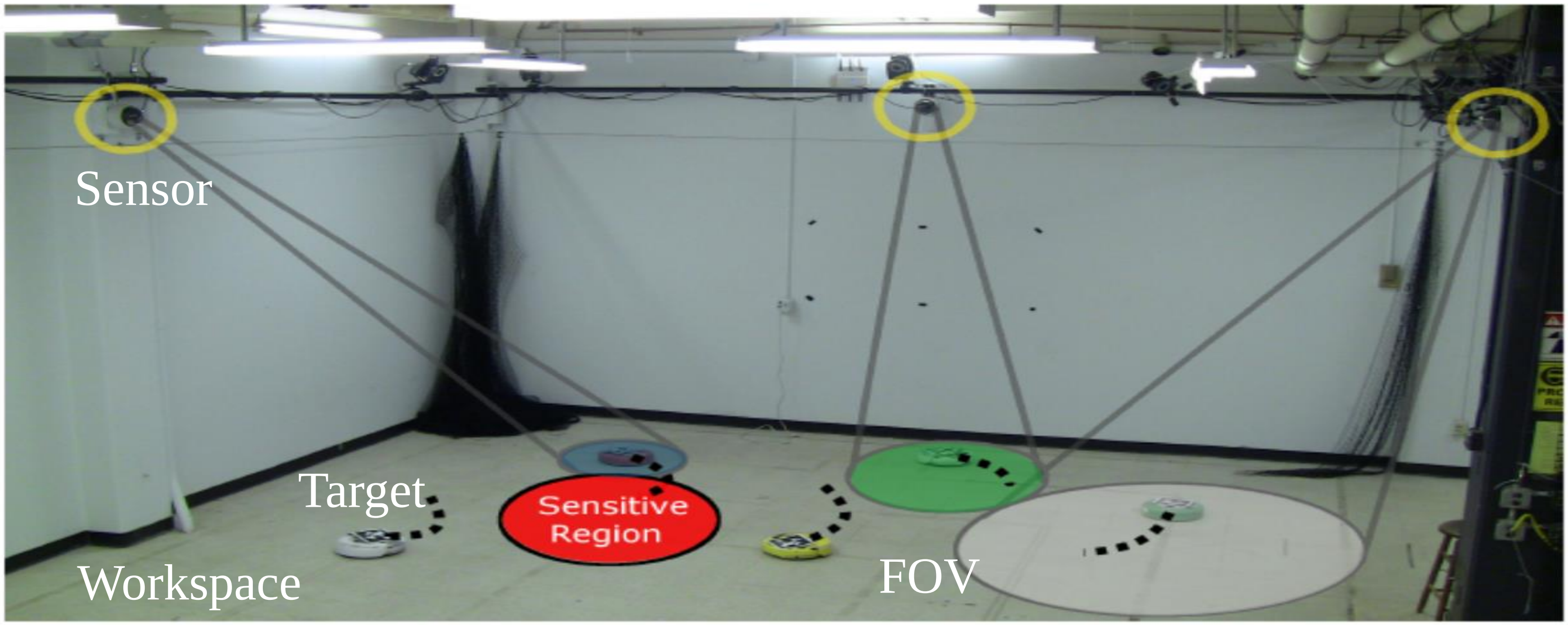
- Measurement model: known with additive noise

$$\mathbf{m}(t) = \mathbf{h}[\mathbf{x}(t), \mathbf{s}(t)] + \mathbf{n}, \;\; \mathbf{n} \sim \mathcal{N}(0, \sigma^2)$$

- Goal: determine optimal control $\mathbf{u}^*(t)$ of sensors, such that collected measurements are most useful for learning $\mathbf{f}$

LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Example Problem



- Workspace: MIT raven testbed
- Target dynamics: unknown velocity field

$$\mathbf{f}\colon [x\ y]^T \rightarrow [\dot{x}\ \ \dot{y}]^T$$

- Sensor dynamics: linear with constraints
- Detection model: camera FOV
- Measurement model: $\mathbf{m}_j(t) = [\mathbf{x}_j(t)\ \ \mathbf{v}_j(t)] + \mathbf{n}$

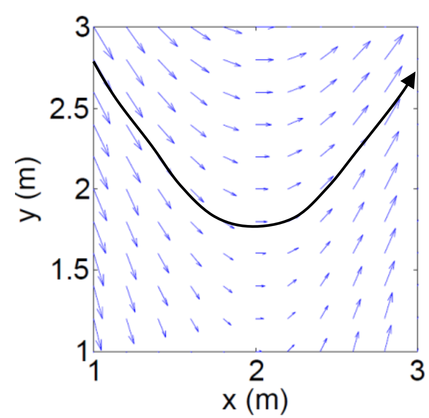


Fig. example of $\mathbf{f}[\mathbf{x}_j(t)]$

Methodology Part I:
Target Dynamics Modelling

Methodology: Target Modeling

Single target: Gaussian process (GP) regression

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t)] \longrightarrow \{\mathbf{x}(t), \dot{\mathbf{x}}(t)\} \Rightarrow \mathbf{f}(\cdot)$$

A Gaussian process is a stochastic process, that is a collection of random variables $\{ f(\mathbf{x}) \mid \mathbf{x} \in \mathcal{X} \}$ indexed by a set $\mathcal{X}$, for which any finite subset, $\{ f(\mathbf{x}_1), ..., f(\mathbf{x}_k)\}$, has a joint multivariate Gaussian distribution.

- Notation: $f(\mathbf{x}) \sim \mathrm{GP}[\theta(\mathbf{x}), \phi(\mathbf{x}, \mathbf{x}')]$
- Mean function: $\theta(\mathbf{x}) = \mathbb{E}_f[f(\mathbf{x})]$
- Covariance function: $\phi(\mathbf{x}, \mathbf{x}') = \mathbb{E}_f\{[f(\mathbf{x}) - \theta(\mathbf{x})][f(\mathbf{x}') - \theta(\mathbf{x}')]\}^T$
- Covariance matrix: $\Phi(\mathbf{a}, \mathbf{b}) = \begin{bmatrix} \phi(\mathbf{a}_1, \mathbf{b}_1) & \cdots & \phi(\mathbf{a}_1, \mathbf{b}_n) \\ \vdots & \ddots & \vdots \\ \phi(\mathbf{a}_m, \mathbf{b}_1) & \cdots & \phi(\mathbf{a}_m, \mathbf{b}_n) \end{bmatrix}$

Gaussian Process Regression

Prediction on $\mathbf{f}(\xi) = [f(\xi_1) \;\; \cdots \;\; f(\xi_k)]^T$ given data $\{\mathbf{p}, \mathbf{o}\}$:

$$\mathbf{f}(\xi) \,|\{\mathbf{p}, \mathbf{o}\} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- mean: $\boldsymbol{\mu} = \boldsymbol{\theta}(\xi) + \boldsymbol{\Phi}(\xi, \mathbf{p})\boldsymbol{\Phi}(\mathbf{p}, \mathbf{p})^{-1}[\mathbf{o} - \boldsymbol{\theta}(\mathbf{p})]$
- covariance: $\boldsymbol{\Sigma} = \boldsymbol{\Phi}(\xi, \xi) - \boldsymbol{\Phi}(\xi, \mathbf{p})\boldsymbol{\Phi}(\mathbf{p}, \mathbf{p})^{-1}\boldsymbol{\Phi}(\mathbf{p}, \xi)$

