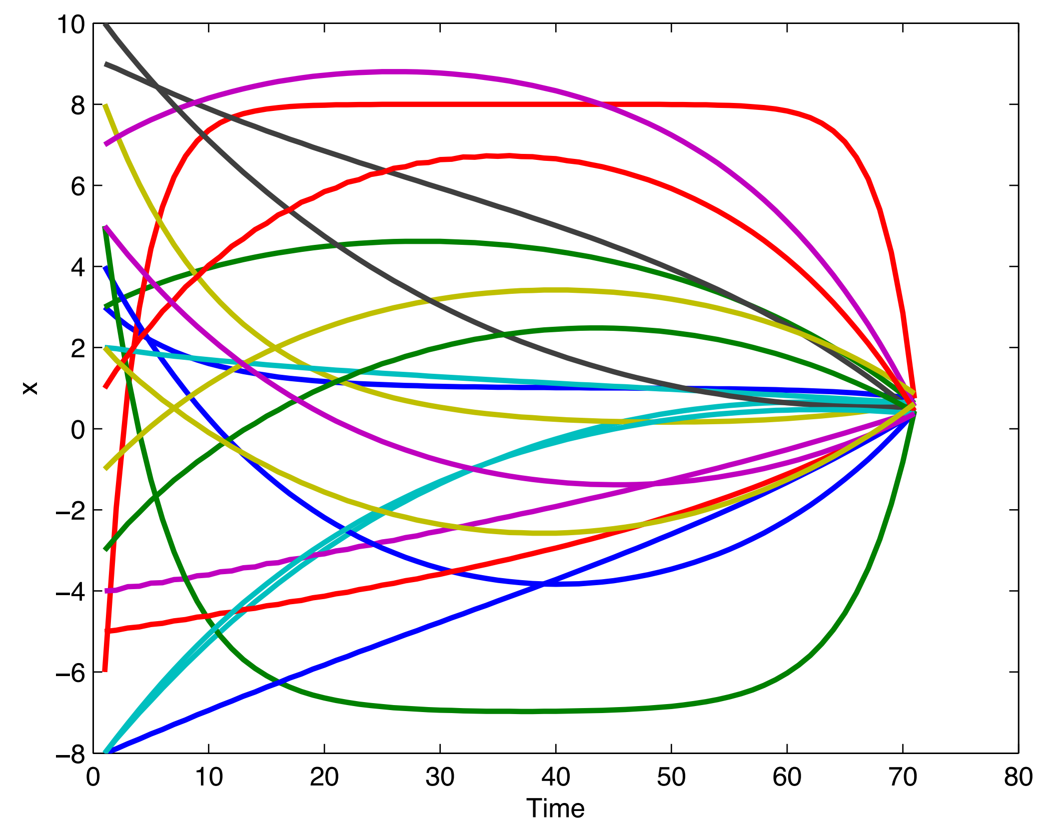
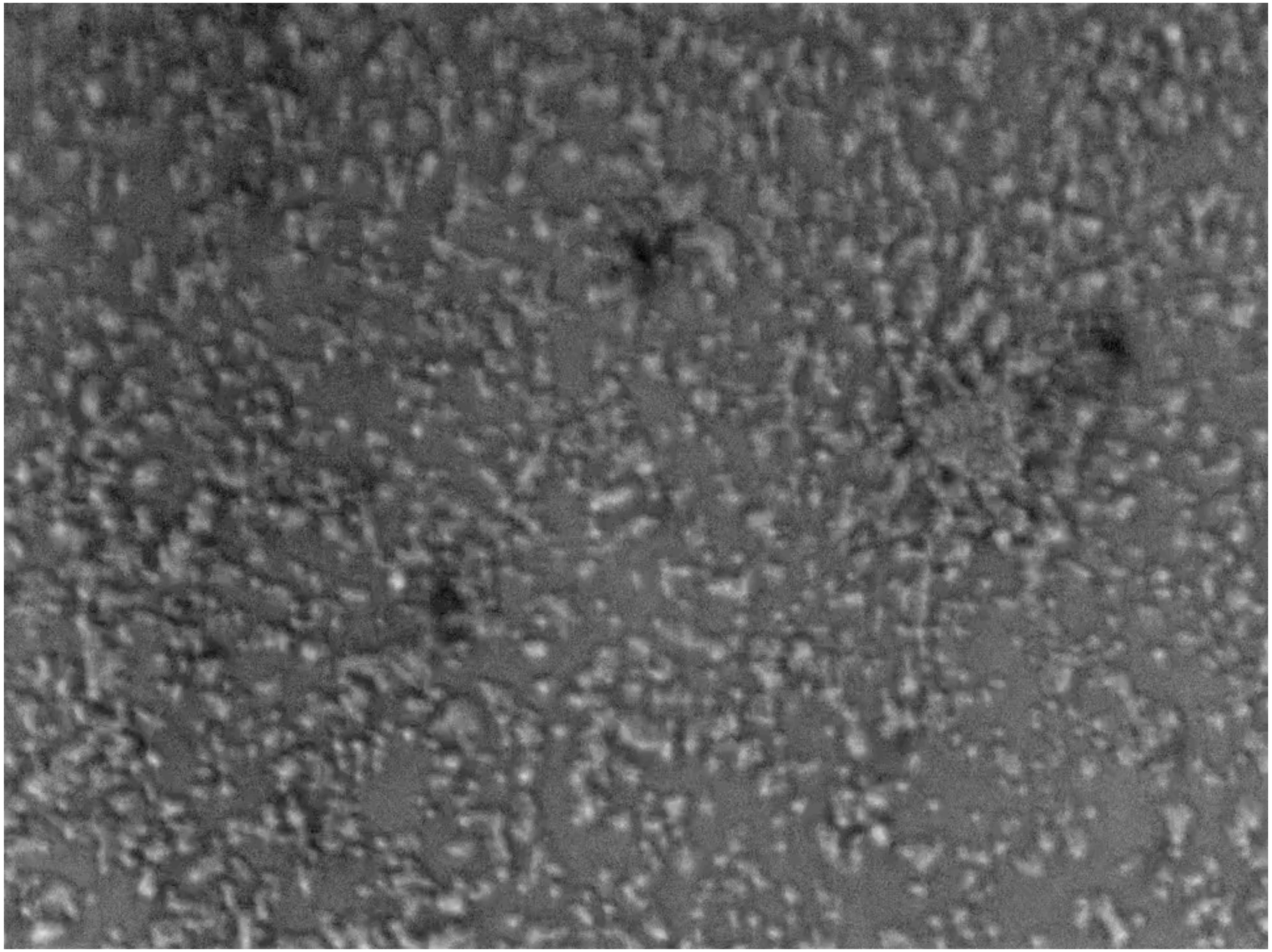
Distributed Negotiation Methods for Multi-Agent Dynamical Systems



Daniel Zelazo
Faculty of Aerospace Engineering
Technion-Israel Institute of Technology

Freiburg University
July 16, 2014

Coordination in Multi-agent Systems



Goldbeter, Bulletin of Mathematical Biology 2006

Aggregation of Dictyostelium

Team-Players or Selfish?

Origins Space Missions

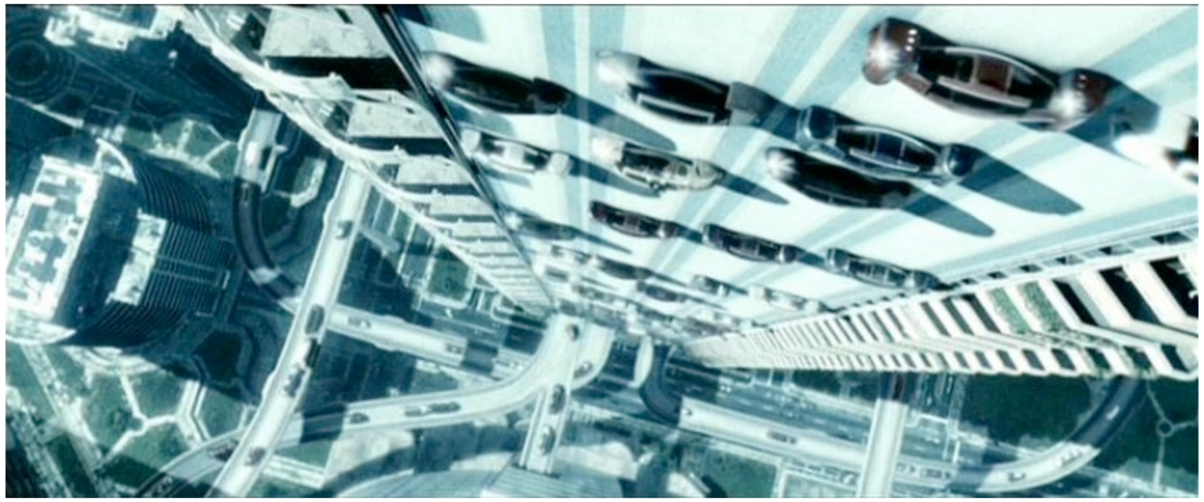
mission success depends on precise coordination and control of all agents in the system

all agents acting to achieve a *common team objective*

optimization perspective

$$\min_{x_i} J(x_1, \ldots, x_n)$$

Team-Players or Selfish?



Minority Report

Automated Transportation Networks

coordination of agents is only needed to safely complete their individual mission

all agents acting to minimize *selfish objectives*

<u>optimization perspective</u>

$$\min_{x_i} \sum_{i=1}^{n} J_i(x_i)$$

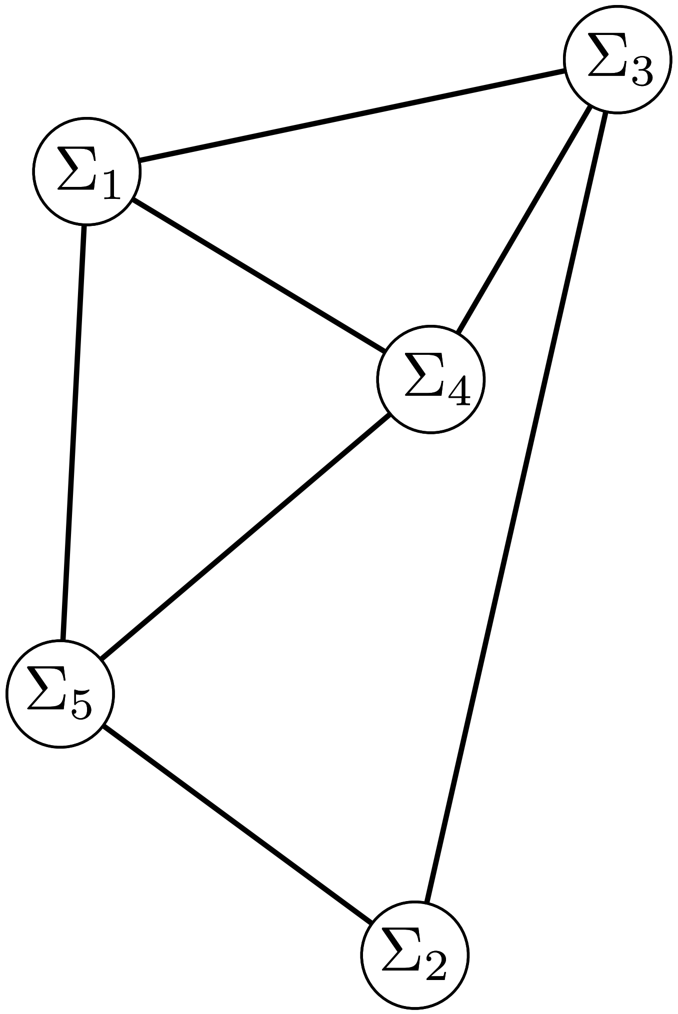
This Talk...

