LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Value Function Approximation for Multiscale Dynamical Systems

Pingping Zhu, Julian Morelli
Advisor : Silvia Ferrari
Laboratory for Intelligent Systems and Controls

Conference for Decision and Control
Las Vegas
December 14, 2016

LISC

Adaptive Planning for Intelligent Collaborative Systems

- Overview:
  - Multi-agent collection of autonomous robotic vehicles, $O(10^2)$
    - Large temporal and spatial scales
  - Uncertainty in environmental conditions
- Applications:
  - Coastal & Environmental monitoring
  - Surveillance
  - Search and Rescue
  - Recognizance

United States Coast Guard Rescue Team [1]

REMUS Autonomous submarine [2]

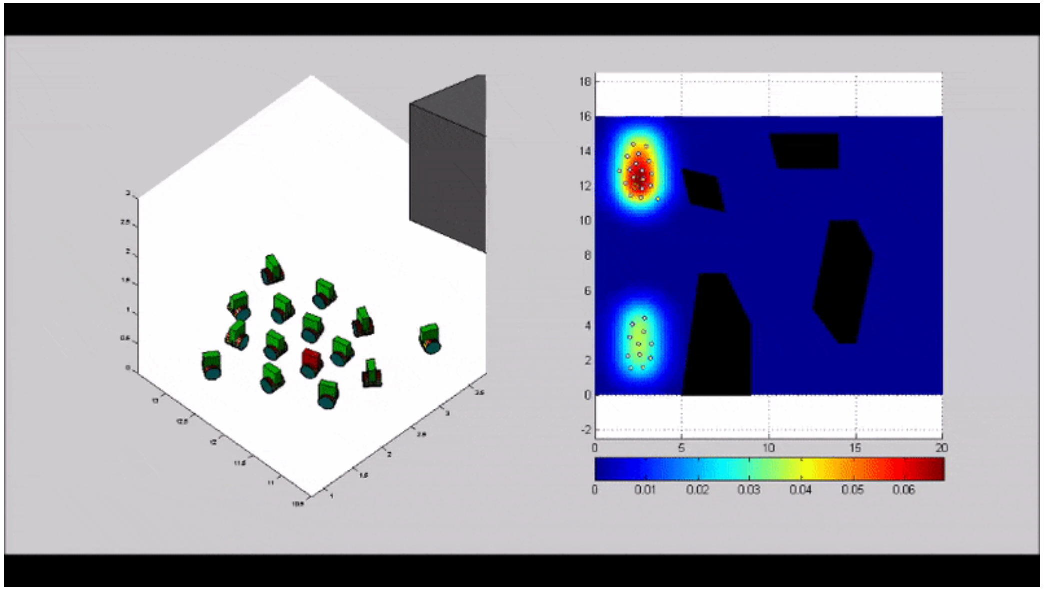
1. uscg.mil
2. auvac.org

LISC

Background

- Multi-Agent Planning Methods:
  - Prioritized Path Planning
  - Multi-Agent Potential Field Methods
  - Distributed Optimal Control
- Limitations:
  - High computational complexity
  - Sub- or non-optimal
  - Non-adaptive/Offline methods
- Distributed Optimal Control:
  - Currently best method for a problem this size
  - Numerical solution via Generalized Reduced Gradient Method

- Technical Challenges:
  - Hundreds of ODEs
    - Nonlinear dynamics
  - Functional approximation
  - Rapidly changing environment



LISC

Distributed Optimal Control

- Consider N autonomous robotic agents with dynamics described by the following system of ordinary differential equations:

$$\dot{\mathbf{x}}_i(t) \quad = \quad \mathbf{f}[\mathbf{x}_i(t), \mathbf{u}_i(t), t] + \mathbf{G}\mathbf{w}_i(t)$$