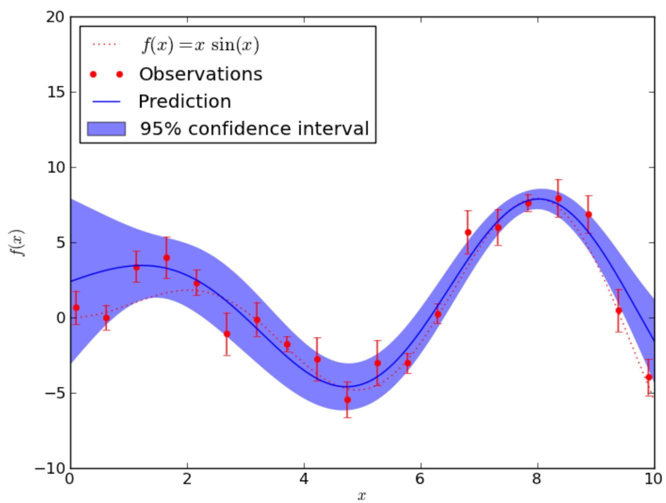


Fig. 1D GP regression example

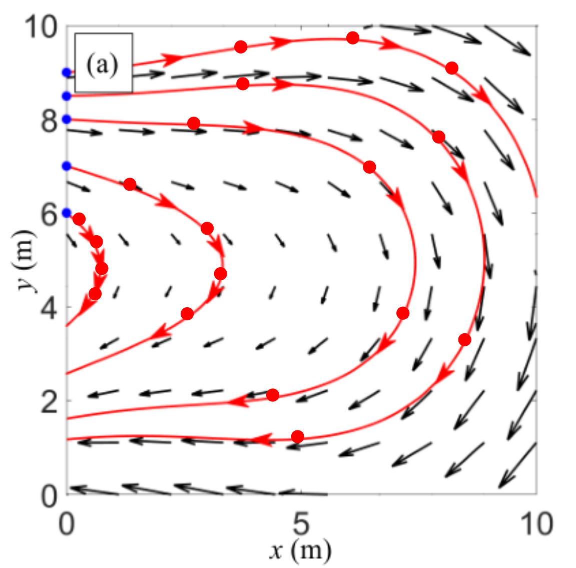


Fig. 2D GP regression example

http://scikit-learn.org/0.11/_images/plot_gp_regression_1.png

Dirichlet Distribution

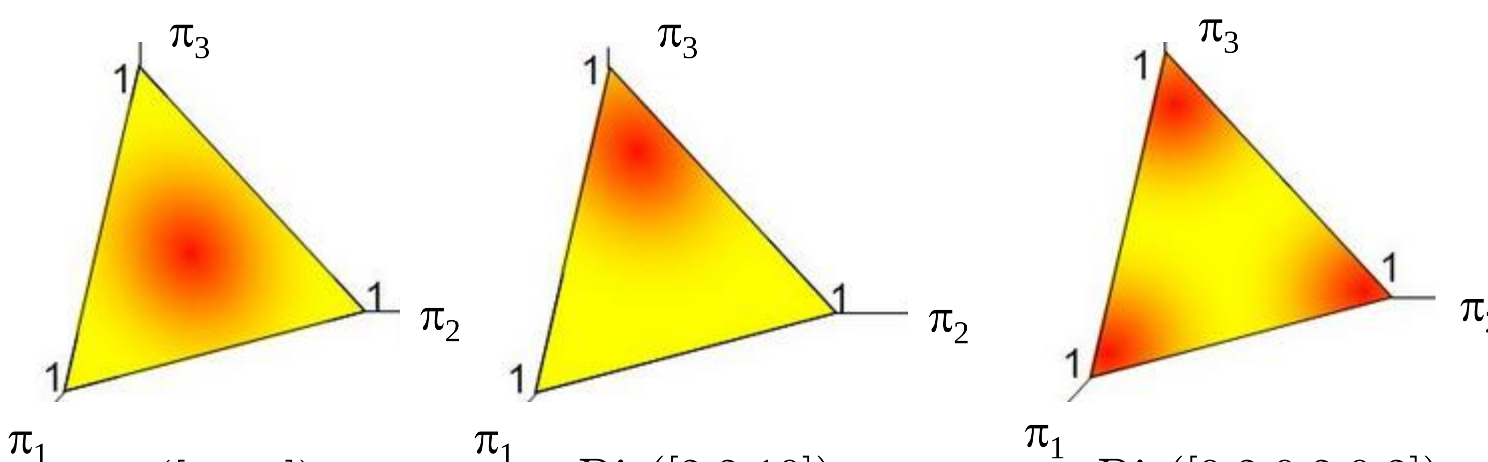
Dirichlet distribution: distribution over k-dimensional probability simplex

$$\boldsymbol{\pi} = [\pi_1 \;\; \cdots \;\; \pi_k]^T, \quad \text{for } \pi_k \geq 0, \;\; \Sigma_k \pi_k = 1$$

- Density function of $\boldsymbol{\pi} \sim \mathrm{Dir}([\alpha_1 \;\; \ldots \;\; \alpha_k])$

$$p(\pi) = \frac{\Gamma(\Sigma_k \alpha_k)}{\Pi_k \Gamma(\alpha_k)} \Pi_k \pi_k^{\alpha_k - 1}$$

- Example



Dir([2,2,2]) Dir([2,2,10]) Dir([0.2,0.2,0.2])

http://slideplayer.us/slide/783946/

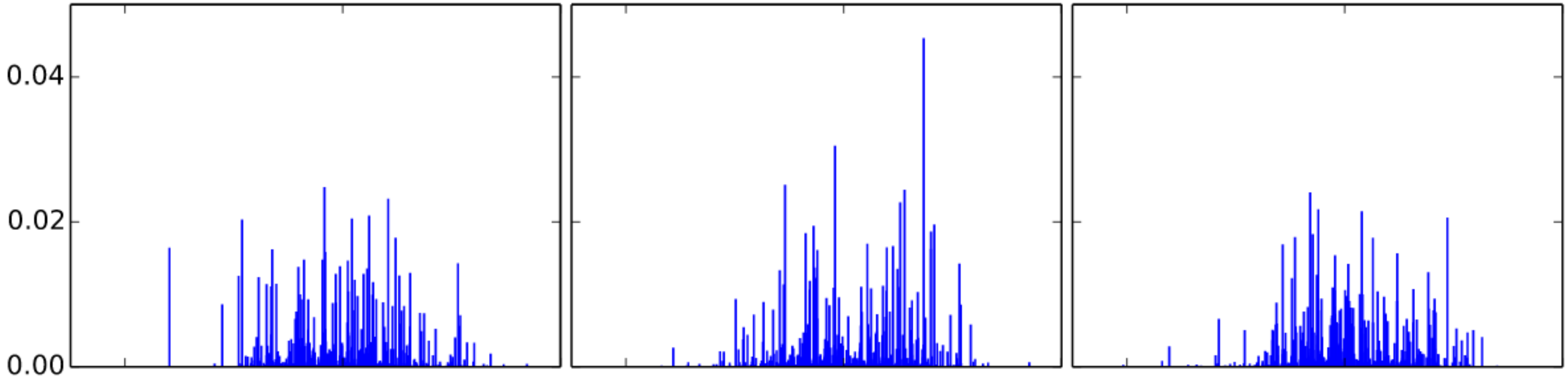
LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Dirichlet Process

Dirichlet process: distribution of infinite-dimension $\boldsymbol{\pi}$

A Dirichlet process with parameters H and α, denoted by DP$[\alpha, H(A)]$, is a distribution of a random probability measure P, if for any finite measureable partition $\{B_i \mid 1 \le i \le n\}$ of A, it is true that

$$[P(B_1) \quad \cdots \quad P(B_n)]^T \sim \mathrm{Dir}([\alpha H(B_1) \quad \cdots \quad \alpha H(B_n)])$$



- Three samples drawn from DP$(100, \mathcal{N}(0,0.1))$
- discrete distribution
- countably infinite number of point masses

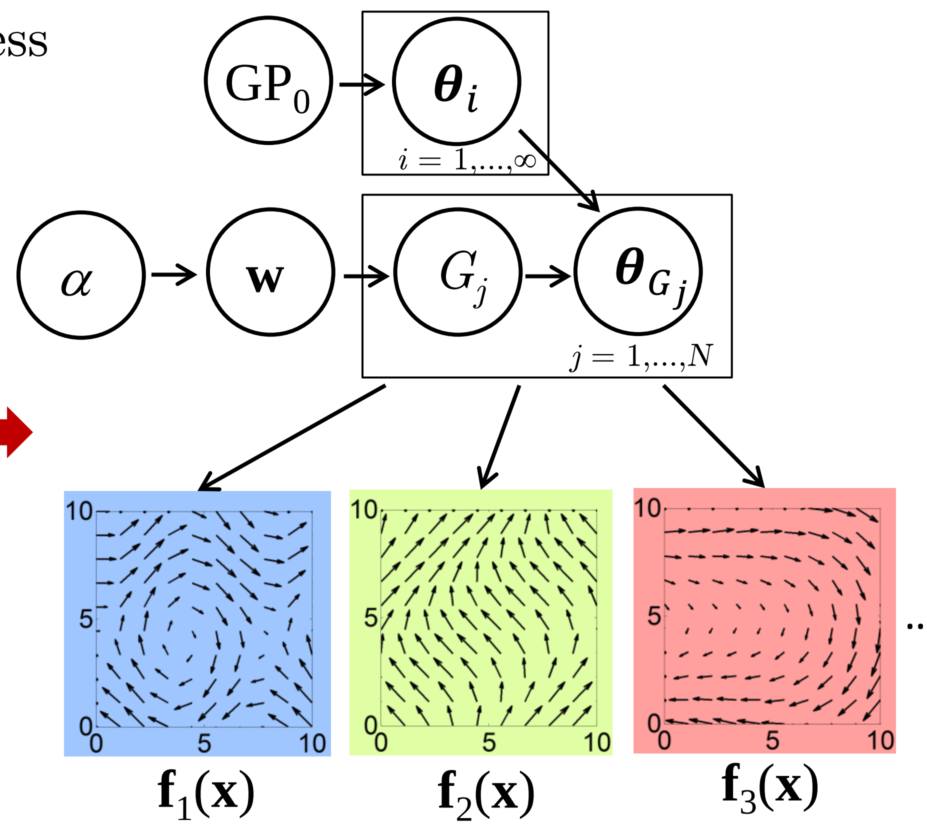
https://upload.wikimedia.org/wikipedia/commons/d/d3/Dirichlet˙process˙draws.svg