A *Preference Agreement Problem*

a team of *selfish* dynamical systems

coupled by a strict *team constraint*

real-time requirements



Shrinking Horizon Preference Agreement Algorithm

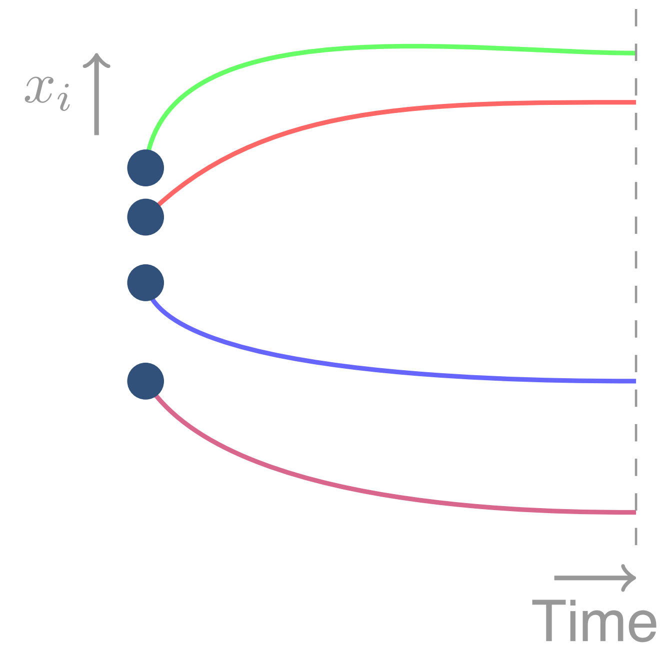
Preliminaries

a collection of *n* agents

∗discrete time
∗integrator dynamics

$$x_i(t+1) = x_i(t) + u_i(t)$$

preference is captured by associated objective functions



∗quadratic objective
∗different weights and desired state for each agent

$$J_i(t_0, T, x_i, u_i) = \frac{1}{2}\left(\sum_{t=t_0}^{T-1} q_i(x_i(t+1) - \xi_i)^2 + r_i u_i(t)^2\right)$$

agents coupled by a *terminal time agreement constraint*

$$x_i(T) = \cdots = x_n(T)$$

Preliminaries

agents can communicate over a network

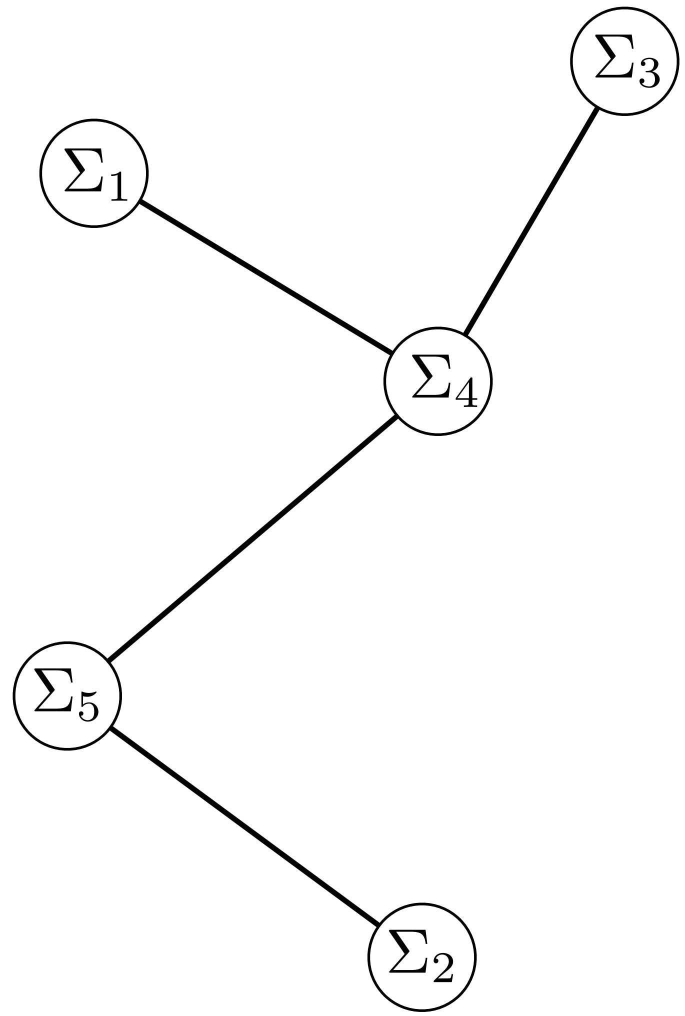
*fixed spanning tree

$$\mathcal{G} = (\mathcal{V}, \mathcal{E})$$

$$E(\mathcal{G}) \in \mathbb{R}^{n \times n-1}$$

node-edge incidence matrix

$$E(\mathcal{G}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \\ -1 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$



Preliminaries

agents can communicate over a network

∗fixed spanning tree

agents coupled by a *terminal time agreement constraint*

$$x_i(T) = \cdots = x_n(T)$$

$$\Updownarrow$$

$$E(\mathcal{G})^T x(T) = 0$$

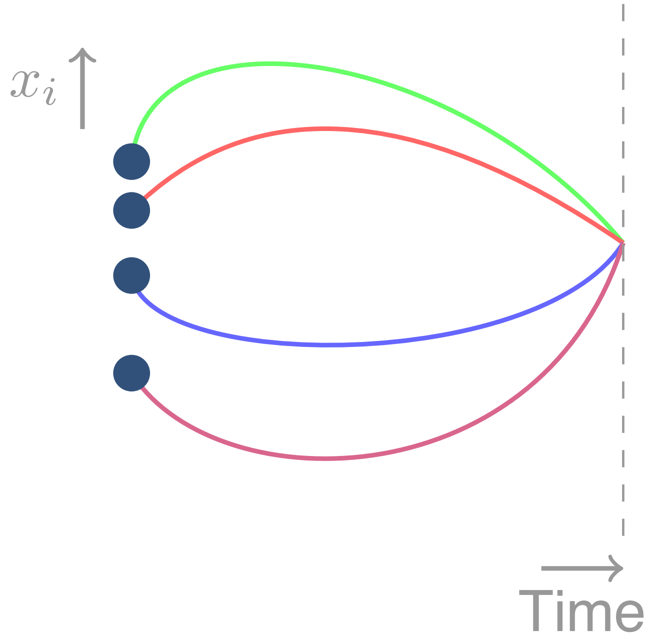
An Optimal Control Problem

the *centralized* approach

$$OCP(t_0, T, x_0): \min_{x,u} \sum_{i=1}^{n} J_i(t_0, T, x_i, u_i)$$

$$\text{s.t.} \quad x(t+1) = x(t) + u(t),\ x(t_0) = x_0$$

$$E(\mathcal{G})' x(T) = 0.$$



can be reformulated as a *quadratic program*

$$x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix}$$

An Optimal Control Problem

the *centralized* approach

$$
\begin{aligned}
OCP(t_0, T, x_0): \quad \min_{x,u} \quad & \sum_{i=1}^{n} J_i(t_0, T, x_i, u_i) \\
\text{s.t.} \quad & x(t+1) = x(t) + u(t), \;\; x(t_0) = x_0 \\
& E(\mathcal{G})' x(T) = 0.
\end{aligned}
$$

$$
\min_{x,u} \frac{1}{2} \left(\left[\begin{array}{cc} x^T & u^T \end{array} \right] \left[\begin{array}{cc} Q & \\ & R \end{array} \right] \left[\begin{array}{c} x \\ u \end{array} \right] + \left[\begin{array}{cc} F(Q,\xi)^T & 0 \end{array} \right] \left[\begin{array}{c} x \\ u \end{array} \right] \right)
$$

$$
s.t. \; A \left[\begin{array}{c} x \\ u \end{array} \right] = b
$$

An Optimal Control Problem

recall: Quadratic programs with only equality constraints have an *analytic solution*

$$\text{QP:} \quad \min_{x} \quad \frac{1}{2}x^T Q x + c^T x$$
$$s.t. \quad Ax = b$$

① Form the Lagrangian

$$\mathcal{L}(x,\lambda) = \frac{1}{2}x^T Q x + c^T x + \lambda^T (Ax - b)$$

'Lagrange' multiplier

② First-order optimality conditions

a linear equation!

$$\nabla_x \mathcal{L}(x,\lambda) = Qx + c + A^T\lambda = 0$$
$$\nabla_\lambda \mathcal{L}(x,\lambda) = Ax - b = 0$$

$$\Rightarrow \begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}$$

An Optimal Control Problem

recall: Quadratic programs with only equality constraints have an *analytic solution*

QP:
$$\min_x \quad \frac{1}{2}x^T Q x + c^T x$$
$$s.t. \quad Ax = b$$

① Form the Lagrangian

$$\mathcal{L}(x,\lambda) = \frac{1}{2}x^T Q x + c^T x + \lambda^T (Ax - b)$$

'Lagrange' multiplier

② First-order optimality conditions

$$\nabla_x \mathcal{L}(x,\lambda) = Qx + c + A^T\lambda = 0$$

$$\Rightarrow x^* = -Q^{-1}(A^T\lambda + c)$$

optimal solution is parameterized by the Lagrange multiplier

An Optimal Control Problem

recall: Quadratic programs with only equality constraints have an *analytic solution*

$$\text{QP:} \quad \min_x \quad \frac{1}{2}x^T Q x + c^T x$$
$$s.t. \quad Ax = b$$

③ Form the 'dual' function

$$g(\lambda) = \min_x \frac{1}{2}x^T Q x + c^T x + \lambda^T (Ax - b)$$
$$\Rightarrow g(\lambda) = -\frac{1}{2}\lambda^T A Q^{-1} A^T \lambda - b^T \lambda \quad (c = 0)$$

$$\Rightarrow x^* = -Q^{-1}(A^T \lambda + c)$$

④ Solve the 'dual problem'

$$\max_\lambda g(\lambda)$$

An Optimal Control Problem

the *centralized* approach

$$
\begin{aligned}
OCP(t_0, T, x_0):\ \min_{x,u} \quad & \sum_{i=1}^{n} J_i(t_0, T, x_i, u_i) \\
\text{s.t.} \quad & x(t+1) = x(t) + u(t),\ \ x(t_0) = x_0 \\
& E(\mathcal{G})' x(T) = 0.
\end{aligned}
$$

Lagrange duality motivates an iterative algorithm to solve a quadratic program

A Distributed Algorithm

$$\begin{aligned} OCP(t_0, T, x_0): \min_{x,u} \quad & \sum_{i=1}^{n} J_i(t_0, T, x_i, u_i) \\ \text{s.t.} \quad & x(t+1) = x(t) + u(t),\ x(t_0) = x_0 \\ & E(\mathcal{G})' x(T) = 0. \end{aligned}$$

dual sub-gradient algorithm

the (partial) Lagrangian

$$\mathcal{L}(\mathbf{x}, \mathbf{u}, \mu) = \sum_{i=1}^{n} J_i(t_0, T, \mathbf{x}_i, \mathbf{u}_i) + \underline{\mu'} E(\mathcal{G})' \mathbf{x}(T)$$

Multipliers are associated with the *edges* in the graph

separable form of the Lagrangian

$$\mathcal{L}(\mathbf{x}, \mathbf{u}, \gamma) = \sum_{i=1}^{n} J_i(t_0, T, \mathbf{x}_i, \mathbf{u}_i) + \underline{\gamma'} \mathbf{x}(T)$$

uniquely defined on “nodes”

$$\gamma = E(\mathcal{G})\mu$$

A Distributed Algorithm

the (partial) Lagrangian

$$\mathcal{L}(\mathbf{x}, \mathbf{u}, \mu) \quad = \quad \sum_{i=1}^{n} J_i(t_0, T, \mathbf{x}_i, \ \mathbf{u}_i) + \mu' E(\mathcal{G})' \mathbf{x}(T)$$

recall the first-order optimality conditions

$$\nabla_{\mu} \mathcal{L}(\mathbf{x}, \mathbf{u}, \mu) = E(\mathcal{G})' \mathbf{x}(T)$$

(separable form)

$$\nabla_{\gamma} \mathcal{L}(\mathbf{x}, \mathbf{u}, \gamma) = \mathbf{x}(T)$$

the dual problem

$$\max_{\mu} g(\mu)$$

A quadratic program!

can be solved using a *gradient ascent!*

A Distributed Algorithm

dual sub-gradient algorithm

① Solve *local* quadratic program $QP_i(k)$

$$(\hat{\mathbf{x}}_i^{[k+1]}, \hat{\mathbf{u}}_i^{[k+1]}) = \arg\min_{\hat{\mathbf{x}}_i^{[k]}, \hat{\mathbf{u}}_i^{[k]}} J_i(t_0, T, \hat{\mathbf{x}}_i^{[k]}, \hat{\mathbf{u}}_i^{[k]}) + \hat{\gamma}_i^{[k]} \hat{\mathbf{x}}_i^{[k]}(T)$$

$$\text{s.t. Dynamic Constraints}$$

② Update multipliers

$$\hat{\gamma}_i^{[k+1]} = \hat{\gamma}_i^{[k]} + \alpha^{[k]} L(\mathcal{G}) \hat{\mathbf{x}}^{[k+1]}(T)$$

$^{*} L(\mathcal{G}) = E(\mathcal{G}) E(\mathcal{G})^T$

✳ multiplier updated by inter-agent communication
✳ choice of step-size is non-trivial - required for convergence
✳ *asymptotically* converges to the primal optimal solution

Not good enough...