- Macroscopic state, *restriction operator*: $\wp(\mathbf{x}, t)$ $\Phi : \mathcal{X} \times \mathbb{R} \to \mathcal{P}$
- Macroscopic dynamics represented by the *advection-diffusion* equation:
  - $\mathbf{G}$ is a constant matrix, $\boldsymbol{w}_i(t)$ is an additive Gaussian noise
  - Where $\mathbf{v} \equiv \dot{\boldsymbol{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathrm{t})$ and $\nu = \nabla(\mathbf{G}\mathbf{G}^{\mathrm{T}})$

$$\frac{\partial \wp}{\partial t} = -\nabla \cdot [\wp(\mathbf{x}, t)\mathbf{v}] + \frac{1}{2}\nabla \cdot [(\mathbf{G}\mathbf{G}^T)\nabla \wp(\mathbf{x}, t)]$$
$$= -\nabla \cdot [\wp(\mathbf{x}, t)\mathbf{f}(\mathbf{x}, \mathbf{u}, t)] + \nu \nabla^2 \wp(\mathbf{x}, t)$$

Distributed Optimal Control

- Cost function in terms of restriction operator

$$J = \phi[\wp(\cdot, T_f)] + \int_{T_0}^{T_f} \int_{\mathcal{X}} \mathscr{L}[\wp(\mathbf{x}, t), \mathbf{u}(\mathbf{x}, t), t] d\mathbf{x}_i dt$$

- PDF restriction operator must also satisfy: $P(\mathbf{x} \in B, t) = \int_B \wp(\mathbf{x}, t) d\mathbf{x}$

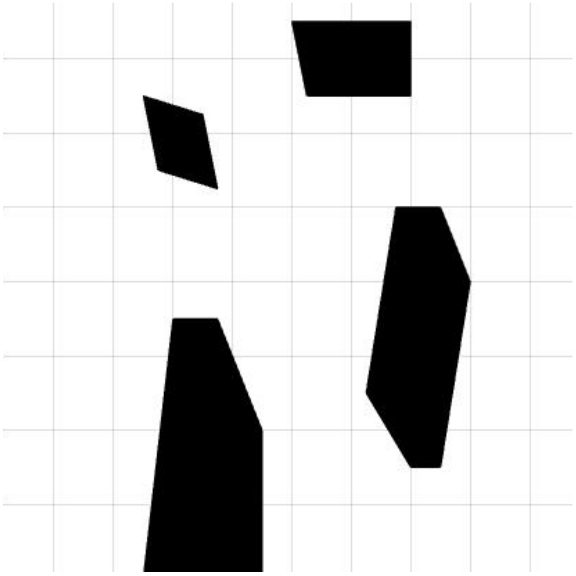
and these other conditions:

$$\begin{aligned}
&\text{I.C.:} & \wp(\mathbf{x}, T_0) &= \wp_0(\mathbf{x}); \\
&\text{B.C.:} & \wp(\mathbf{x} \in \partial\mathcal{X}, t) &= 0, \quad \forall t \in [T_0, T_f]; \\
&\text{A.C.:} & \int_{\mathcal{X}} \wp(\mathbf{x}, t) d\mathbf{x}_i &= 1; \\
& & \wp(\mathbf{x} \notin \partial\mathcal{X}, t) &= 0, \quad \forall t \in [T_0, T_f].
\end{aligned}$$

LISC

Distributed Optimal Control

- Currently the most effective way of solving optimal control of large multiscale systems
- Limitations:
  - Offline Method / Non-adaptive
  - Must know *a priori:*
    - Microscopic agent dynamics
    - Macroscopic evolution equation
    - Definition of the restriction operator
    - Environmental conditions



Obstacle Configuration

LISC

Research Problem and Motivation

- Workspace: $\mathcal{X} \subset \mathbb{R}^2$
- Agent Dynamics: $\dot{\mathbf{x}}_i(t) \quad = \quad \mathbf{f}[\mathbf{x}_i(t), \mathbf{u}_i(t), t] + \mathbf{G}\mathbf{w}_i(t)$
- Restriction Operator: $\wp(\mathbf{x}, t)$ $\Phi : \mathcal{X} \times \mathbb{R} \to \mathcal{P}$