DPGP Mixture Model

Multiple targets: Dirichlet process prior over mixture of Gaussian process models (DPGP mixture model)

- Objective of DPGP-MM: describe target dynamics, $\{\mathcal{F}, \boldsymbol{\pi}\}$
  - Velocity field: 2D spatial phenomenon $\leftarrow$ Gaussian process
  - Clustering $\leftarrow$ Dirichlet process
- DPGP mixture model[1]:

$$\{\boldsymbol{\theta}_i, \mathbf{w}\} \sim \mathrm{DP}(\alpha, \mathrm{GP}_0),\ i = 1,\ldots,\infty$$
$$G_j \sim \mathrm{Cat}(\mathbf{w}),\ j = 1,\ldots,N$$
$$\mathbf{f}_{Gj}(\mathbf{x}) \sim \mathrm{GP}(\boldsymbol{\theta}_{Gj}, \boldsymbol{\Phi}),\ j = 1,\ldots,N$$



Methodology Part II: Information Value of Measurement

DPGP Information Value

DPGP info-value: difference between posterior and prior DPGP model

$$\varphi[\boldsymbol{\nu};\, \mathbf{m}(k+1)] = D\{p[\boldsymbol{\nu} | Q(k+1)] \| p[\boldsymbol{\nu} | Q(k)]\}$$

- D: Kullback-Leibler (or other) divergence

$$D(P_1 \parallel P_2) = \int_{-\infty}^{\infty} \ln\left(\frac{p_1(x)}{p_2(x)}\right) p_1(x)\, dx$$

- $\boldsymbol{\nu}_i = [\mathbf{f}_i(\boldsymbol{\xi}_1)^T \;\; \cdots \;\; \mathbf{f}_i(\boldsymbol{\xi}_L)^T]^T$
- $\boldsymbol{\nu} = [\boldsymbol{\nu}_1^T \;\; \cdots \;\; \boldsymbol{\nu}_M^T]^T$
- $Q(\mathrm{k}) = \{\mathbf{m}(\ell),\ G(\ell) \mid 1 \leq \ell \leq k\}$
- velocity field-target association, G, is unknown
- future measurement, $\mathbf{m}(k+1)$, is unknown

DPGP-EKLD

Expected info-value: DPGP-EKLD

$$\hat{\varphi}[\boldsymbol{\nu};\, \mathbf{m}(k{+}1)] = \mathbb{E}_{G_j}\left[\mathbb{E}_{\mathbf{m}(k+1)}\left[\varphi[\boldsymbol{\nu};\, \mathbf{m}(k{+}1)]\right]\right]$$

- Assumptions:
  - VF-target association distribution learnt by DPGP model
  - measurement consistent with GP regression
- Theorem I: The GP-EKLD can be simplified as

$$\hat{\varphi}\left[\boldsymbol{v};\mathbf{m}(k+1)\right] = \sum_{i,j} w_{ij} \int_{\mathcal{S}} h_i[\mathbf{x}_j(k+1)] \times p\left(\mathbf{x}_j(k+1)|Q(k)\right) d\mathbf{x}_j$$

where

$$h_i[\mathbf{x}_j(k+1)] = \frac{1}{2}\left[\mathrm{tr}\left(\boldsymbol{\Sigma}_{i,k}^{-1}\boldsymbol{\Sigma}_{i,k+1}\right) - \ln\left(|\boldsymbol{\Sigma}_{i,k+1}\boldsymbol{\Sigma}_{i,k}^{-1}|\right) - 2L + \mathrm{tr}(\mathbf{Q}^{-1}\mathbf{R}^T\boldsymbol{\Sigma}_{i,k}^{-1}\mathbf{R}\mathbf{Q}^{-1})\sigma_v^2\right]$$

$$\mathbf{A} = \boldsymbol{\Phi}[\mathbf{Y}_i(k), \mathbf{Y}_i(k)] + \sigma_v^2\mathbf{I}_{2k}$$

$$\mathbf{B} = \boldsymbol{\Phi}[\mathbf{Y}_i(k), \mathbf{x}_j(k+1)]$$

$$\mathbf{D} = \boldsymbol{\Phi}[\mathbf{x}_j(k+1), \mathbf{x}_j(k+1)] + \sigma_v^2\mathbf{I}_2$$

$$\mathbf{R} = \boldsymbol{\Phi}[\boldsymbol{\xi}, \mathbf{x}_j(k+1)] - \boldsymbol{\Phi}[\boldsymbol{\xi}, \mathbf{Y}_i(k)]\mathbf{A}^{-1}\mathbf{B}$$

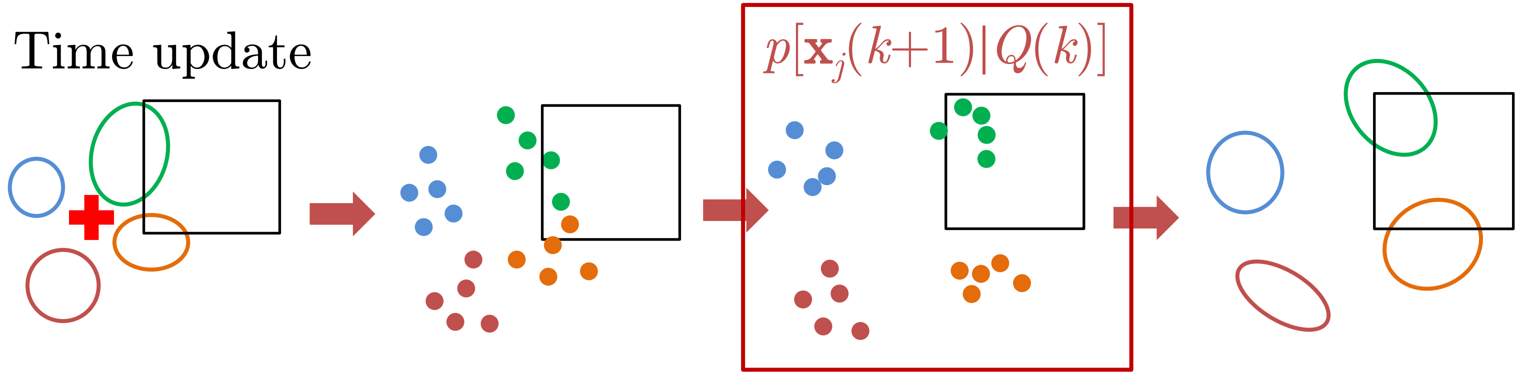
$$\mathbf{Q} = \mathbf{D} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B}$$