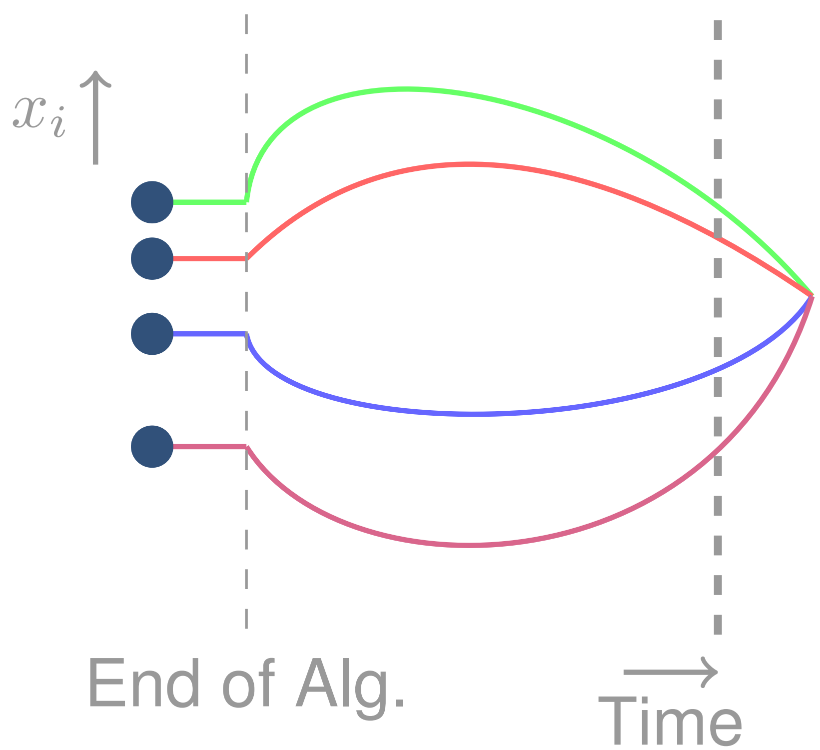
$$\lim_{k\to\infty} (\hat{\mathbf{x}}^{[k]}, \hat{\mathbf{u}}^{[k]}, \hat{\gamma}^{[k]}) = (\overline{\mathbf{x}}, \overline{\mathbf{u}}, E(\mathcal{G})\overline{\mu})$$

$$OCP(t_0, T, x_0)$$

infinity is a *long* time!

$$\infty > T$$

∗assume T is a *hard deadline*
∗agents do not want to wait around to compute their trajectories
∗communication also takes time

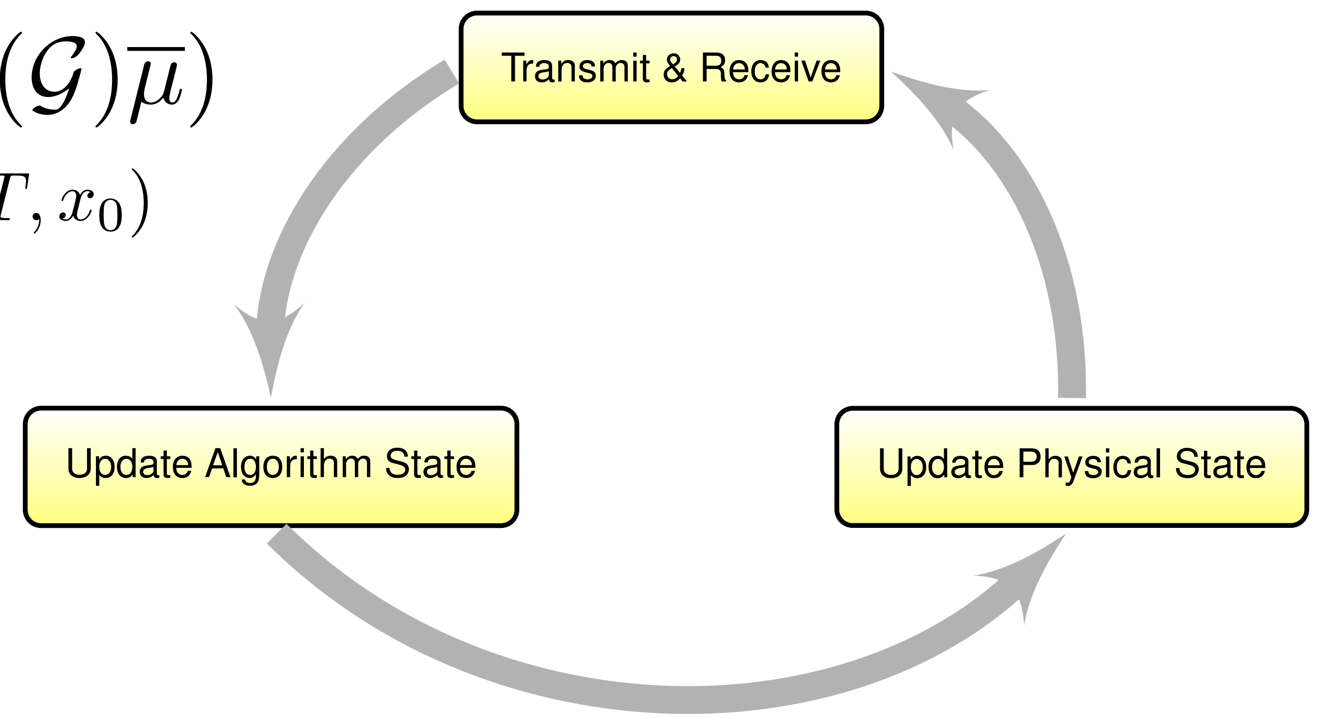


"wait and solve" can lead to significant disagreement

'Real-Time' Modification

$$\lim_{k\to\infty}(\hat{\mathbf{x}}^{[k]}, \hat{\mathbf{u}}^{[k]}, \hat{\gamma}^{[k]}) = (\overline{\mathbf{x}}, \overline{\mathbf{u}}, E(\mathcal{G})\overline{\mu})$$

$$OCP(t_0, T, x_0)$$



Requirements

∗at each time-step, agents *move* in a direction they consider optimal

∗agents communicate at each time-step to *negotiate* the terminal-state constraint → agents are trying to estimate the multiplier value

∗trajectories are updated to reflect progress in the negotiation process

A *dynamic negotiation process!*

"Shrinking Horizon"

Shrinking Horizon Preference Agreement (SHPA) Algorithm

for $t := 0$ **to** *T-1* **do**

$$\gamma^t = E\mu(t), \tilde{T} = T - t$$

① Solve *local* quadratic program $QP_i(k)$

$$\min_{\hat{\mathbf{x}}_i(t), \hat{\mathbf{u}}_i(t)} J_i(t, T, \hat{\mathbf{x}}_i^t, \hat{\mathbf{u}}_i^t) + \gamma_i^t \hat{\mathbf{x}}_i^t(T)$$

$$\text{s.t. } \hat{\mathbf{x}}_i^t = \mathbb{1}_{\tilde{T}} x_i(t) + B_{\tilde{T}} \hat{\mathbf{u}}_i^t$$

② *Propagate* physical state and *update* multipliers

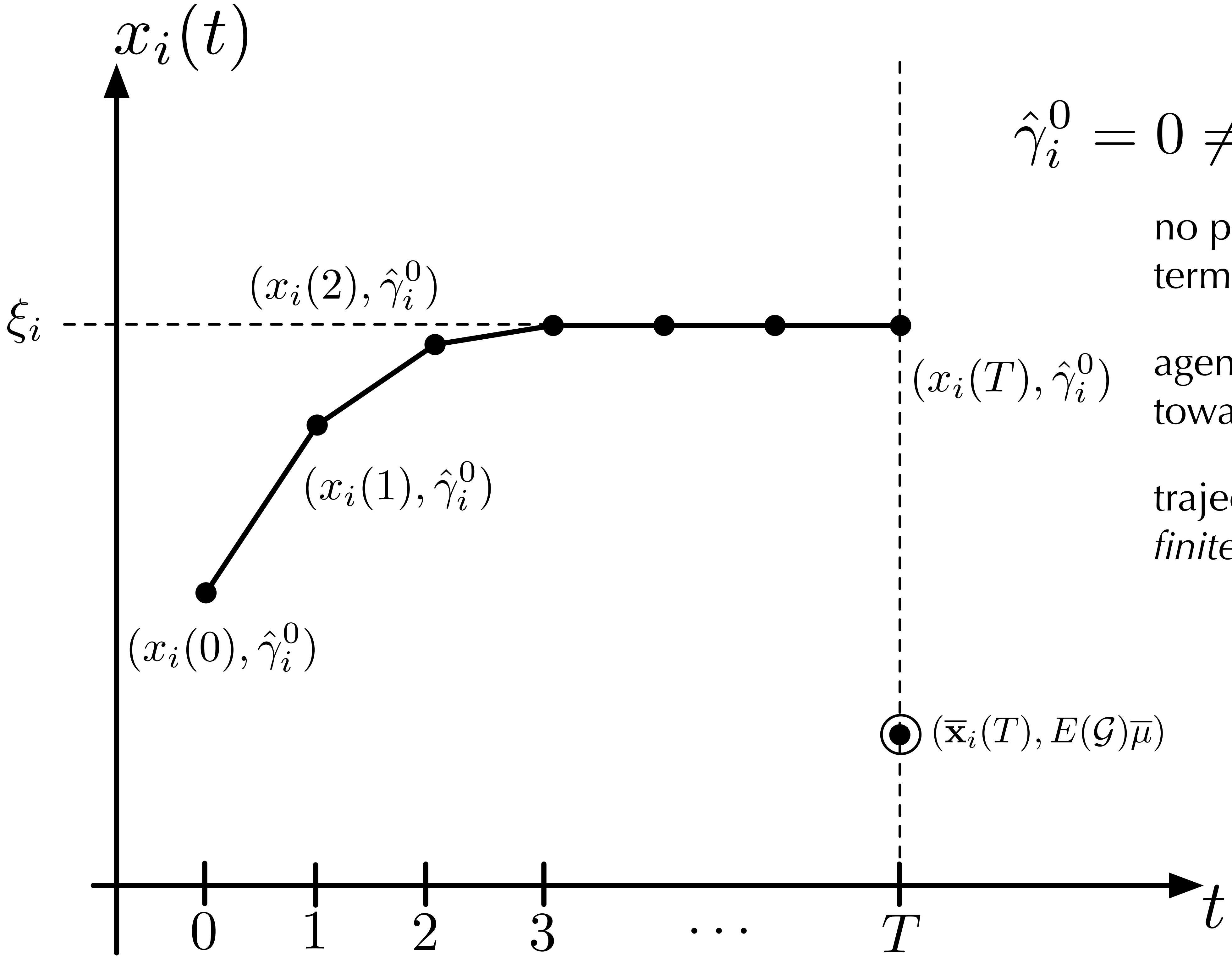
$$x_i(t+1) = x_i(t) + \hat{\mathbf{u}}_i^t(t), \; i = 1, \ldots, n$$

$$\mu(t+1) = \mu(t) + \alpha(t) E' \hat{\mathbf{x}}^t(T)$$

∗ optimization horizon is "shrinking" from "the left"

∗choice of step-size is non-trivial

"Shrinking Horizon"



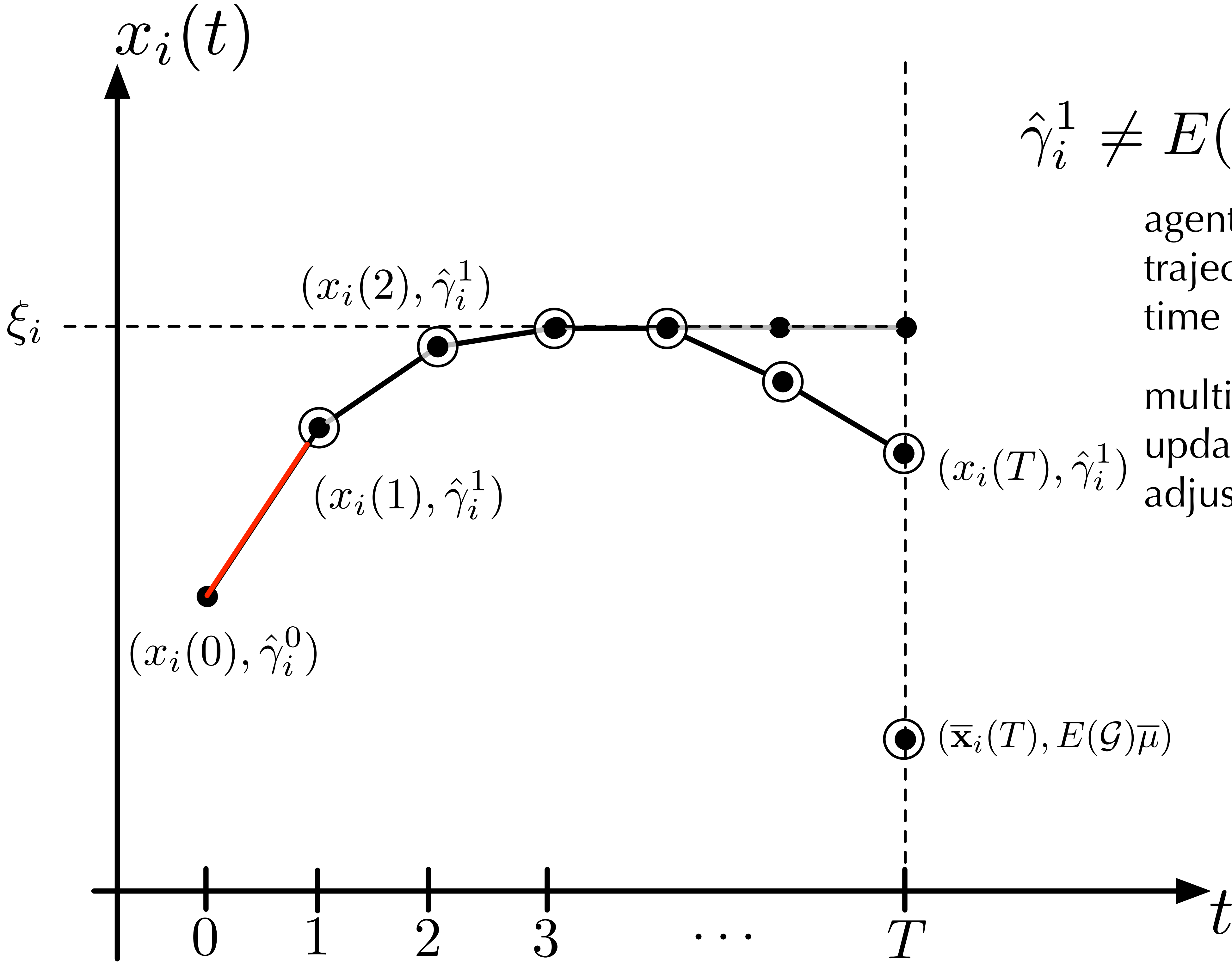
$$\hat{\gamma}_i^0 = 0 \neq E(\mathcal{G})\overline{\mu}$$

no penalty for ignoring terminal constraint

agent moves directly towards preference

trajectory is the optimal *finite-time LQR* control

"Shrinking Horizon"