  Gaussian Mixed Model
  Expectation Maximization algorithm
- Cost Function: $J = \phi[\wp(\cdot, T_f)] + \int_{T_0}^{T_f} \int_{\mathcal{X}} \mathscr{L}[\wp(\mathbf{x}, t), \mathbf{u}(\mathbf{x}, t), t] d\mathbf{x}_i dt$
- Optimal Value Function: $\mathcal{V}^* = \min_{\mathbf{u}_k} \left\{ \int_{\mathcal{X}} \mathscr{L}(\wp_k^*, \mathbf{u}_k) d\mathbf{x}_i + \mathcal{V}^*[\wp_{k+1}^*, \mathcal{C}^*(\cdot)] \right\}$

LISC

Adaptive Critics

- Dynamic Programming optimizes a *Value function:*

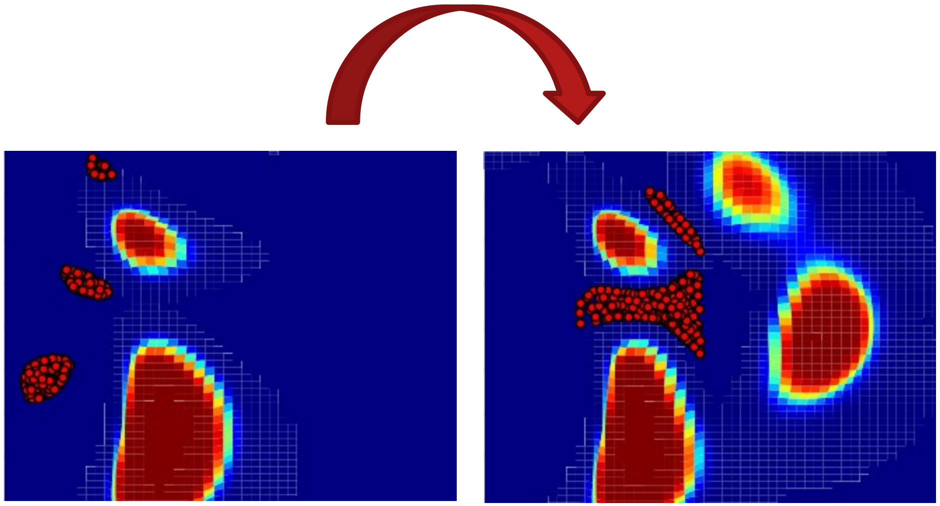
$$\mathcal{V}^* = \min_{\mathbf{u}_k} \left\{ \int_{\mathcal{X}} \mathscr{L}(\wp_k^*, \mathbf{u}_k) d\mathbf{x}_i + \mathcal{V}^*[\wp_{k+1}^*, \mathcal{C}^*(\cdot)] \right\}$$