- $p[\mathbf{x}_j(k{+}1)|Q(k)]$ is hard to compute

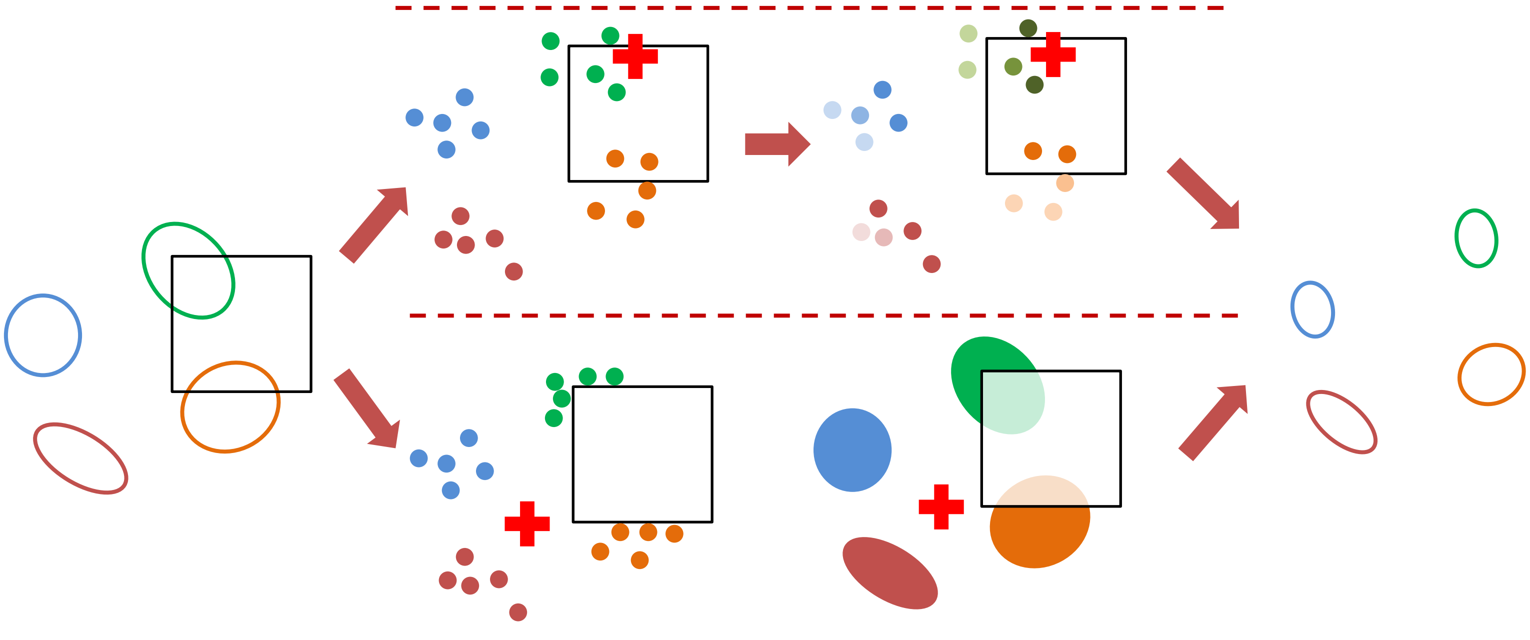
Gaussian Process Sum Particle Filter

Estimate $p[\mathbf{x}_j(k)|Q(k)] \sim \sum_{i=1}^{M} \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$

- Time update



- Measurement update



DPGP-EKLD (2)

Approximation of DPGP-EKLD

$$\hat{\varphi}\left[\boldsymbol{v};\mathbf{m}(k+1)\right] \approx \sum_{i,j} \frac{w_{ij}}{S} \sum_{\mathbf{x}_j^{(\ell)} \in \mathcal{S}(k)} h_i[\mathbf{x}_j^{(\ell)}]$$

- Theorem II: The approximation is unbiased and variance of error decreases with sample number.

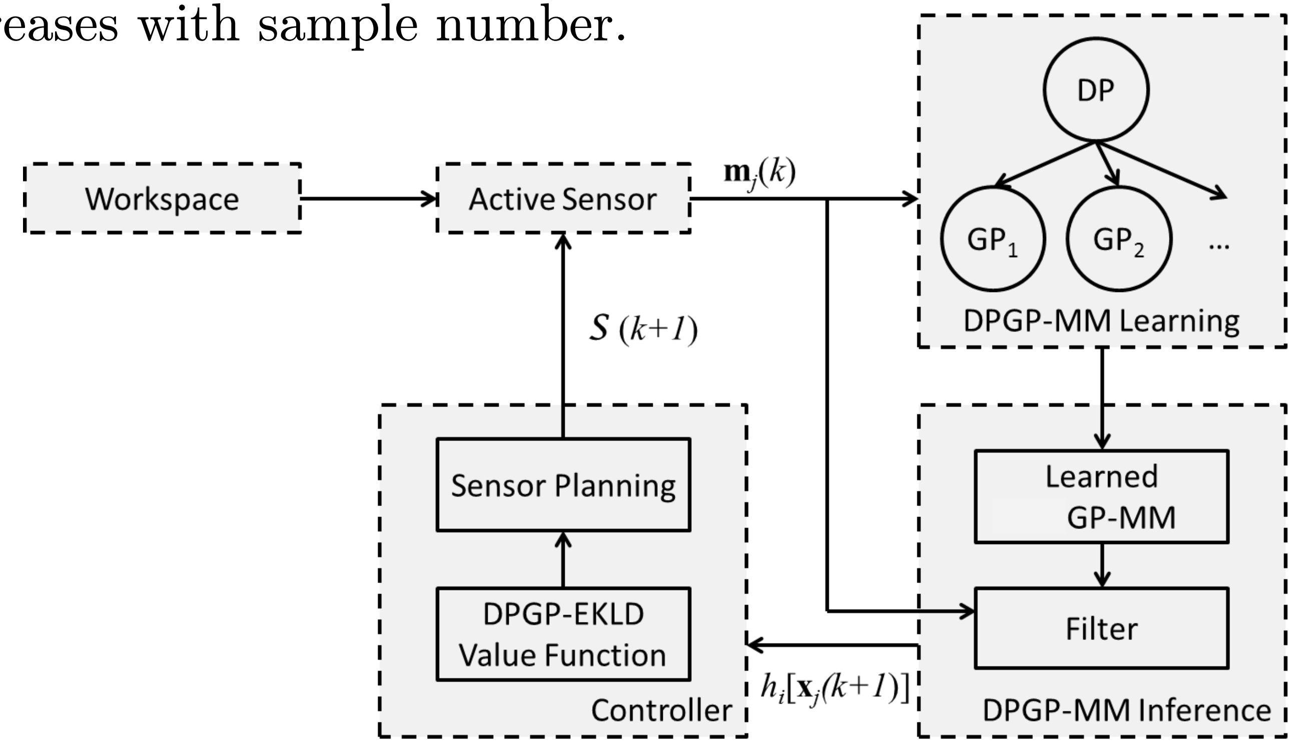
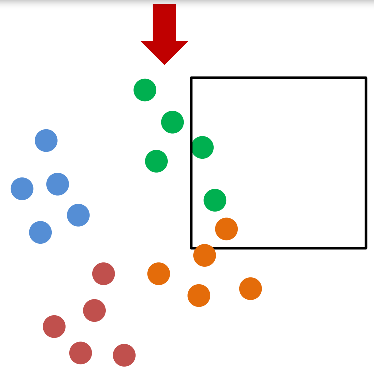


Fig. Framework of DPGP-EKLD sensor planning

Greedy DPGP-EKLD Sensor Planning Algorithm

$$\sum_{\mathbf{x}_j^{(\ell)} \in \mathcal{S}(k), i, j} \frac{w_{ij}}{S} h_i[\mathbf{x}_j^{(\ell)}]$$



Examining DPGP-EKLD

- Free-flying sensor dynamics
- FOV shape does not change

Greedy algorithm to sensor planning

- Reduce planning to weighted points cover
- Each sample is weighted by $w_{ij} h_i[\mathbf{x}_j^{(\ell)}]/S$
- Complexity using segment tree: $O(MNS \log(MNS))$

Algorithm 2 DPGP-EKLD

Require: $\theta(\cdot)$, $\phi(\cdot,\cdot)$, $\mathcal{S}$, $\boldsymbol{\xi}$, N_f

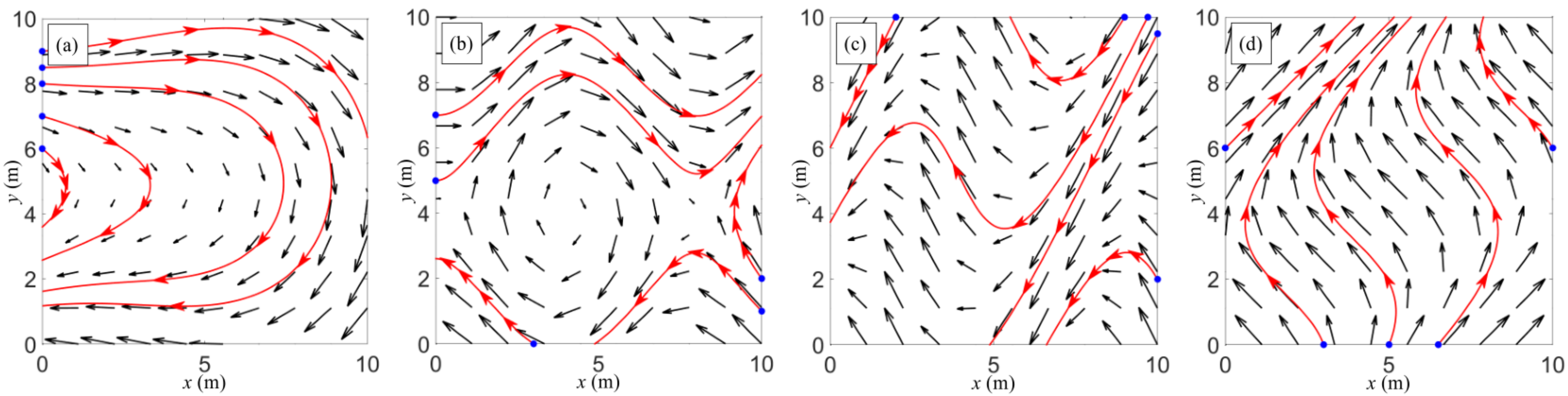
```
for k do = 1:N_f
    Sample target positions from the current target position distribution
    Propagate the target positions to the next time step
    Calculate the DPGP-EKLD for each propagated target position
    Solve the weighted sum problem for every zoom level
    Report the optimal FOV center position and the optimal zoom level
    Carry out the control
end for
```