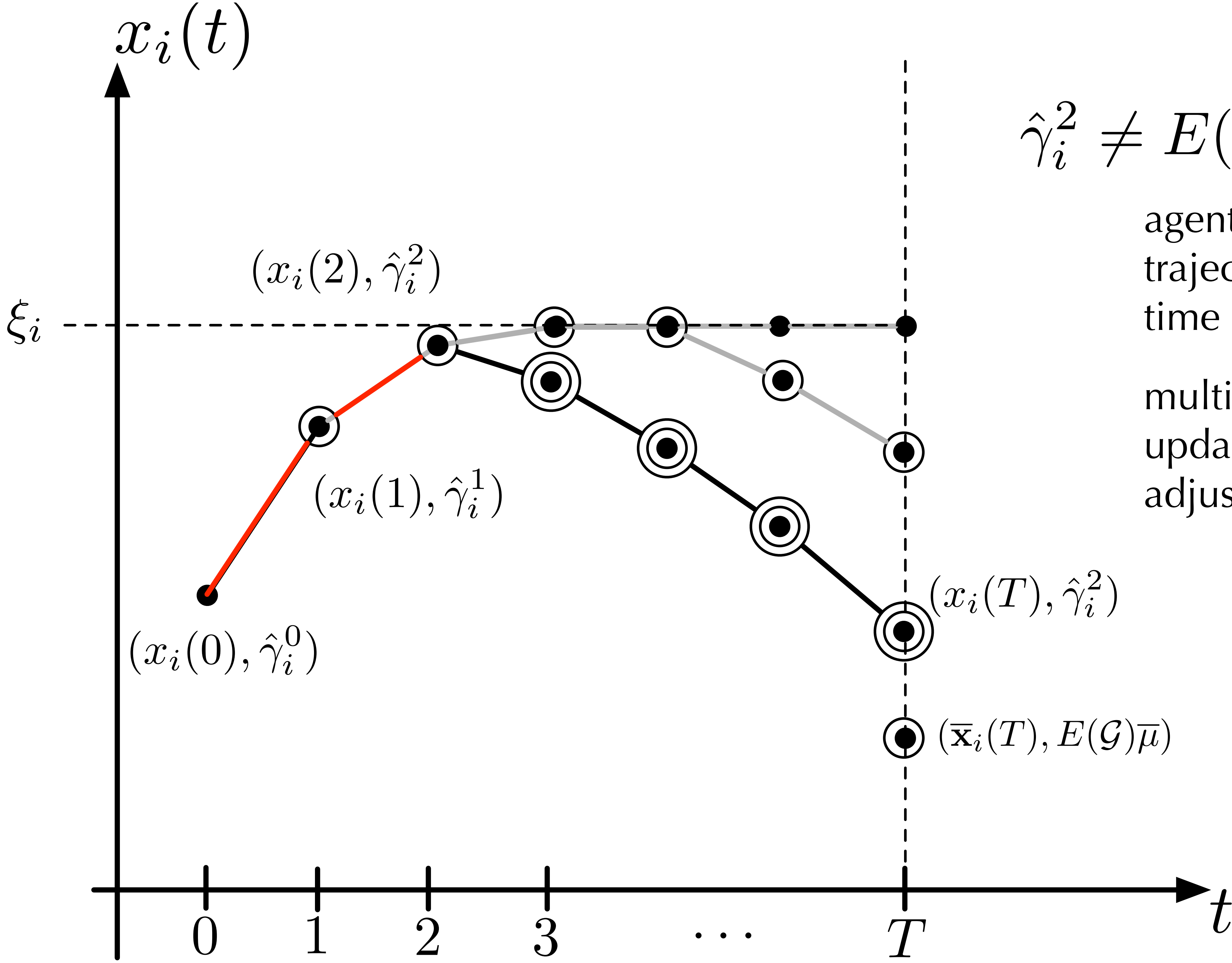


$$\hat{\gamma}_i^1 \neq E(\mathcal{G})\overline{\mu}$$

agent moves along optimal trajectory from previous time step

multiplier has been updated, forcing agent to adjust its planned trajectory

"Shrinking Horizon"



$$\hat{\gamma}_i^2 \neq E(\mathcal{G})\overline{\mu}$$

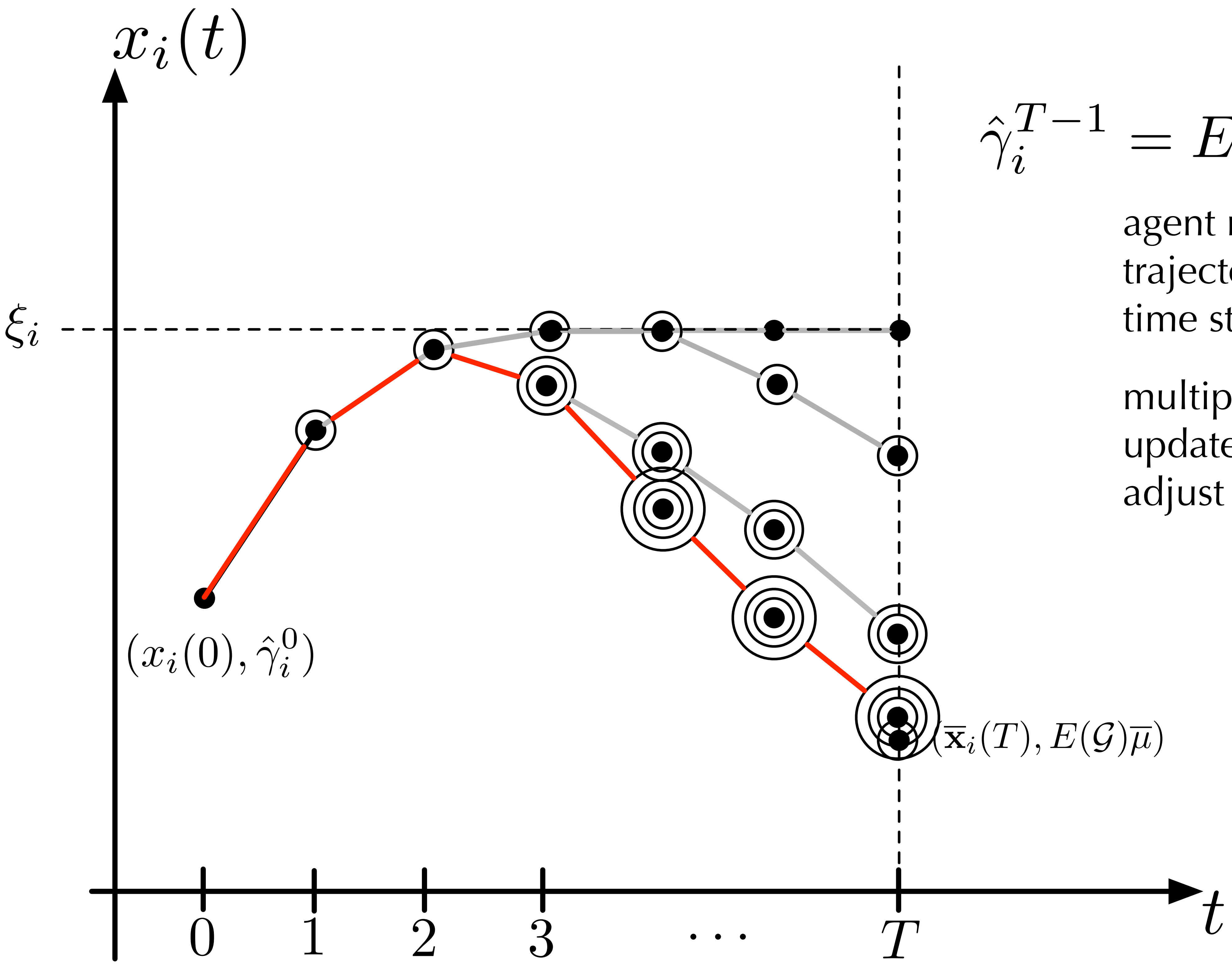
agent moves along optimal trajectory from previous time step

multiplier has been updated, forcing agent to adjust its planned trajectory

“Shrinking Horizon”

• • •

“Shrinking Horizon”



$$\hat{\gamma}_i^{T-1} = E(\mathcal{G})\overline{\mu} \quad ?$$

agent moves along optimal trajectory from previous time step

multiplier has been updated, forcing agent to adjust its planned trajectory

Does it Work?

Algorithm 1: Shrinking Horizon Preference Agreement Algorithm

Data: Initial conditions $x_i(0) = x_{i0}$ and $\mu(0) = \mu_0$; $t = 0$.
begin
  for $t := 0$ **to** *T-1* **do**
    $\gamma^t = E\mu(t), \tilde{T} = T - t$
    Each agent solves the sub-problem $QP_i(t)$:
$$\min_{\hat{\mathbf{x}}_i(t),\hat{\mathbf{u}}_i(t)} J_i(t,T,\hat{\mathbf{x}}_i^t,\, \hat{\mathbf{u}}_i^t) + \gamma_i^t \hat{\mathbf{x}}_i^t(T) \ \text{ s.t. } \ \hat{\mathbf{x}}_i^t = \mathbb{1}_{\tilde{T}} x_i(t) + B_{\tilde{T}} \hat{\mathbf{u}}_i^t$$
    The physical state and multipliers are propagated forward using the solution of $QP_i(t)$:
$$x_i(t+1) = x_i(t) + \hat{\mathbf{u}}_i^t(t),\ i = 1, \ldots, n$$
$$\mu(t+1) = \mu(t) + \alpha(t)E(\mathcal{G})'\hat{\mathbf{x}}^t(T)$$
    where $\alpha(t)$ satisfies some step-size rule.

∗does this generate optimal trajectories?
∗do the multiplier estimates converge to the optimal multipliers?
∗if not, how good is it? what analysis tools are suitable for this problem?

Theorem: The shrinking horizon preference agreement algorithm is equivalent to a time-varying linear dynamical system.

LTV Systems

discrete-time linear dynamical systems

$$x(t+1) = Ax(t) + Bu(t) \qquad x(0) = x_0$$

$$x(t) = A^t x(0) + A^{t-1}Bu(0) + A^{t-2}Bu(1) + \cdots + Bu(t-1)$$

Theorem: The discrete-time linear dynamical system is asymptotically stable if and only if all the eigenvalues of the state matrix satisfy $|\lambda_i(A)| < 1$

LTV Systems

Linear Time-Varying (LTV) dynamical system

$$x(t+1) = A(t)x(t)$$

Definition: The discrete-time autonomous linear time-varying dynamical system is said to be *uniformly decreasing* if

$$\|x(t+1)\| < \|x(t)\|$$

for each time *t* and independent of the initial condition.

a useful notion for *finite-time* problems

Linear System Representation

Algorithm 1: Shrinking Horizon Preference Agreement Algorithm

Data: Initial conditions $x_i(0) = x_{i0}$ and $\mu(0) = \mu_0$; $t = 0$.

begin

  for $t := 0$ **to** *T-1* **do**

    $\gamma^t = E\mu(t), \tilde{T} = T - t$

    Each agent solves the sub-problem $QP_i(t)$:

$$\min_{\hat{\mathbf{x}}_i(t),\hat{\mathbf{u}}_i(t)} J_i(t, T, \hat{\mathbf{x}}_i^t, \hat{\mathbf{u}}_i^t) + \gamma_i^t \hat{\mathbf{x}}_i^t(T) \text{ s.t. } \hat{\mathbf{x}}_i^t = \mathbb{1}_{\tilde{T}} x_i(t) + B_{\tilde{T}} \hat{\mathbf{u}}_i^t$$

    The physical state and multipliers are propagated forward using the solution of $QP_i(t)$:

$$x_i(t+1) = x_i(t) + \hat{\mathbf{u}}_i^t(t),\ i = 1, \ldots, n$$
$$\mu(t+1) = \mu(t) + \alpha(t) E(\mathcal{G})' \hat{\mathbf{x}}^t(T)$$

    where $\alpha(t)$ satisfies some step-size rule.

$$\begin{bmatrix} x(t+1) \\ \mu(t+1) \end{bmatrix} = \begin{bmatrix} I - P(\tilde{T}) & -R^{-1} K(\tilde{T}) E(\mathcal{G}) \\ \alpha(t) E(\mathcal{G})' K(\tilde{T}) & I - \alpha(t) E(\mathcal{G})' Q^{-1} P(\tilde{T}) E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} x(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} P(\tilde{T}) \\ E(\mathcal{G})' \left(I - \alpha(t) K(\tilde{T}) \right) \end{bmatrix} \xi$$

Linear System Representation

$$\begin{bmatrix} x(t+1) \\ \mu(t+1) \end{bmatrix} = \begin{bmatrix} I - P(\tilde{T}) & -R^{-1}K(\tilde{T})E(\mathcal{G}) \\ \alpha(t)E(\mathcal{G})'K(\tilde{T}) & I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} x(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} P(\tilde{T}) \\ E(\mathcal{G})'\left(I - \alpha(t)K(\tilde{T})\right) \end{bmatrix} \xi$$

<u>proof:</u>

not here...too messy!

but look here...