  - Hamilton-Jacobi-Bellman Equation

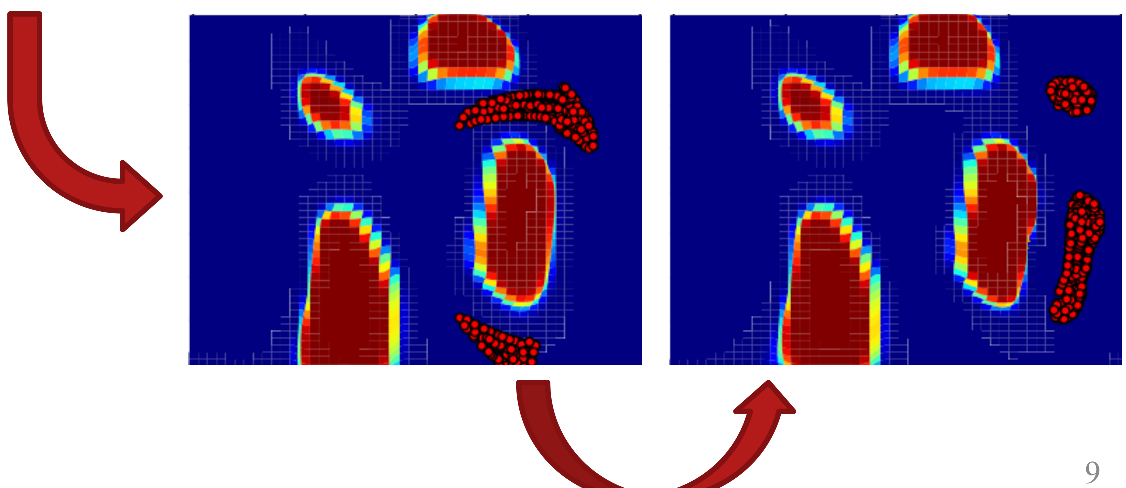
- Associated with a cost function (at time k):

$$\mathscr{L}(\wp_k) = \int_{\mathbf{x}\in\mathcal{X}} \underbrace{\wp_k(\mathbf{x}) \log \frac{\wp_k(\mathbf{x})}{g(\mathbf{x})}}_{\text{Kullback-Leibler Divergence}} + \underbrace{\wp_k(\mathbf{x}) U_{\text{rep}}}_{\text{Obstacle Repulsion}} + \underbrace{\wp_k(\mathbf{x}) e^{\{w_u[u_1(x,k)^2 + u_2(x,k)^2]/2\}}}_{\text{Energy Consumption}} \, d\mathbf{x}$$

Adaptive Critics



$$\mathscr{L}(\wp_k) = \int_{\mathbf{x}\in\mathcal{X}} \overbrace{\wp_k(\mathbf{x}) \log \frac{\wp_k(\mathbf{x})}{g(\mathbf{x})}}^{\text{KL-Divergence}} + \overbrace{\wp_k(\mathbf{x}) U_{\text{rep}}}^{\text{Obstacle Repulsion}} + \underbrace{\wp_k(\mathbf{x}) e^{\{w_u[u_1(x,k)^2 + u_2(x,k)^2]/2\}}}_{\text{Energy Consumption}} \, d\mathbf{x}$$



LISC

Adaptive Critics

- Given a value function corresponding to a control law, and improved control law can be obtained as follows:

$$\mathcal{C}_{\ell+1}(\wp_k) = \arg\min_{\mathbf{u}_k} \left\{ \int_{\mathcal{X}} \mathscr{L}(\wp_k, \mathbf{u}_k) d\mathbf{x} + \mathcal{V}[\wp_{k+1}, \mathcal{C}_\ell(\cdot)] \right\}$$

- Given a control law, the value function can be updated according to the following rule:

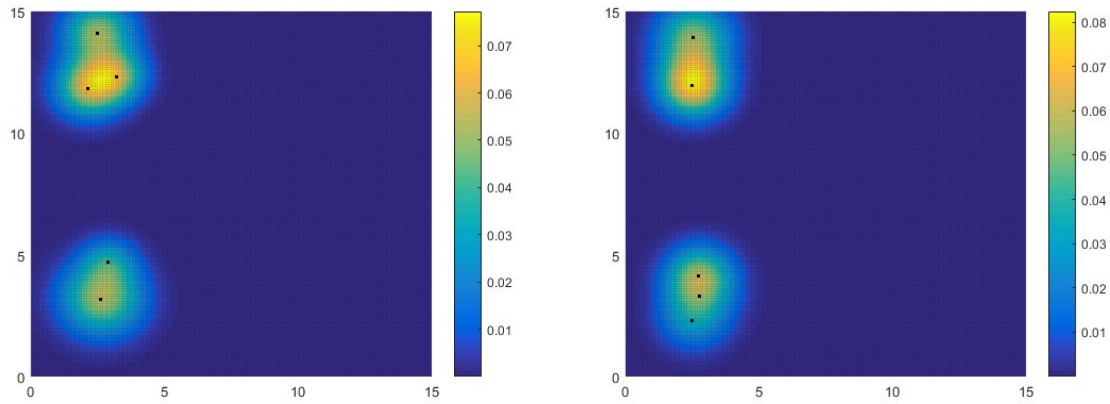
$$\boxed{\mathcal{V}_{\ell+1}[\wp_k, \mathcal{C}(\cdot)] = \int_{\mathcal{X}} \mathscr{L}(\wp_k, \mathbf{u}_k) d\mathbf{x} + \mathcal{V}_\ell[\wp_{k+1}, \mathcal{C}(\cdot)]}$$