LISC

LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Example Problem for Examining DPGP-EKLD

- Workspace: $\mathcal{W} = \{\mathrm{x} \in \mathbb{R}^2 \mid 1 \le x \le 10, 1 \le y \le 10\}$
- Sensor dynamics: free-flying objective
- Velocity fields: $\mathcal{F} = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}_4\}$



$$\mathbf{f}_1(\mathbf{x}) = \mathbf{\Phi}(\mathbf{x}, \mathbf{p}_1)\mathbf{M}_1\mathbf{O}_1$$

$$\mathbf{f}_2(\mathbf{x}) = \left[-\cos\left(\frac{\pi}{8}\mathbf{x}_{(2)}\right) \quad \sin\left(\frac{\pi}{4}\mathbf{x}_{(1)}\right)\right]^T$$

$$\mathbf{f}_3(\mathbf{x}) = \left[-0.5 \quad \sin\left([\pi/4 \quad 0.3]^T\mathbf{x}\right)\right]^T$$

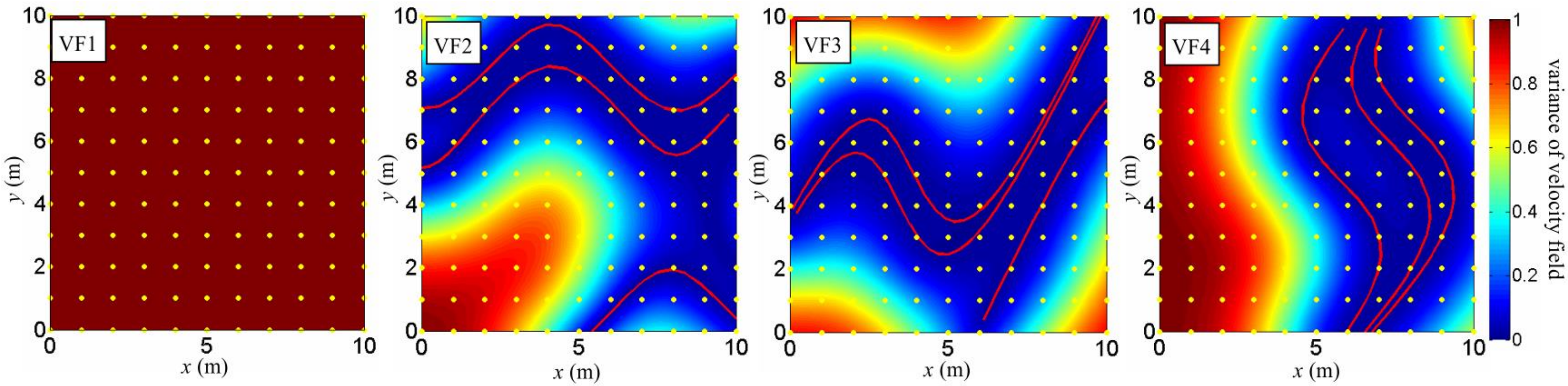
$$\mathbf{f}_4(\mathbf{x}) = \left[-\cos\left([\pi/8 \quad -0.5]^T\mathbf{x}\right) \quad 1\right]^T$$

- Probability of choosing every velocity field: $\boldsymbol{\pi} = \begin{bmatrix}\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4}\end{bmatrix}$

LISC

LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Result: Less Informative Prior



: Observed target trajectories

: Points of interest

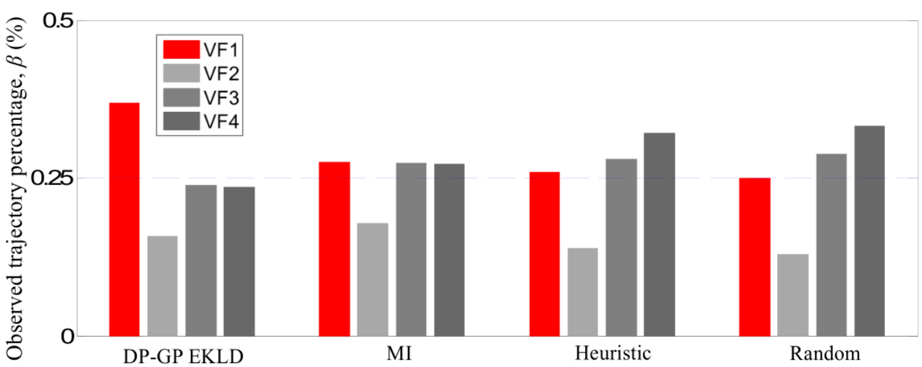


Fig. The distribution of observed trajectory percentage, averaged on the 50 runs of simulations

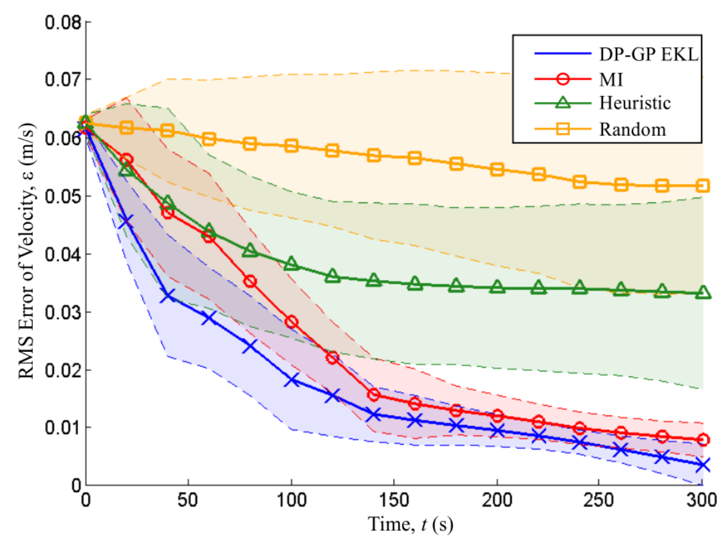


Fig. DPGP error

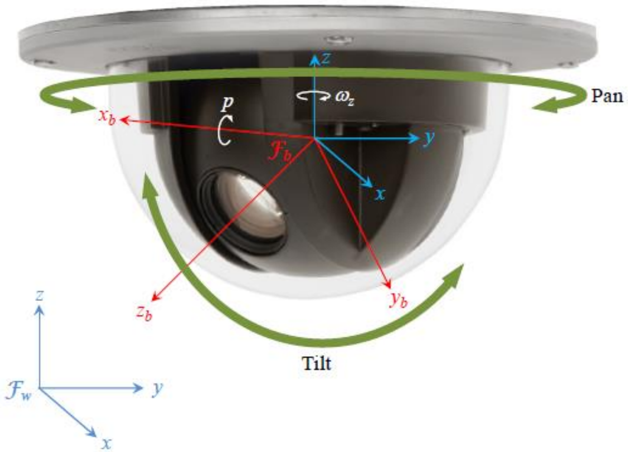
Methodology Part III: Incorporating Sensor Dynamics

Sensor Dynamics

- Linear sensor dynamics with constraints

$$\dot{\mathbf{s}}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{B}\mathbf{u}(t)$$

$$\mathbf{u}(t) \in U$$



- Example: Pan-tilt (PT) Camera dynamics:

$$\mathbf{s} = [\psi \quad \Phi \quad \dot{\psi} \quad \dot{\Phi}]^T$$

$$\mathbf{u} = [u_1 \quad u_2]^T$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ and } \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_1 & 0 \\ 0 & k_2 \end{bmatrix}$$

$$\mathbf{C}\mathbf{s} \leq \mathbf{1}_c \quad \text{and} \quad -\mathbf{1}_2 \leq \mathbf{u} \leq \mathbf{1}_2$$

LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Hardness of Using DPGP-EKLD

- Equivalence to mutual information

$$\hat{\varphi}\left(\boldsymbol{v}; M_j(k,k')\right) = \mathbb{E}_{M_j(k,k')}\left\{\mathbb{E}_{G_j}\left\{\varphi\big(\boldsymbol{v}; M(k,k')\big)\right\}\right\}$$
$$= \sum_{i=1}^{M} w_{ij} I\big(\boldsymbol{v}_i; M_j(k,k')\big)$$

- Theorem III: Given rational number m and a rational covariance matrix $\boldsymbol{\Lambda}$ over a set of Gaussian random variables $V = S \cup U$, deciding whether there exists a subset $A \subset S$ of cardinality k such that $I(A;U) \geq$ m is NP-complete.
- Theorem IV: Finding the optimal control trajectory that maximizes the DPGP-EKLD, subject to the constraints on the camera state and the control input, is NP-hard.
  - Proof by restriction
  - Targets are static
  - Sensor moves fast enough as if free-flying