- analytic solutions of QP
- Sherman-Morrison-Woodbury-Schur formula
- derivation of recursions
- Kalman Filter

Linear System Representation

$$\begin{bmatrix} x(t+1) \\ \mu(t+1) \end{bmatrix} = \begin{bmatrix} I - P(\tilde{T}) & -R^{-1}K(\tilde{T})E(\mathcal{G}) \\ \alpha(t)E(\mathcal{G})'K(\tilde{T}) & I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} x(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} P(\tilde{T}) \\ E(\mathcal{G})'\left(I - \alpha(t)K(\tilde{T})\right) \end{bmatrix} \xi$$

$$P_i(\tilde{T}+1) = \frac{1+\frac{r_i}{q_i}P_i(\tilde{T})}{1+\frac{r_i}{q_i}+\frac{r_i}{q_i}P_i(\tilde{T})}, \qquad P_i(1) = \frac{q_i}{r_i+q_i}$$

$$K_i(\tilde{T}+1) = \frac{r_i}{q_i}\frac{K_i(\tilde{T})}{1+\frac{r_i}{q_i}+\frac{r_i}{q_i}P_i(\tilde{T})}, \qquad K_i(1) = \frac{r_i}{r_i+q_i}.$$

$P_i(\tilde{T})$ *is the finite-time LQR gain!*

∗can be computed off-line
∗*independent* of graph, number of agents, step-size, etc...

Linear System Representation

$$\begin{bmatrix} x(t+1) \\ \mu(t+1) \end{bmatrix} = \begin{bmatrix} I - P(\tilde{T}) & -R^{-1}K(\tilde{T})E(\mathcal{G}) \\ \alpha(t)E(\mathcal{G})'K(\tilde{T}) & I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} x(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} P(\tilde{T}) \\ E(\mathcal{G})'\left(I - \alpha(t)K(\tilde{T})\right) \end{bmatrix} \xi$$

$$I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G})$$

acts like a weighted consensus algorithm!*

LQR gains also used in the negotiation process

* the *consensus protocol* is a distributed averaging scheme $\dot{x} = -L(\mathcal{G})x$

Linear System Representation

$$\begin{bmatrix} x(t+1) \\ \mu(t+1) \end{bmatrix} = \begin{bmatrix} I - P(\tilde{T}) & -R^{-1}K(\tilde{T})E(\mathcal{G}) \\ \alpha(t)E(\mathcal{G})'K(\tilde{T}) & I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} x(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} P(\tilde{T}) \\ E(\mathcal{G})'\left(I - \alpha(t)K(\tilde{T})\right) \end{bmatrix} \xi$$

$\alpha(t)$ is the *only* design parameter

choice of step-size now akin to a *stabilization* problem

linear systems theory is the correct tool to analyze performance of SHPA

Performance of SHPA Algorithm

$$\begin{bmatrix} x(t+1) \\ \mu(t+1) \end{bmatrix} = \begin{bmatrix} I - P(\tilde{T}) & -R^{-1}K(\tilde{T})E(\mathcal{G}) \\ \alpha(t)E(\mathcal{G})'K(\tilde{T}) & I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} x(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} P(\tilde{T}) \\ E(\mathcal{G})'\left(I - \alpha(t)K(\tilde{T})\right) \end{bmatrix} \xi$$

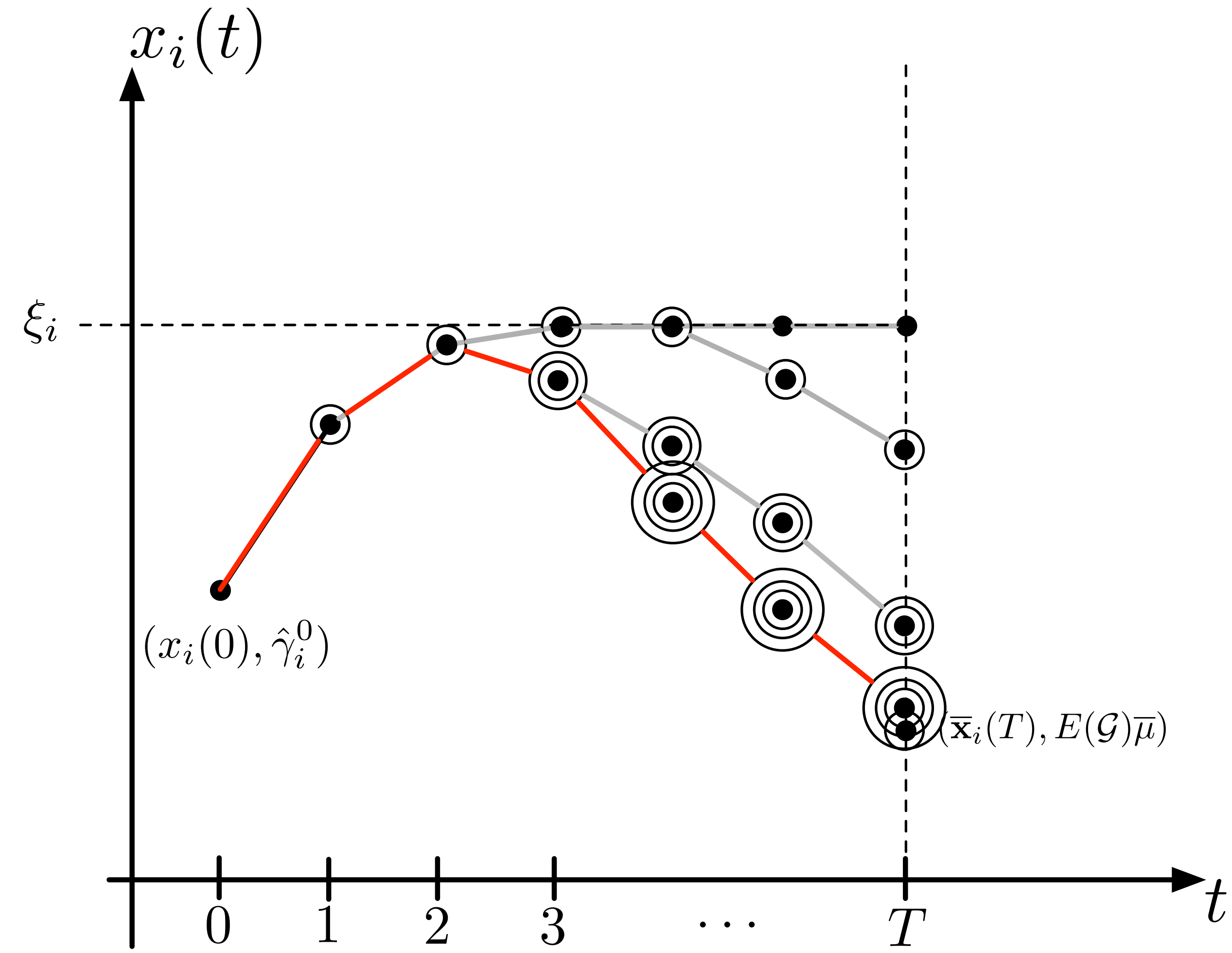
Two important error signals

∗multiplier error

$$\epsilon(t) = \mu(t) - \overline{\mu}^{\,t}$$

∗predicted disagreement

$$\mathbf{e}(t) = E(\mathcal{G})'\hat{\mathbf{x}}^t(T)$$



Performance of SHPA Algorithm

Corollary: The optimal multipliers associated with the problem *OCP(t,T,x(t))* evolves according to a time-varying linear dynamical system

$$\overline{\mu}^{\,t} = \left(E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G})\right)^{-1} E(\mathcal{G})' \left[K(\tilde{T})(x(t)-\xi)+\xi\right]$$

want this...

$$\lim_{t\to T} \|\mu(t)-\overline{\mu}^{t}\| \to 0$$

analyze multiplier error dynamics

$$\epsilon(t) = \mu(t) - \overline{\mu}^{\,t}$$

Performance of SHPA Algorithm

Theorem: The multiplier error dynamics evolves according to a time-varying linear dynamical system.

$$\epsilon(t+1) = \left((E(\mathcal{G})'Q^{-1}P(\tilde{T}-1)E(\mathcal{G}))^{-1} - \alpha(t)I \right) E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G})\epsilon(t)$$

Lemma: There exists a step-size rule such that the multiplier error dynamics is uniformly decreasing if and only if the following LMI condition is feasible

$$-I \leq L_t^{1/2} L_{t+1}^{-1} L_t^{1/2} - \alpha(t) L_t \leq I$$

$$L_t = E(\mathcal{G})'Q^{-1}P(\tilde{T})E(\mathcal{G})$$

Performance of SHPA Algorithm

$$-I \leq L_t^{1/2} L_{t+1}^{-1} L_t^{1/2} - \alpha(t) L_t \leq I$$

$$L_t = E(\mathcal{G})' Q^{-1} P(\tilde{T}) E(\mathcal{G})$$

insight gained by considering a simplified problem set-up

$$Q = qI \quad R = rI$$

all agents have the same state and control weight (but different preferences)

Corollary: There exists a step-size rule such that the multiplier error dynamics is uniformly decreasing if and only if

$$\frac{\lambda_{\max}(E(\mathcal{G})'E(\mathcal{G}))}{\lambda_{\min}(E(\mathcal{G})'E(\mathcal{G}))} < 3 + 2\left(\left(\frac{q}{r}\right)^2 + 3\frac{q}{r}\right)$$

Performance of SHPA Algorithm

Theorem: The predicted disagreement evolves according to a time-varying linear dynamical system.

$$\mathbf{e}(t+1) = \left(I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T}-1)E(\mathcal{G})\right)\mathbf{e}(t)$$

want this...

$$\lim_{t\to T} \|\mathbf{e}(t)\| \to 0$$

Performance of SHPA Algorithm

$$\mathbf{e}(t+1) = \Big(I - \alpha(t)E(\mathcal{G})'Q^{-1}P(\tilde{T}-1)E(\mathcal{G})\Big)\,\mathbf{e}(t)$$

Corollary: The predicted disagreement is uniformly decreasing if and only if

$$0 < \alpha(t) < 2\lambda_{\max}^{-1}(E(\mathcal{G})'Q^{-1}P(\tilde{T}-1)E(\mathcal{G}))$$

$Q = qI \quad R = rI$

Corollary: The predicted disagreement is uniformly decreasing if and only if

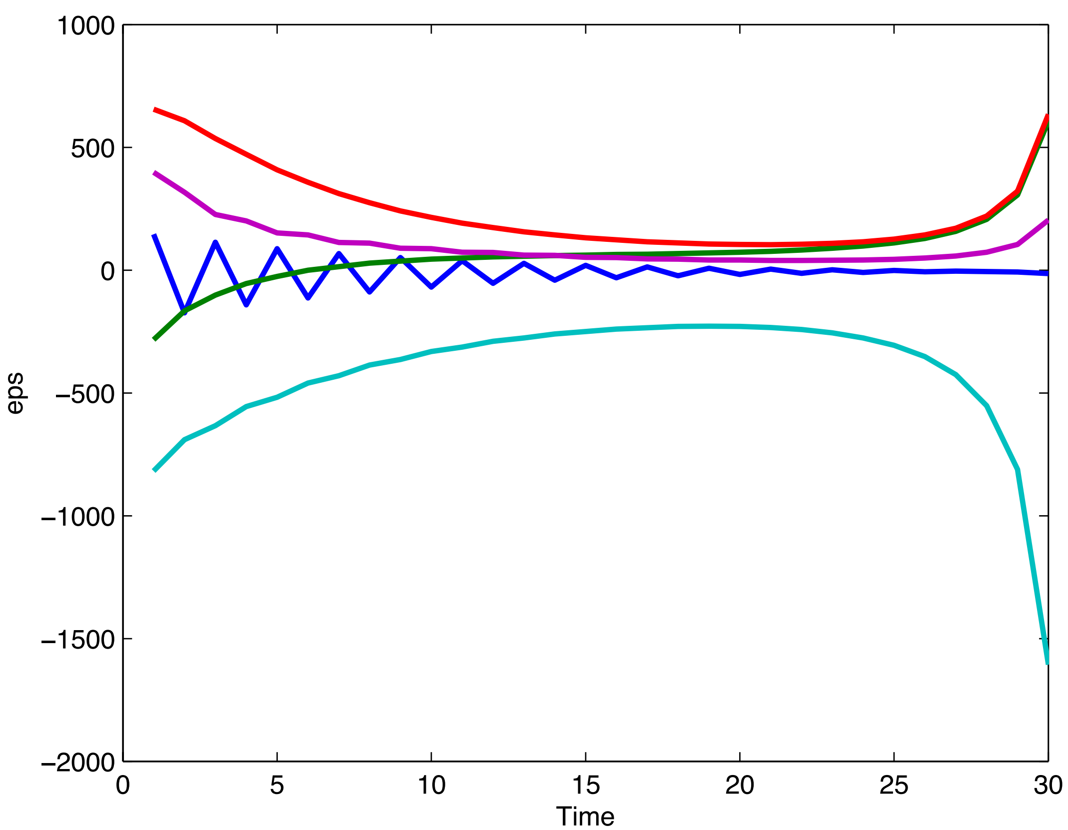
$$0 < \alpha(t) < 2\frac{q}{P(T-1)\lambda_{max}(E(\mathcal{G})'E(\mathcal{G}))}$$

Performance of SHPA Algorithm

an interesting observation...