Gaussian Mixed Model

LISC

Gaussian Mixed Model

- GMM used to approximate macroscopic state PDF
- Form: $$\tilde{\wp}_k(\mathbf{x}) = \sum_{\tau=1}^{n} \omega_\tau \mathcal{N}(\mathbf{x}|\mu_{k,\tau}, \Sigma_{k,\tau})$$
- Parameters cannot be ordered!
  - Parameter tube is discontinuous with respect to the time step $(\omega_{k,\tau}, \mu_{k,\tau}, \Sigma_{k,\tau})$



Projection onto Fixed Basis

LISC

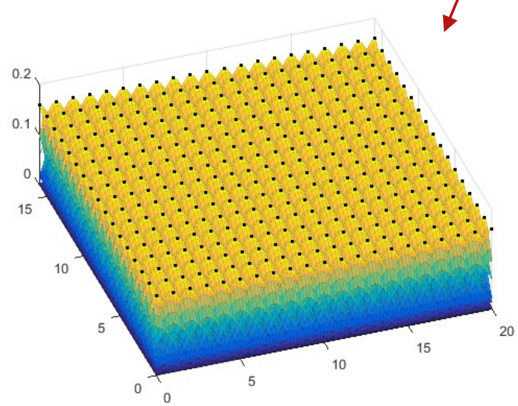
Projection onto Fixed Basis

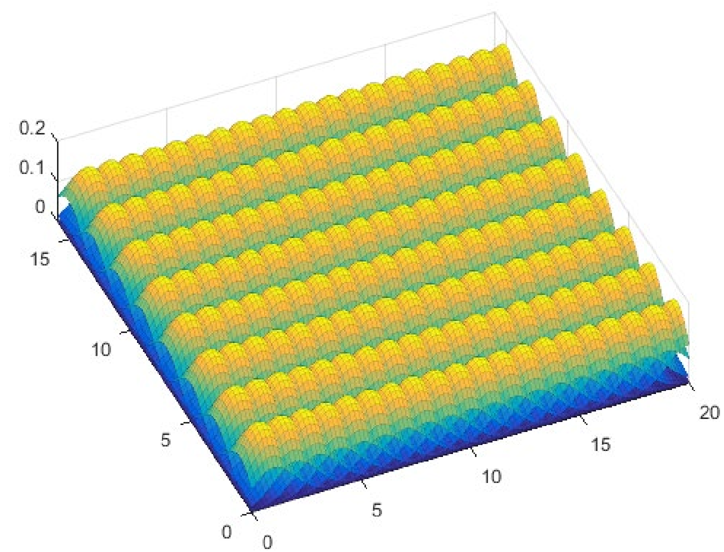
- Project onto fixed basis of Gaussians using inner product:

$$\boldsymbol{p}_k = I_j(\tilde{\wp}_k) = \int_{\mathbf{x}\in\mathcal{X}} \tilde{\wp}_k(\mathbf{x}) b_j(\mathbf{x}) d\mathbf{x}, \quad \text{for } j = 1, ..., m$$