Lower Bound of DPGP-EKLD

- Theorem V: The DPGP-EKLD evaluated for measurements obtained between time step k and time step k' is lower bounded by discounted summation of the mutual information as follows,

$$\hat{\varphi}\left(\boldsymbol{v}; M_j(k, k')\right) \geq \sum_{i=1,\ell=k}^{M,k'} (1-\gamma)\gamma^{\ell-k} I\left(\boldsymbol{v}_i; \mathbf{m}_j(\ell)\right)$$

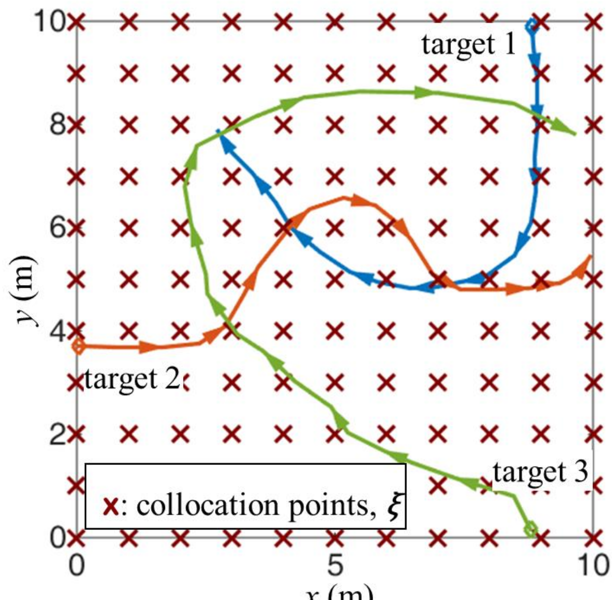


Fig. 1 Simulation scenario for validation of the additive lower bound.

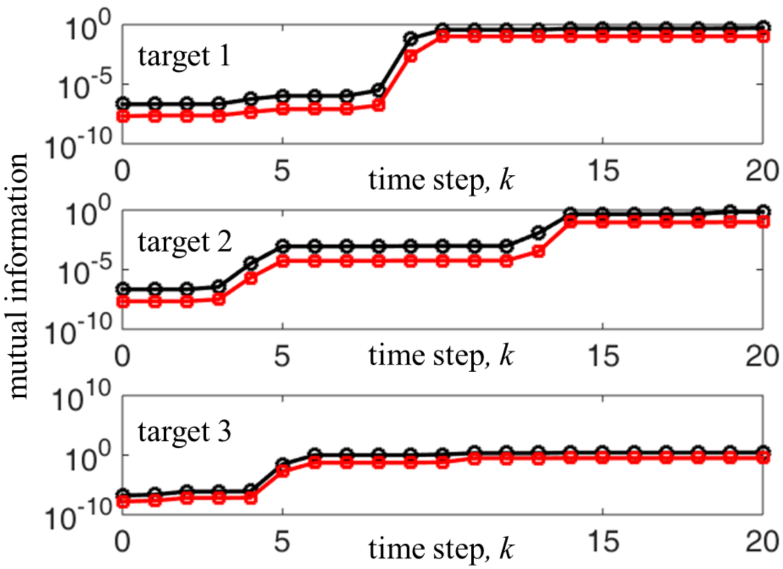


Fig. 2 DPGP-EKLD (black line) and the additive lower bound (red line)

LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Receding Horizon Control

- Objective function:

$$J_{ij} \triangleq w_{ij} \sum_{\ell=k}^{k'} (1-\gamma)\gamma^{\ell-k} I\big(\boldsymbol{v}_i; \mathbf{m}_j(\ell)\big) P\big(\mathbf{s}(\ell), \mathbf{x}_j(\ell)\big)$$

$$P(\mathbf{s}, \mathbf{x}_j) = 1 - \|\mathbf{H}\mathbf{s} - \mathbf{h}(\mathbf{x}_j)\|^2 / h$$

- Receding horizon control
  - closed-loop control strategy: target/sensor state update
  - constraints: at least finite horizon
  - infinite horizon solution obsolete after DPGP-MM update
  - lower computational complexity

$$\begin{aligned} \underset{\mathbf{u}(\ell),\ k \le \ell \le k'}{\text{maximize}} \quad & \begin{bmatrix} J_{11} & \cdots & J_{MN} \end{bmatrix}^T \\ \text{subject to} \quad & \mathbf{s}(k) = \mathbf{s}_0 \\ & \mathbf{s}(\ell+1) = \mathbf{A}\mathbf{s}(\ell) + \mathbf{B}\mathbf{u}(\ell), \ \ell = k, \ldots, k' \\ & -\mathbf{1} \le \mathbf{u}(\ell) \le \mathbf{1}, \ \ell = k, \ldots, k' \end{aligned}$$

Lexicographic Algorithm

LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Lexicographic algorithm for multiple output optimization

$$\begin{aligned} \max_{\boldsymbol{\chi}} \quad & J_i'(\boldsymbol{\chi}) \\ \text{s.t.} \quad & J_j'(\boldsymbol{\chi}) \geq J_j'^*, \ j = 1, \ldots, i-1 \\ & \boldsymbol{\chi} \triangleq [\mathbf{s}^T(k) \ \cdots \ \mathbf{s}^T(k') \ \ \mathbf{u}^T(k) \ \cdots \ \mathbf{u}^T(k')]^T \in U \end{aligned}$$

- For the same target, objective function corresponding to velocity field with higher target-VF association probability is more important

$$J_{ij} \succeq J_{i'j} \Leftrightarrow w_{ij} \geq w_{i'j}, \ \text{for } i \neq i', \ j = 1, \ldots, N$$

- For different targets, objective function corresponding to velocity field with higher target-VF association probability is also more important

$$J_{ij} \succeq J_{i'j'}, \ \text{for } i < i', \ j \neq j'$$

- When i=i ', use ideal value to determine the sequence

$$J_{ij}^I \triangleq \max_{\boldsymbol{\chi}} \{ J_{ij}(\boldsymbol{\chi}) \mid \boldsymbol{\chi} \in U \}$$

$$J_{ij} \succeq J_{ij'} \Leftrightarrow J_{ij}^I \geq J_{ij'}^I, \ \text{for } i = 1, \ldots, M, \ j \neq j'$$

Lexicographic Algorithm

- First iteration: convex quadratic programming with linear constraints

$$J_{ij} = \left(\sum_{\ell=k}^{k'} \beta(\ell) - \mathbf{c}^T \mathbf{c} \right) - \left(\chi^T \mathbf{Q}^T \mathbf{Q} \chi - \mathbf{c}^T \mathbf{Q} \chi \right)$$

where,

$$\mathbf{Q} \triangleq \begin{bmatrix} \sqrt{\frac{\beta(k)}{h}}\mathbf{H} & \mathbf{0} & \cdots & \mathbf{0} & \\ \mathbf{0} & \sqrt{\frac{\beta(k+1)}{h}}\mathbf{H} & \ddots & \vdots & \\ \vdots & \ddots & \ddots & \mathbf{0} & \mathbf{0}_{2K \times 2K} \\ \mathbf{0} & \cdots & \mathbf{0} & \sqrt{\frac{\beta(k')}{h}}\mathbf{H} & \end{bmatrix}$$

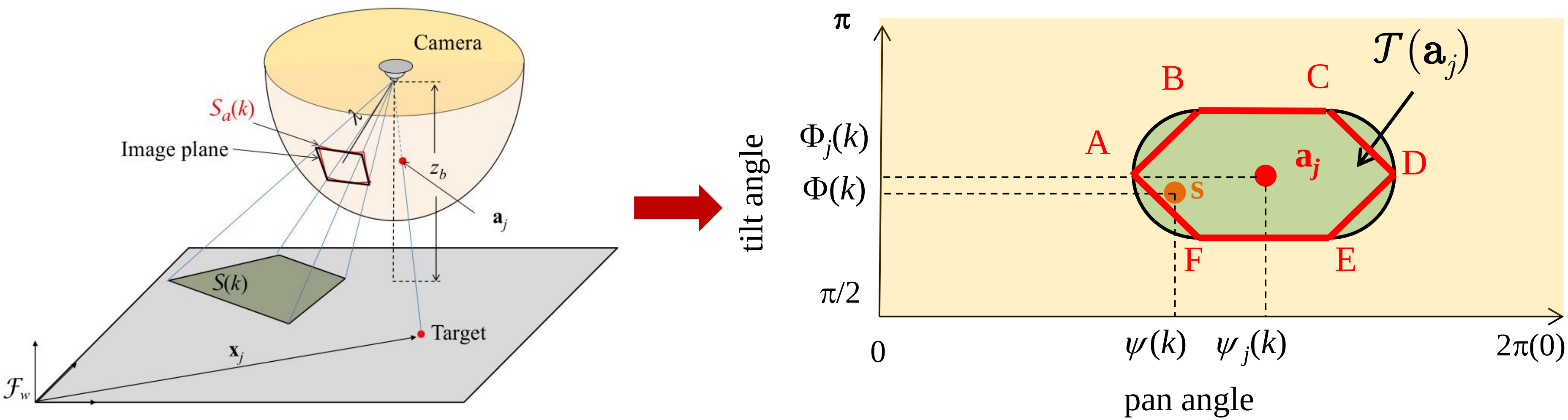
$$\mathbf{c} \triangleq \left[\sqrt{\frac{\beta(k)}{h}} \mathbf{h}^T[\mathbf{x}_j(k)] \ \cdots \ \sqrt{\frac{\beta(k')}{h}} \mathbf{h}^T[\mathbf{x}_j(k')] \right]^T$$

$$\beta(\ell) \triangleq w_{ij}(1-\gamma)\gamma^{\ell-k} I\big(\boldsymbol{v}_i; \mathbf{m}_j(\ell)\big), \ \ \ell = k, \ldots, k'$$