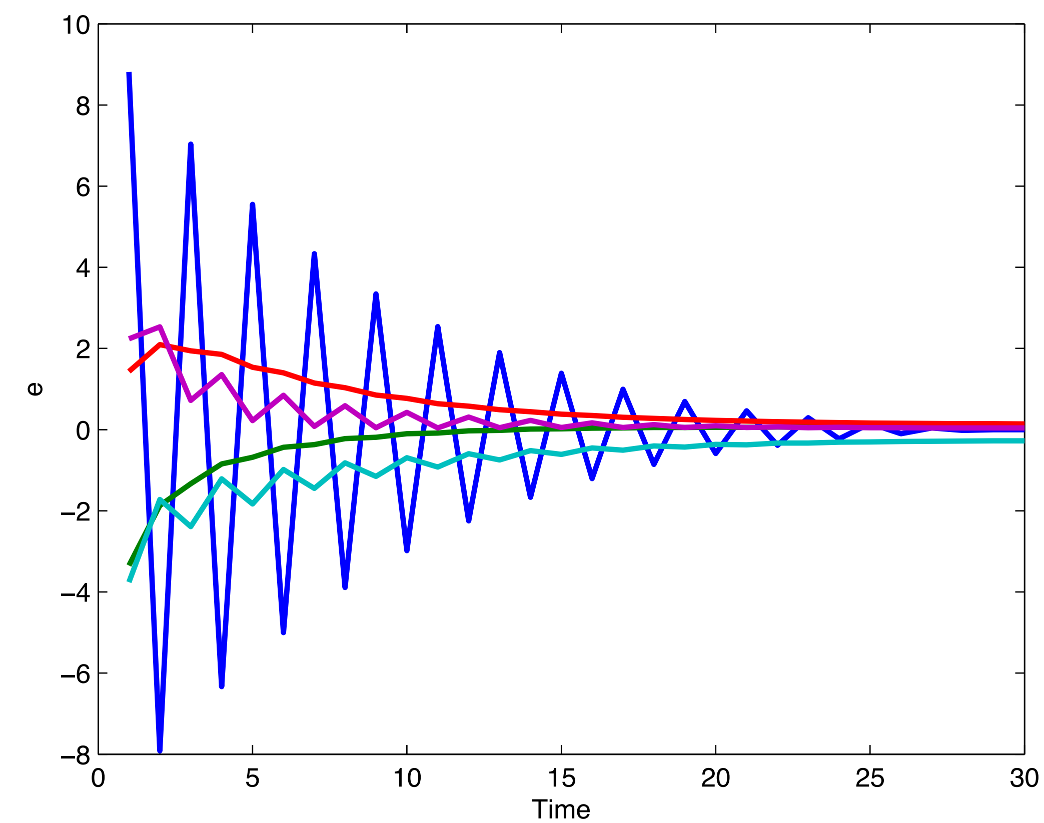
$\epsilon(t)$

$$-I \leq L_t^{1/2} L_{t+1}^{-1} L_t^{1/2} - \alpha(t) L_t \leq I$$



$\mathbf{e}(t)$

$$0 < \alpha(t) < 2\lambda_{\max}^{-1}(E(\mathcal{G})' Q^{-1} P(\tilde{T} - 1) E(\mathcal{G}))$$

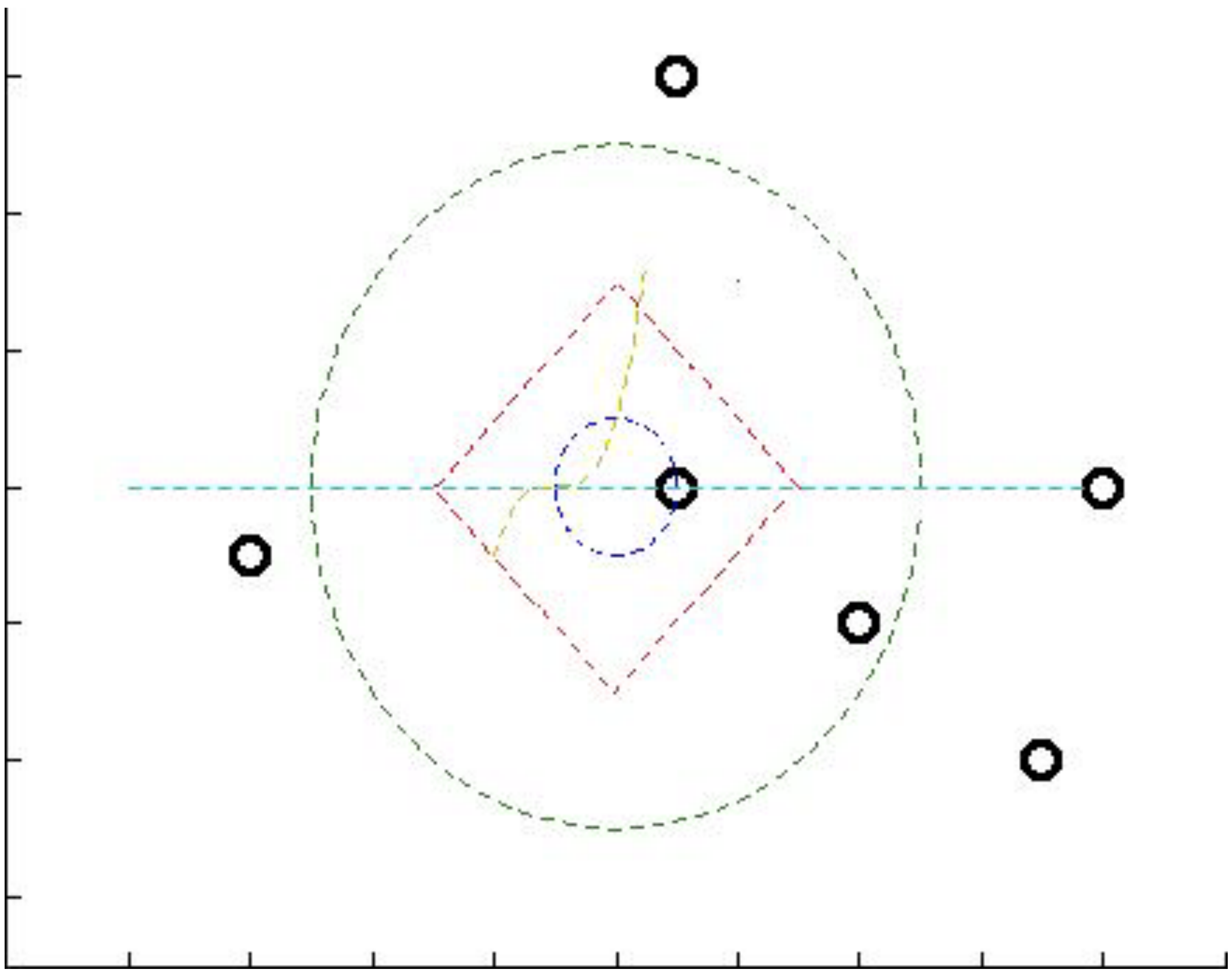


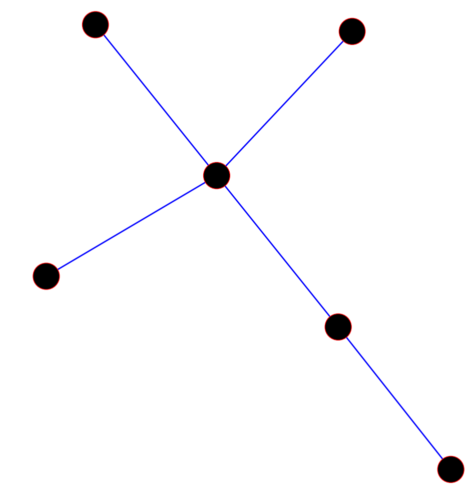
in this case, disagreement can be made arbitrarily small in finite time!

step-sizes exist that guarantee error is *not* uniformly decreases, but not the multiplier error