- $\boldsymbol{p}_k$ is a coefficient vector for the fixed basis set $\{b_j\}_{j=1\dots m}$

- Gaussian basis functions





LISC

Projection onto Fixed Basis

- Inner product can be computed in closed form:
  - *Gaussian Identity Property*:

$$\sum_{\tau=1}^{n} \int_{\mathbf{x}\in\mathcal{X}} \omega_\tau \mathcal{N}(\mathbf{x}|\mu_{k,\tau}, \Sigma_{k,\tau})\mathcal{N}(\mathbf{x}|\mu_j, \Sigma_j)d\mathbf{x} = \sum_{\tau=1}^{n} \omega_\tau \mathcal{N}(\mu_{k,\tau}|\mu_j, \Sigma_{k,\tau} + \Sigma_j)$$

- Using the normalized basis previously defined:

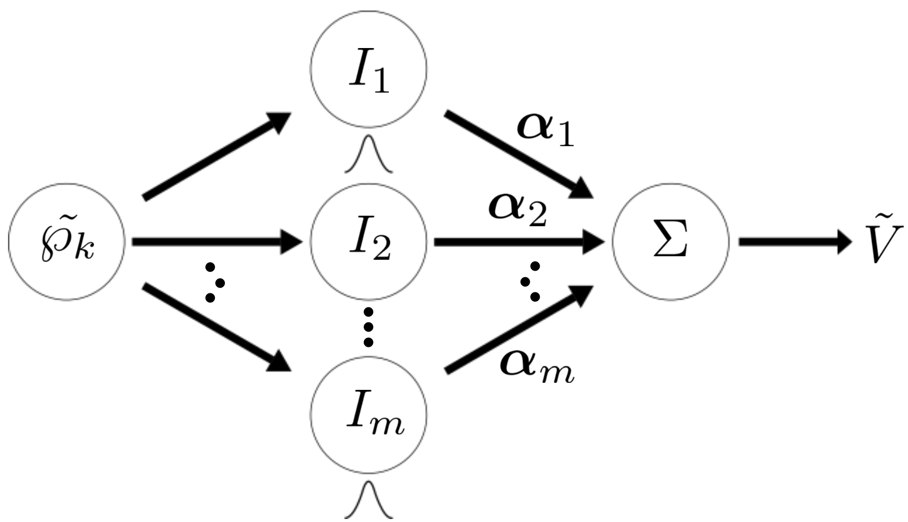
$$\begin{aligned} \boldsymbol{p}_k = I_j(\tilde{\wp}_k) &= \langle \tilde{\wp}_k, \frac{b_j}{\|b_j\|} \rangle_{\mathcal{P}} \\ &= \frac{\sum_{j=1}^{n} \omega_\tau \mathcal{N}(\mu_{k,\tau}|\mu_j, \Sigma_{k,\tau} + \Sigma_j)}{\mathcal{N}(\mu_j|\mu_j, 2\Sigma_j)}, \end{aligned}$$

Functional Learning by Temporal Difference

LISC

Value Function Learning

- Uses existing Temporal Difference (TD) and Recursive Least Squares Temporal Difference (RLSTD):
  - Learn parameters $\boldsymbol{\alpha}$ such that: $\tilde{V}(\tilde{\wp}_k) = \boldsymbol{\alpha}^T \mathbf{p}_k$



Value Function Learning

- Batch Learning (Offline)
  - Matrix **A** is created with the outer product
  - Vector **b** is created with the inner product between **z** and R

$$\boldsymbol{A} = \sum_{i=0}^{t} \boldsymbol{z}_i (\boldsymbol{p}_i - \boldsymbol{p}_{i+1}))^T, \quad \boldsymbol{b} = \sum_{i=0}^{t} \boldsymbol{z}_i R_i$$

  - where

$$\boldsymbol{z}_{t+1} = \lambda \boldsymbol{z}_t + \boldsymbol{p}_{t+1}, \; \boldsymbol{z}_{t_0} \triangleq \boldsymbol{p}_{t_0}$$