Lexicographic Algorithm

- The ith iterations: additional constraints

$$J_j'(\boldsymbol{\chi}) \geq J_j'^*,\ j = 1, \dots, i-1$$



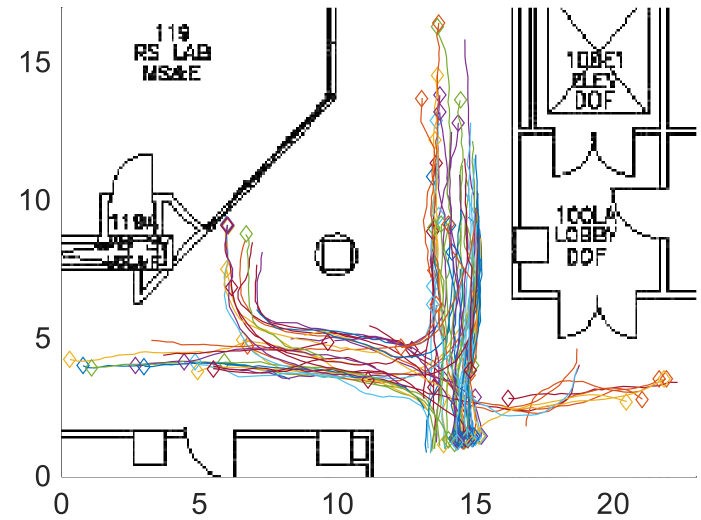
- $\mathbf{a}_j \in S_a(k) \Leftrightarrow \mathbf{s} \in \mathcal{T}(\mathbf{a}_j)$
- $\mathcal{T}(\mathbf{a}_j) : |\Delta\Phi| \leq tan^{-1}[h/(2\lambda)], \quad |\Delta\psi| \leq \tan^{-1}\left(\frac{w}{2\lambda}\sec(\Phi_j)\cos(\Delta\Phi)\right)$
- $\mathcal{T}(\mathbf{a}_j)$ is convex, symmetric, horizontal upper and lower bound
- Theorem VI: Area of polygon $ABCDEF$ divided by area of $\mathcal{T}(\mathbf{a}_j)$ is lower bound by $1 - (\sqrt{2} - 1)$^$2/2 \approx 91.4\%$, if the view angle of camera is no more than 90°.

Simulations and Results

Simulation Setup

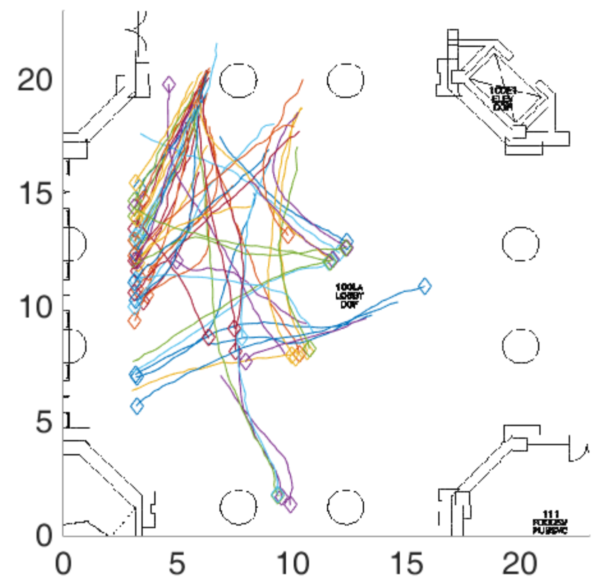
Data set 1 (thanks to Miao Liu@MIT)

- Pedestrian data set collected at Building 4, MIT
- 91 targets: 50 training, 41 testing
- Sampling time: 0.1s
- Subsampled to remove stops
- Workspace: 23m x 17 m



Data set 2 (thanks to Miao Liu@MIT)

- Pedestrian data set collected at Hallway
- 73 targets: 50 training, 23 testing
- Sampling time: 0.1s
- Subsampled to remove stops
- Workspace: 20m x 20 m



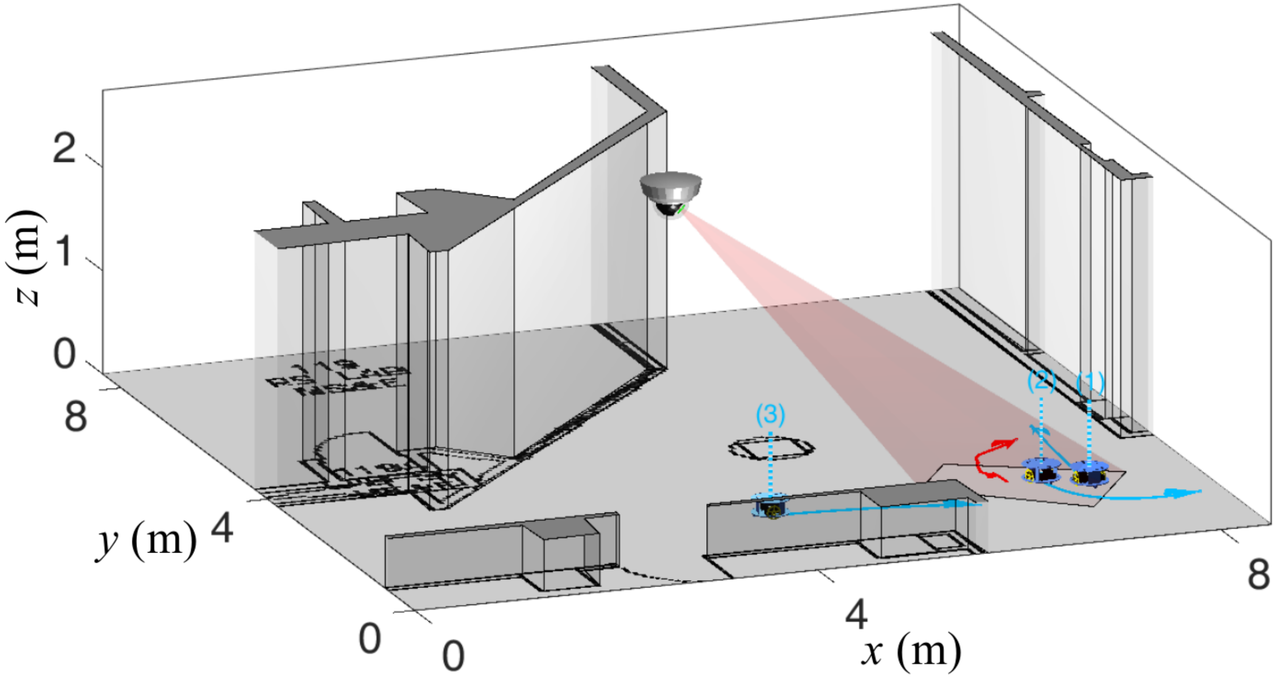
LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Simulation Setup

Sensor (simulated)

- AXIS P5624-E PTZ Dome Network Camera
- Pan: 360 degree endless, 0.2 - 350 degree/second
- Tilt: 180 degree, 0.2 - 350 degree/second
- Positioned in the center of hall way