Simulation Examples

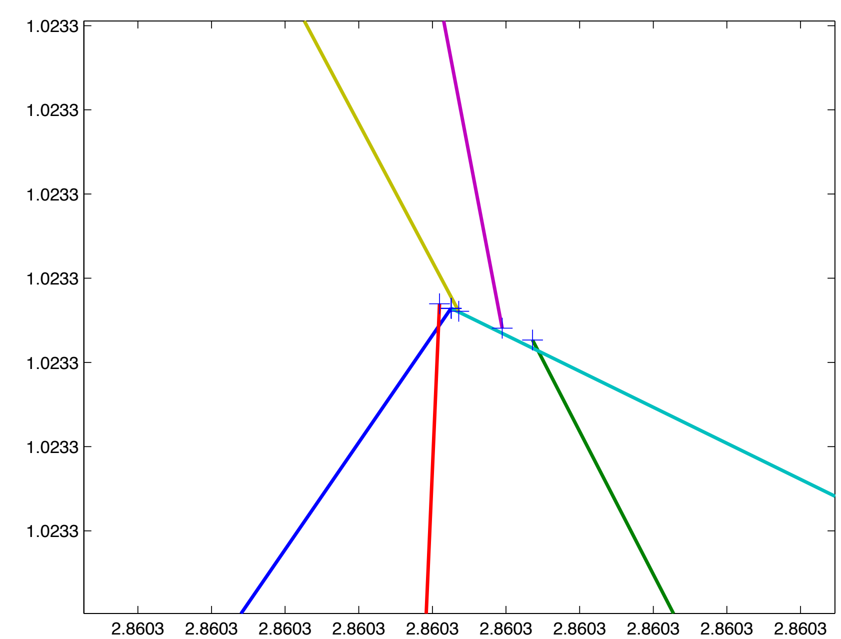
SHPA with time-varying preference





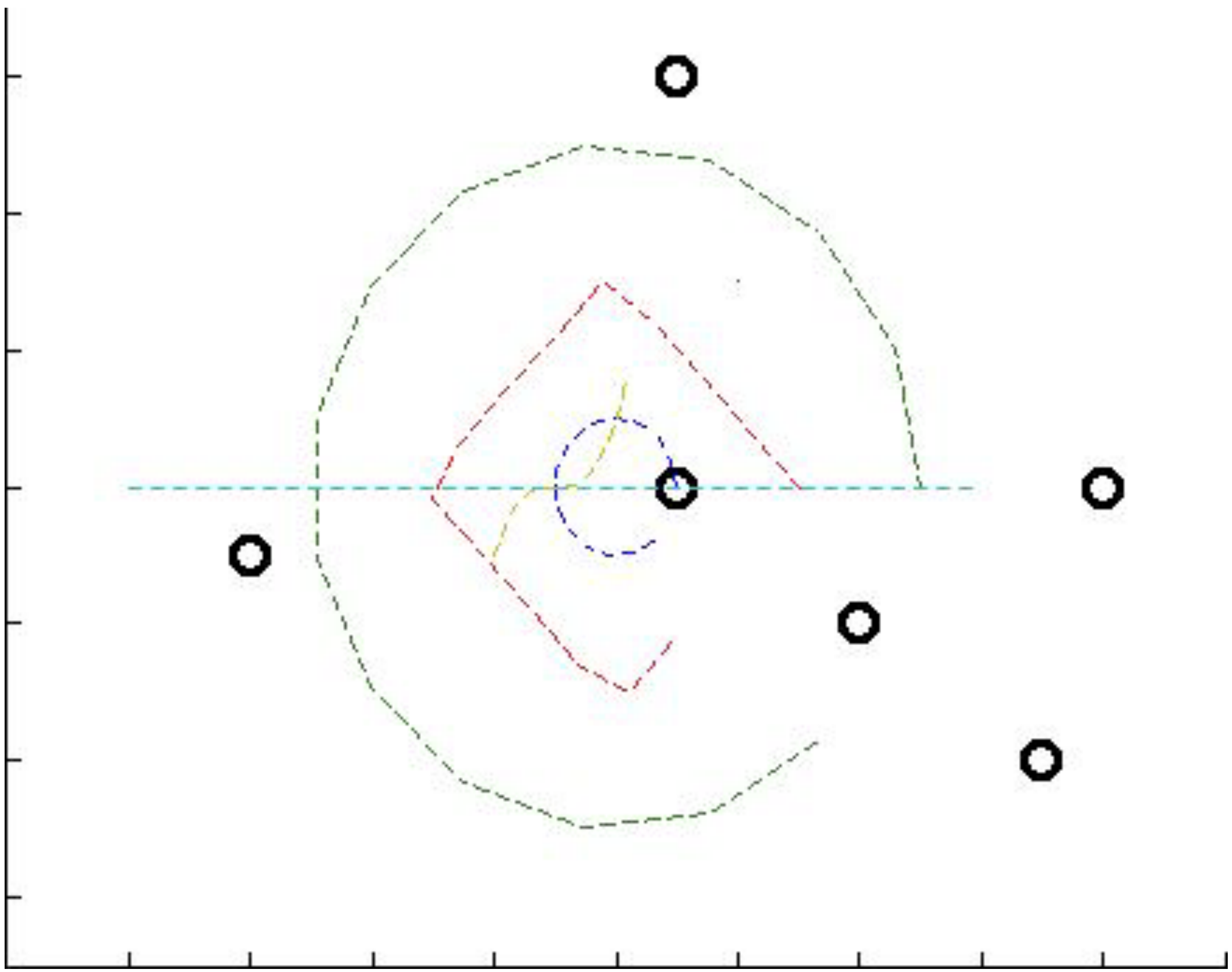
$$T = 150$$

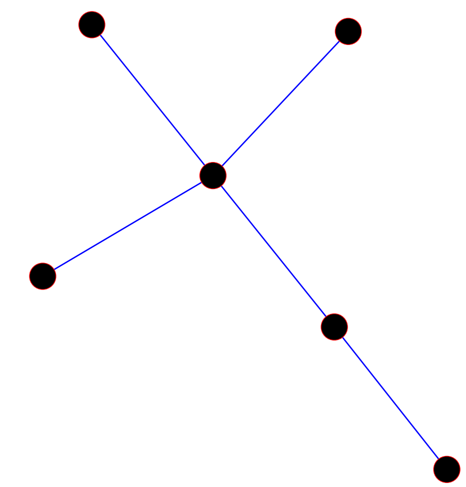
$$\alpha(t) = \alpha = 10$$



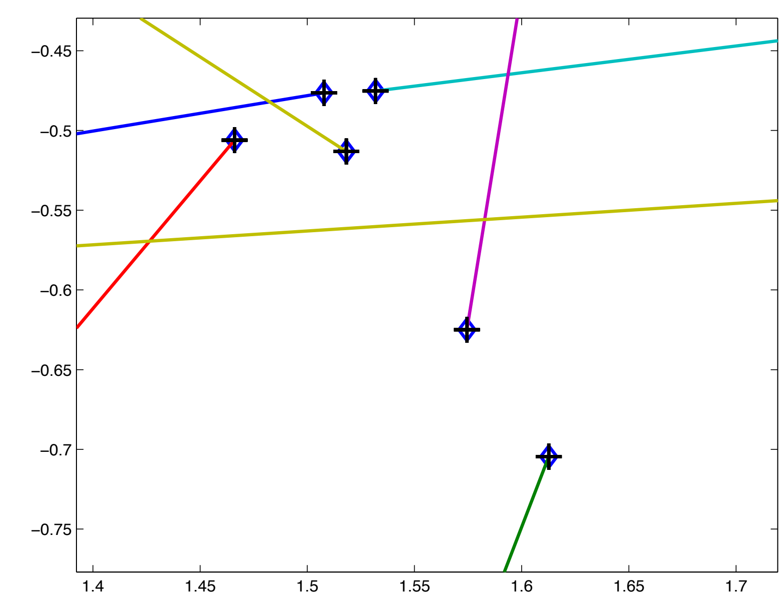
Simulation Examples

SHPA with time-varying preference





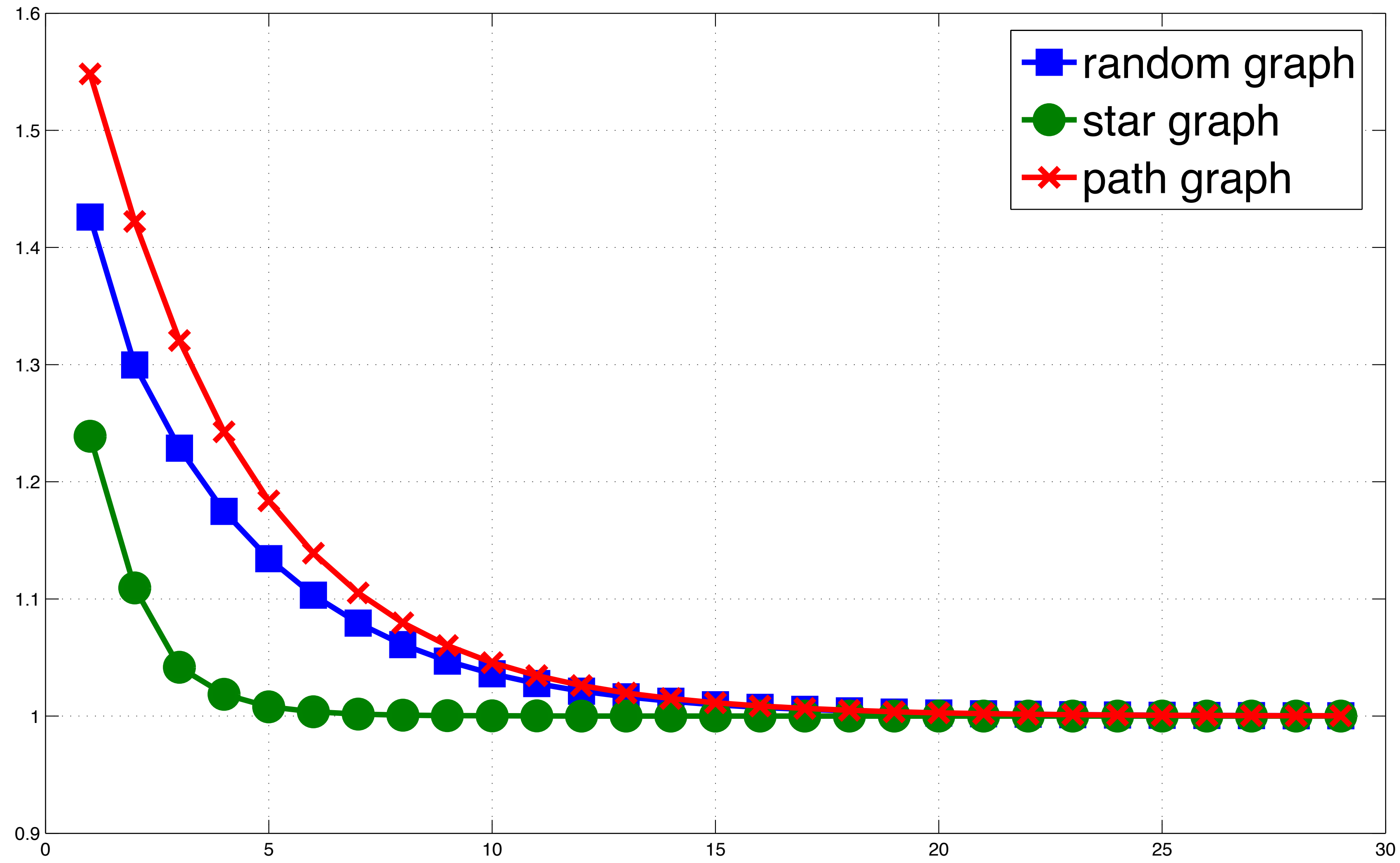
$$T = 15$$
$$\alpha(t) = \alpha = 10$$



Simulation Examples

Optimality Gap

$$\Delta = \frac{\mathcal{L}(x, u, \overline{\mu})}{\mathcal{L}(\overline{\mathbf{x}}, \overline{\mathbf{u}}, \overline{\mu})}$$