- Solve with matrix inverse:

$$\boldsymbol{\alpha} = \boldsymbol{A}^{-1} \boldsymbol{b}$$

LISC

Value Function Learning

- Recursive Learning (Online)
  - e is an error term
  - C is an intermediate term to compute parameters

$$\boldsymbol{e}_t = R_t - (\boldsymbol{p}_t - \gamma \boldsymbol{p}_{t+1})^T \boldsymbol{\alpha}_{t-1}$$

$$\boldsymbol{C}_t = \boldsymbol{C}_{t-1} - \frac{\boldsymbol{C}_{t-1}\boldsymbol{p}_t(\boldsymbol{p}_t - \gamma \boldsymbol{p}_{t+1})^T \boldsymbol{C}_{t-1}}{1 + (\boldsymbol{p}_t - \gamma \boldsymbol{p}_{t+1})^T \boldsymbol{C}_{t-1}\boldsymbol{p}_t}$$

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} + \frac{\boldsymbol{C}_{t-1}}{1 + (\phi_t - \gamma \boldsymbol{p}_{t+1})^T \boldsymbol{C}_{t-1}\boldsymbol{p}_t} \boldsymbol{e}_t \boldsymbol{p}_t$$

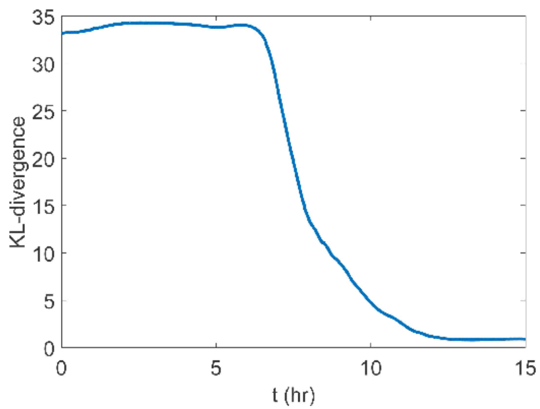
LABORATORY FOR INTELLIGENT SYSTEMS AND CONTROLS

Results and Conclusions

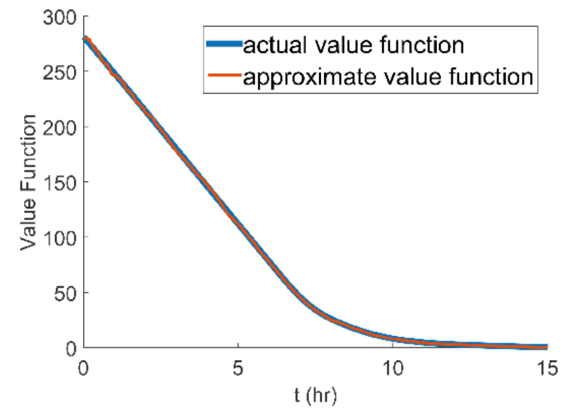
LISC

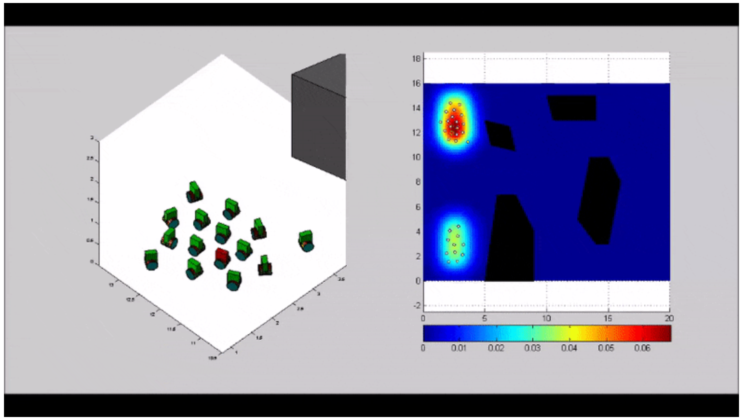
Results

Actual Cost Function

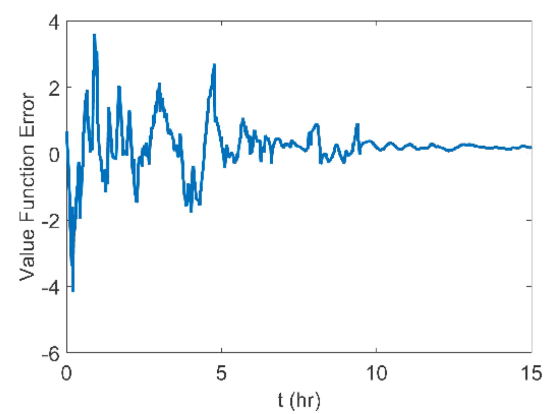


Actual and Approximate Value function





Error:



LISC

Conclusions

- Describe a collective of agents with a restriction operator
- Approximate the optimal value function on-line from the parameters of the PDF
  - Project GMM onto fixed basis
  - Using existing temporal difference algorithms
- Future Work:
  - Learn the optimal control law on-line by iteratively improving it

Value Function Approximation for Multiscale Dynamical Systems