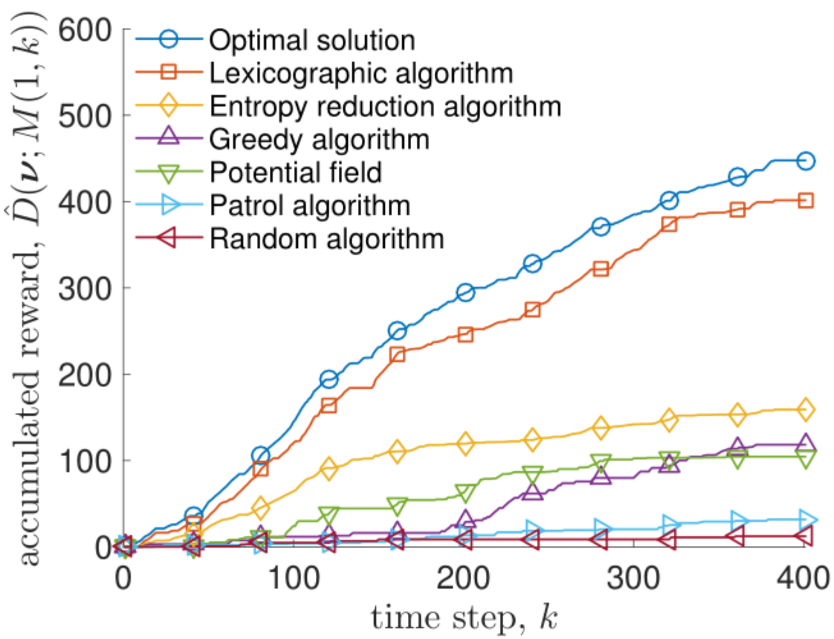
[1]

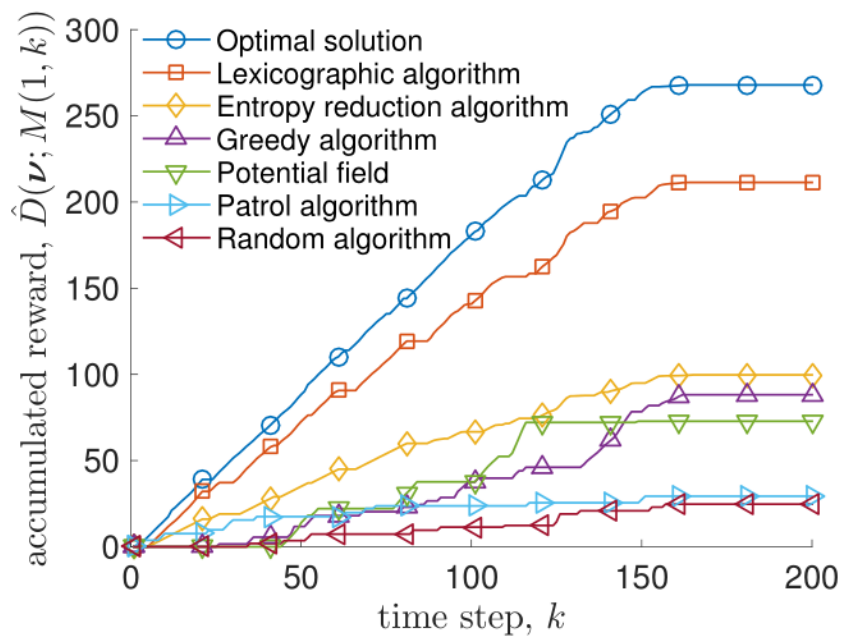


[1] http://www.axis.com/files/datasheet/ds_p5624e_1496709_en_1509.pdf

Simulation Result

- Accumulated reward by different approaches

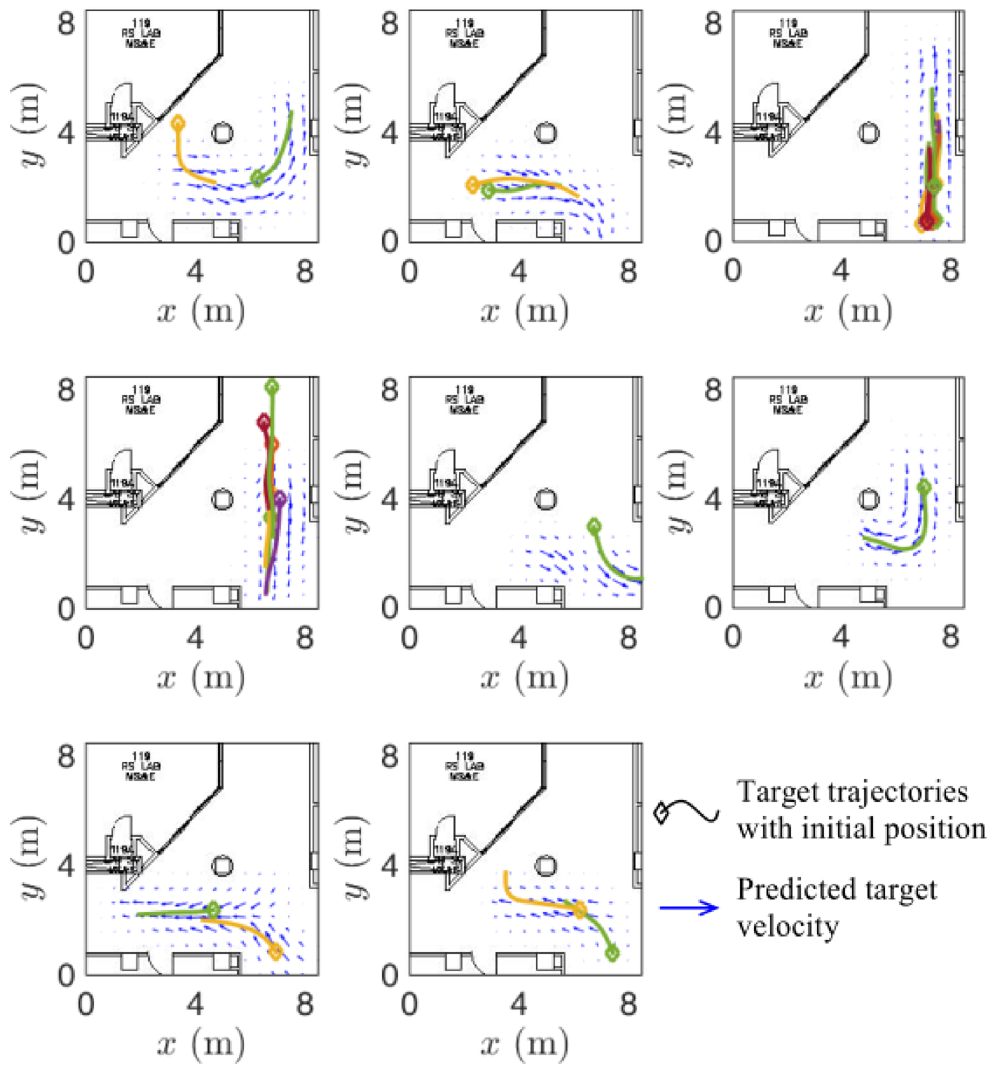




- cope with multiple targets
- limited sensor field of view
- aim at learning the change of target model

Simulation Result

- Final model obtained by the lexicographic algorithm compared with the testing target trajectories



LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Simulation Result

TABLE I
ROOT MEAN SQUARE ERROR (RMSE) OF DPGP-MM

Algorithms	All data	Optimal solution	Lexicographic	Entropy reduction	Greedy	Potential field	Patrol	Random
Bldg4 data	8.97%	9.12%	9.15%	16.25%	15.68%	29.72%	27.47%	92.81%
Hallway data	9.03%	9.58%	10.88%	18.52%	17.89%	30.21%	40.17%	93.51%

$$\epsilon = \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{M} w_{ij}\sqrt{\frac{\delta t}{(t_{f_j}-t_{0_j})}\sum_{k=1}^{(t_{f_j}-t_{0_j})/\delta t}\frac{\|\mathbf{v}_j(k)-\hat{\mathbf{v}}_j(k)\|_2^2}{\|\mathbf{v}_j(k)\|}}$$

TABLE II
COMPUTATIONAL COMPLEXITY

Algorithms	Theoretical complexity	Experimental complexity (s)	
		Bldg4 data	Hallway data
Optimal solution	NP	16.014	15.092
Lexicographic	$O\big([(L+k)^2 + T^2d^3]MNT\big)$	0.081	0.073
Entropy reduction	$O\big((k^2 + T^2d^3)MNT\big)$	0.077	0.072
Greedy	$O\big([(L+k)^2 + \log(MN)]MN\big)$	0.044	0.044
Information potential	$O\big([(L+k)^2 + d]MN\big)$	0.003	0.003
Patrol	$O(1)$	< 0.001	< 0.001
Random	$O\big((L+k)^2MN + dT\big)$	0.002	0.002

LISC
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Conclusion

- Information driven sensor planning
- Target dynamics modelling
  - GP regression for nonlinear dynamics
  - Dirichlet process for clustering
- Information value (DPGP-EKLD) as utility function
  - Free-flying sensor dynamics
  - Linear dynamics with constraints
- Future work
  - Nonlinear sensor dynamics model
  - Decentralized control

…

Thank you!