Switching Communication

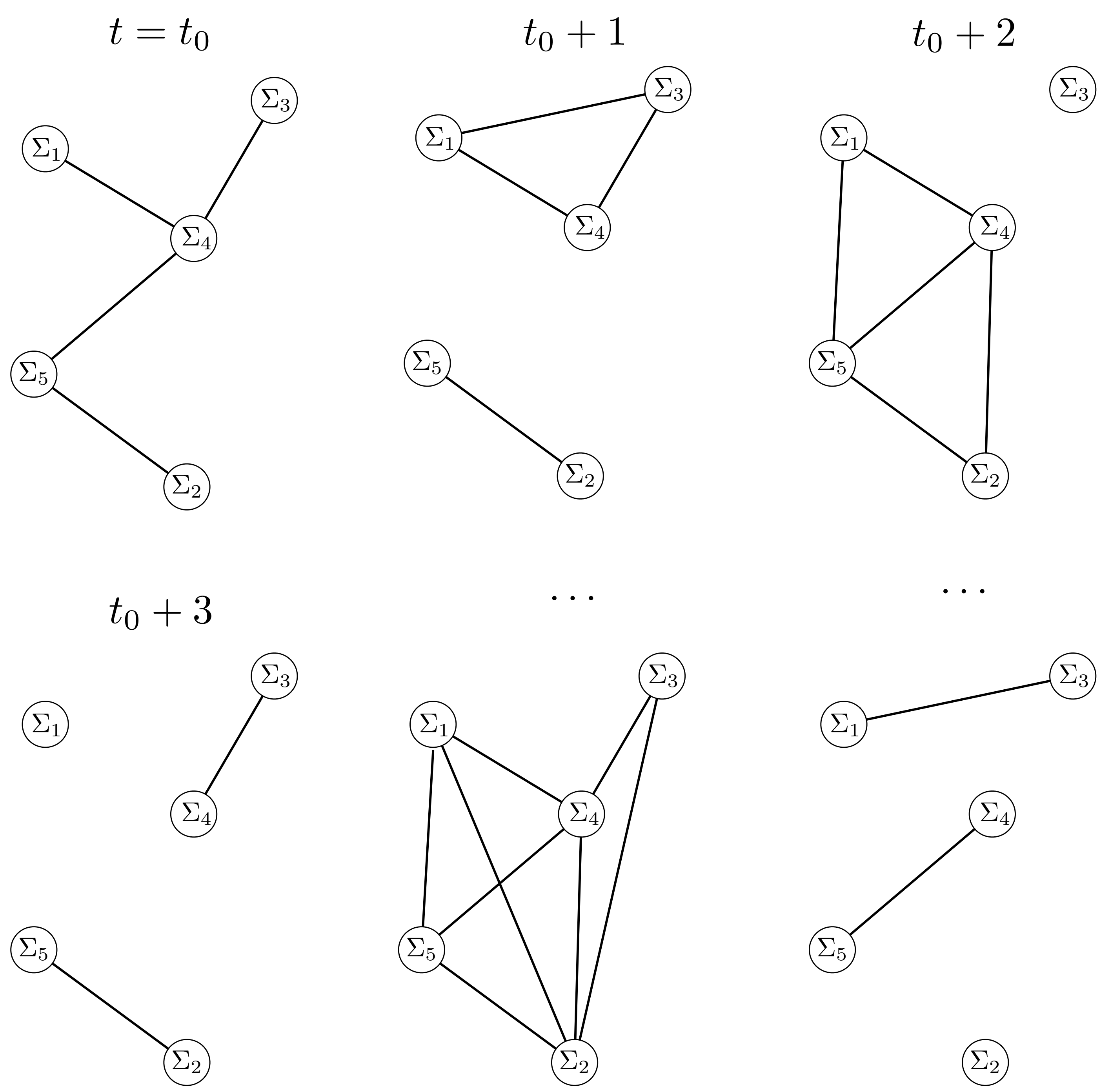
agents can communicate over a network

*switching communication

$$\sigma \;:\; \{0, 1, \ldots\} \rightarrow \mathcal{G}$$

switching signal

$$\mathcal{G}_{\sigma(t)} = (\mathcal{V}, \mathcal{E}_{\sigma(k)})$$



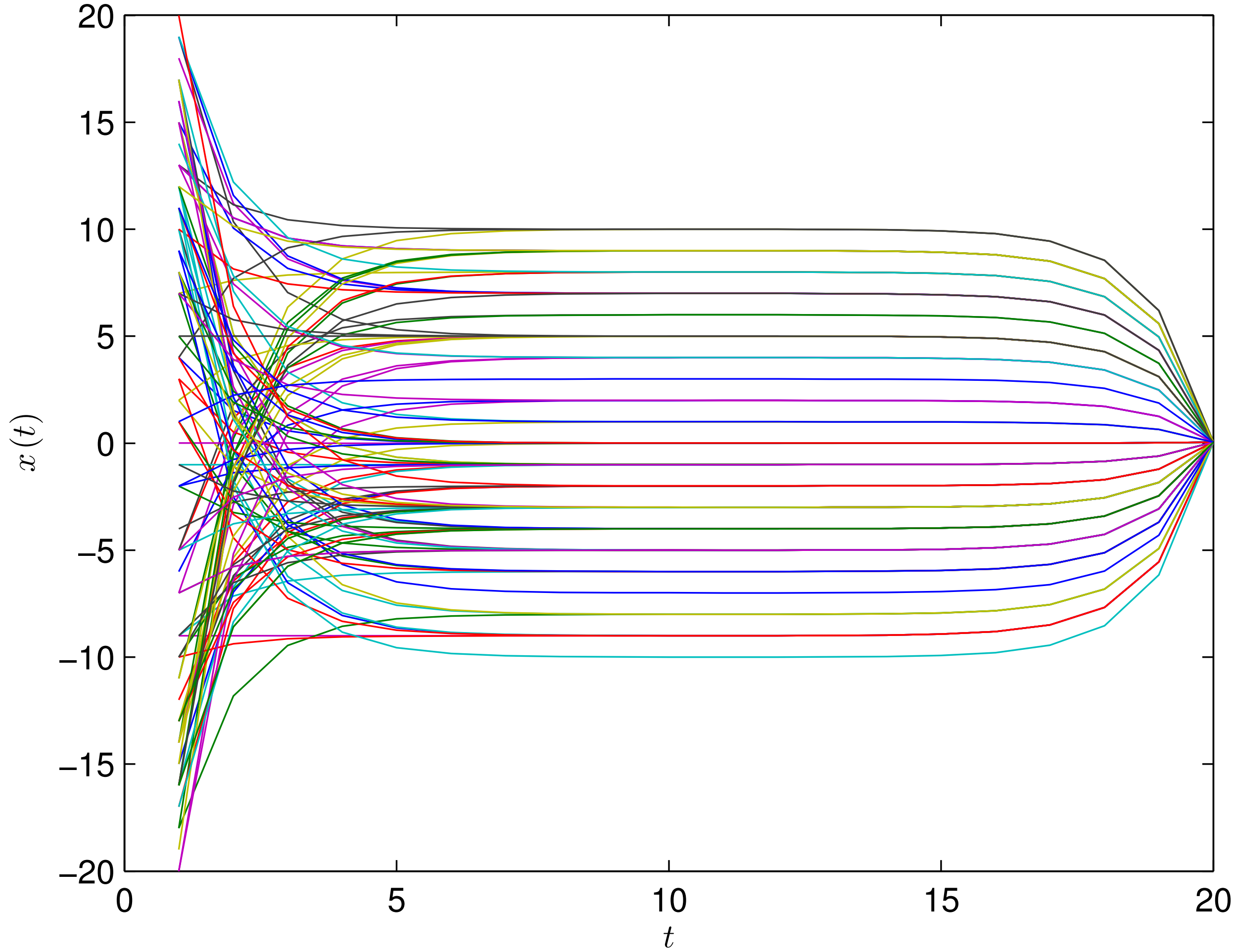
Switching Communication

"similar" analytic results

- uniformly jointly connected graphs

interesting results

- simulations using a random graph model to generate switching signal



Edge probability: $p = 0.1$

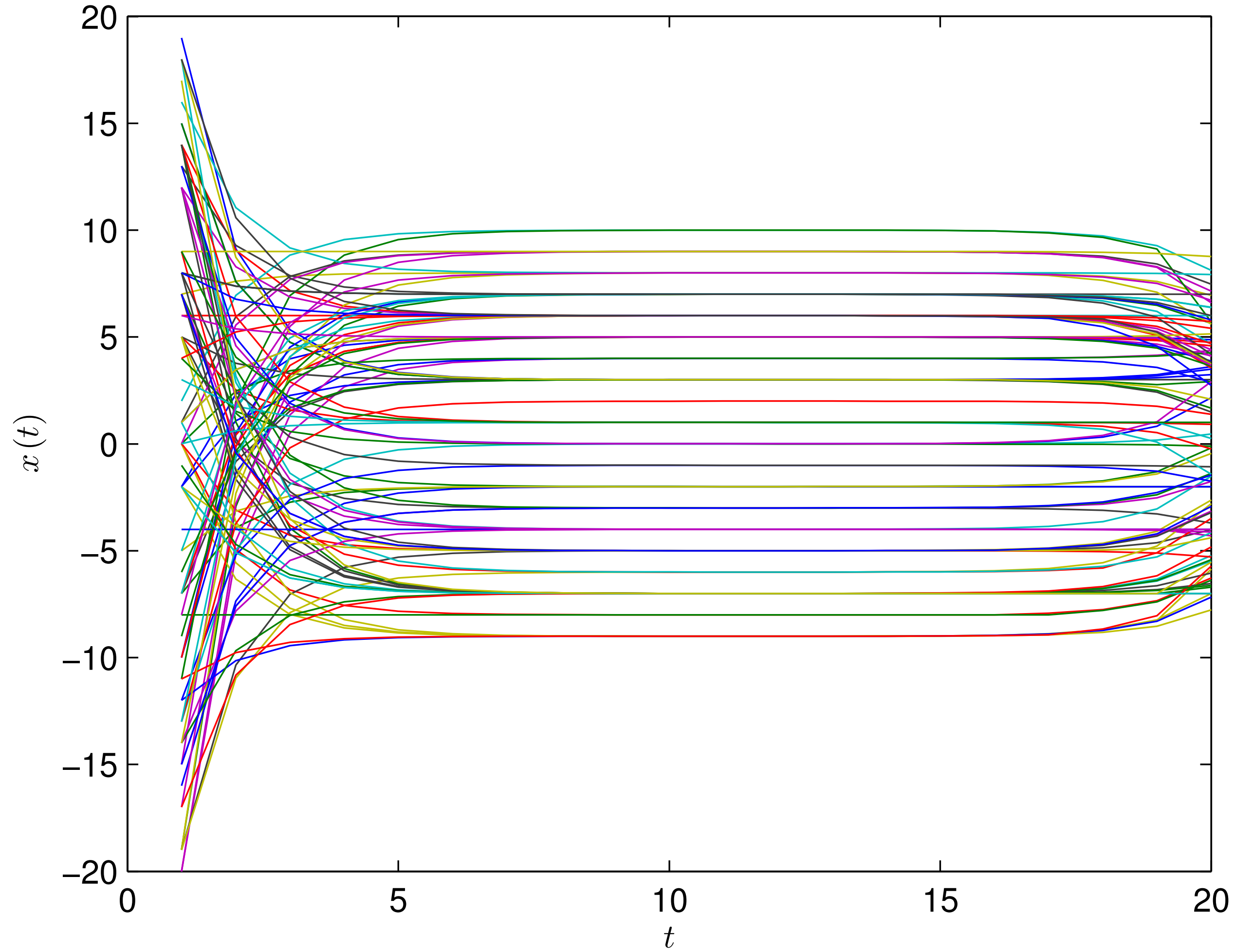
Switching Communication

"similar" analytic results

- uniformly jointly connected graphs

interesting results

- simulations using a random graph model to generate switching signal



Edge probability: $p = 0.01$ (not enough communication)

Switching Communication

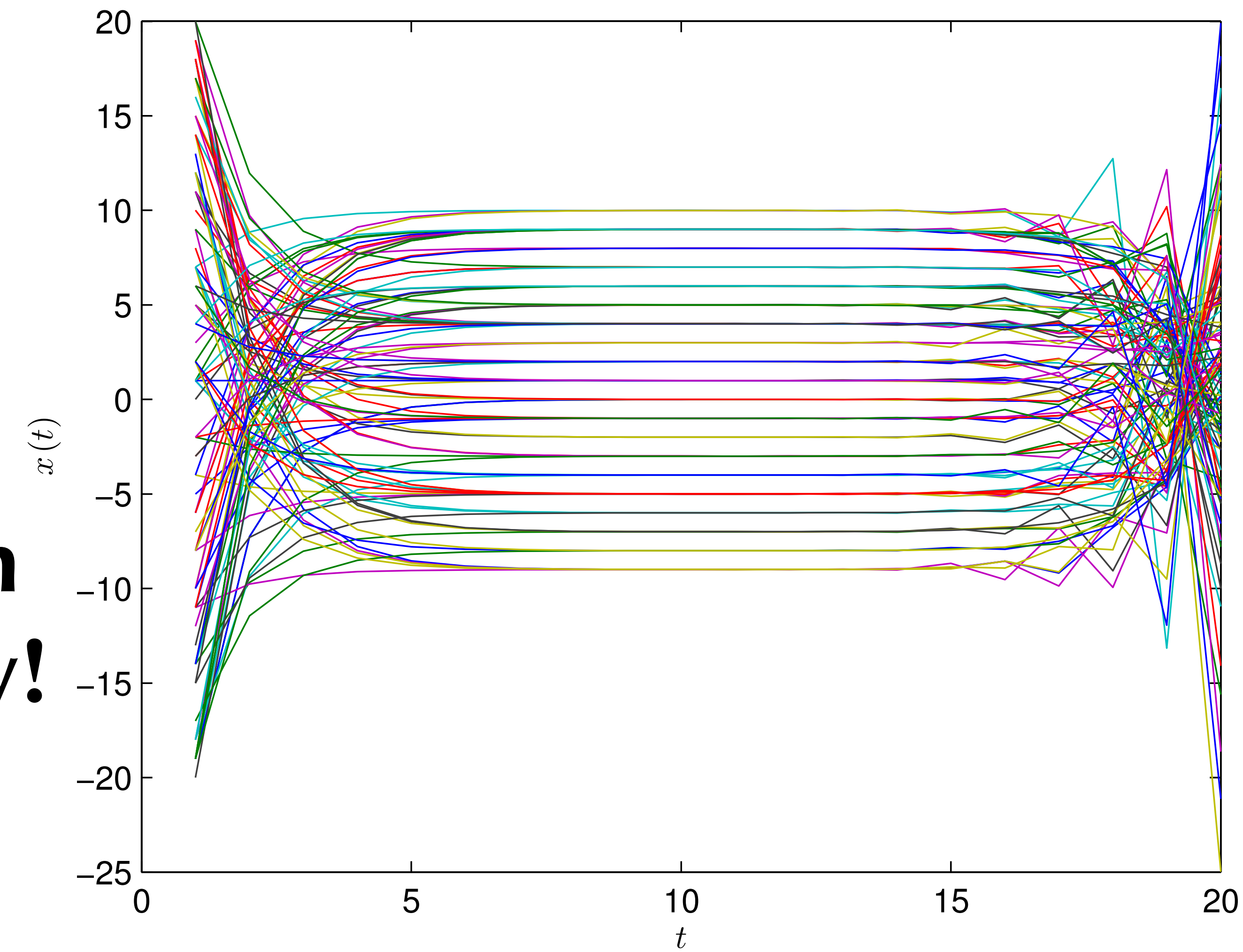
"similar" analytic results

- uniformly jointly connected graphs

interesting results

- simulations using a random graph model to generate switching signal

more communication can lead to instability!



Edge probability: $p = 0.15$

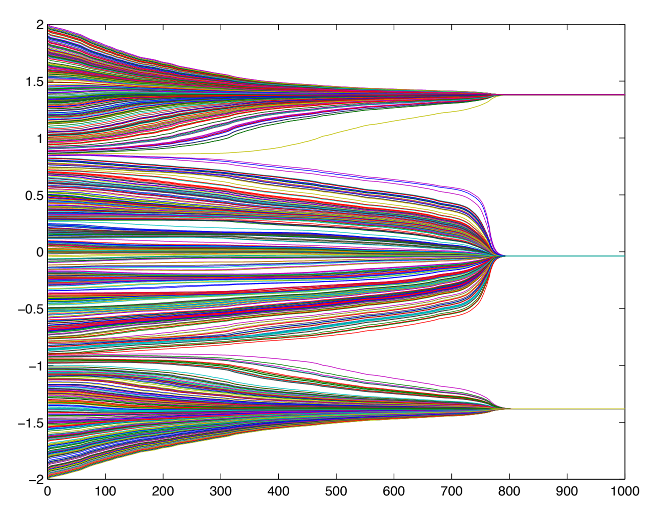
Concluding Remarks

SHPA algorithm is an attempt to understand the complexities of *real-time distributed optimization problems*

- ∗interplay between dynamic systems and distributed optimization
- ∗step-size, graph structure, preferences
- ∗simple set-up, non-trivial results

limitless extensions...

- ∗state-dependent graphs, random graphs
- ∗more sophisticated dynamics
- ∗saddle-point problems and multi-agent systems
- ∗and more...



Acknowledgements

Prof. Dr. -Ing. Frank Allgöwer

Mathias Bürger

References:

[1] D. Zelazo, M. Bürger, and F. Allgöwer, *"A Finite-Time Dual Method for Negotiation between Dynamical Systems,"* SIAM Journal of Control and Optimization, vol. 51, no. 1, pp. 172–194, Jan. 2013.

[2] D. Zelazo, M. Bürger, and F. Allgöwer, *"Dynamic Negotiation Under Switching Communication,"* in Mathematical System Theory -- Festschrift in Honor of Uwe Helmke on the Occasion of his Sixtieth Birthday, K. Hüper and J. Trumpf, Eds. CreateSpace, 2013, pp. 479–500.

Questions?