Pingping Zhu, Julian Morelli
Advisor : Silvia Ferrari
Laboratory for Intelligent Systems and Controls

Conference for Decision and Control

Las Vegas

December 14, 2016

Julian Morelli
jam888@cornell.edu

LISC

Gaussian Identity Property

$$
\begin{aligned}
\boldsymbol{p}_k = I_j(\tilde{\wp}_k) &= \int_{\mathbf{x}\in\mathcal{X}} \tilde{\wp}_k(\mathbf{x}) b_j(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathbf{x}\in\mathcal{X}} \left[\sum_{\tau=1}^{n} \omega_\tau \mathcal{N}(\mathbf{x}|\mu_{k,\tau}, \Sigma_{k,\tau}) \right] \mathcal{N}(\mathbf{x}|\mu_j, \Sigma_j) d\mathbf{x} \\
&= \sum_{\tau=1}^{n} \int_{\mathbf{x}\in\mathcal{X}} \omega_\tau \mathcal{N}(\mathbf{x}|\mu_{k,\tau}, \Sigma_{k,\tau}) \mathcal{N}(\mathbf{x}|\mu_j, \Sigma_j) d\mathbf{x} \\
&= \sum_{\tau=1}^{n} \omega_\tau \mathcal{N}(\mu_{k,\tau}|\mu_j, \Sigma_{k,\tau} + \Sigma_j).
\end{aligned}
$$

LISC

Gaussian Identity Property

$$
\begin{aligned}
I_j(\tilde{\wp}_k) &= \langle \tilde{\wp}_k, \frac{b_j}{\|b_j\|} \rangle_{\mathcal{P}} = \frac{\langle \tilde{\wp}_k, b_j \rangle_{\mathcal{P}}}{\|b_j\|} = \frac{\langle \tilde{\wp}_k, b_j \rangle_{\mathcal{P}}}{\langle b_j, b_j \rangle_{\mathcal{P}}} \\
&= \frac{\int_{\mathbf{x}\in\mathcal{X}} \tilde{\wp}_k(\mathbf{x}) b_j(\mathbf{x}) d\mathbf{x}}{\int_{\mathbf{x}\in\mathcal{X}} b_j(\mathbf{x}) b_j(\mathbf{x}) d\mathbf{x}} \\
&= \frac{\sum_{j=1}^{n} \omega_\tau \mathcal{N}(\mu_{k,\tau} | \mu_j, \Sigma_{k,\tau} + \Sigma_j)}{\mathcal{N}(\mu_j | \mu_j, 2\Sigma_j)},
\end{aligned}
$$