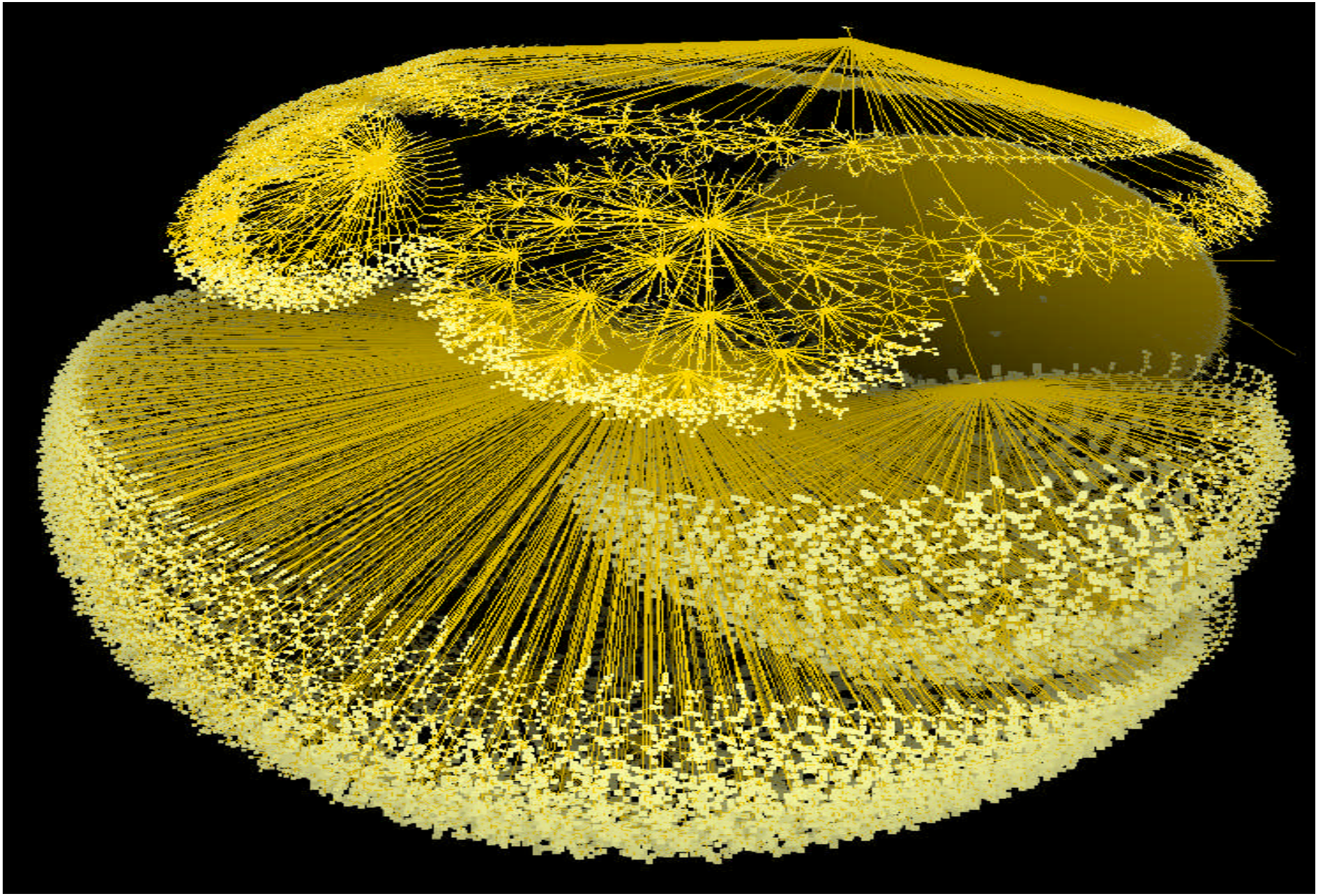
Resilient Control in
Cooperative and Adversarial Multi-agent Networks



F.L. Lewis

National Academy of Inventors

Moncrief-O'Donnell Chair, UTA Research Institute (UTARI)
The University of Texas at Arlington, USA

Reinforcement Learning for Resilient Control in Cooperative and Adversarial Multi-agent Networks: CPS Applications in Microgrid and Human-Robot Interactions

Supported by :
ONR – Tony Seman, Lynn Petersen, Marc Steinberg
US TARDEC – Dariusz Mikulski
AFOSR Europe
NSF

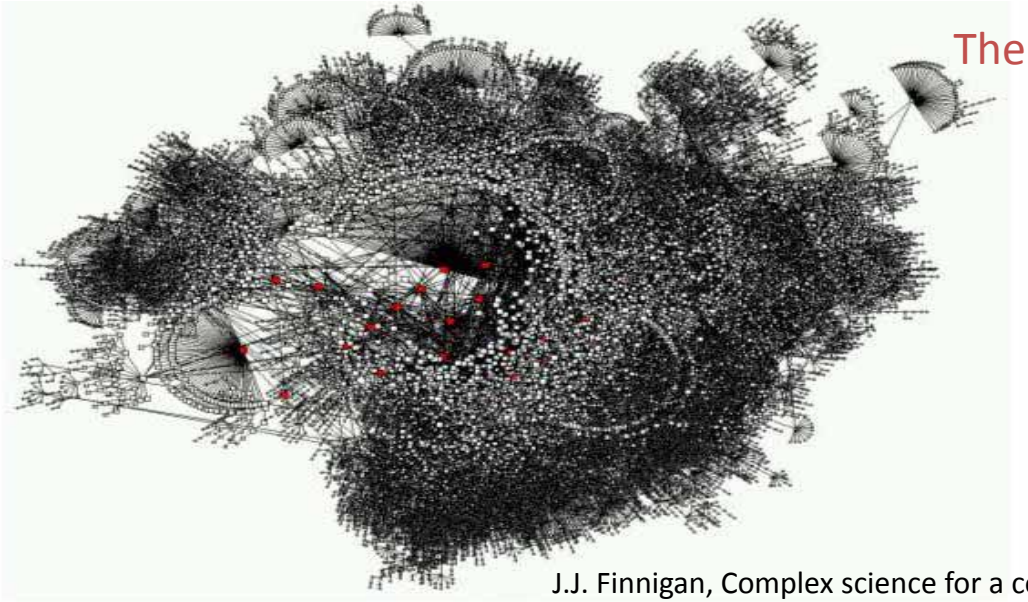
Talk available online at
http://www.UTA.edu/UTARI/acs

Invited by Frank Ferrese

Structure of Natural and Manmade Systems

Peer-to-Peer Relationships
in networked systems
Distributed Decision & Control
Local nature of Physical Laws

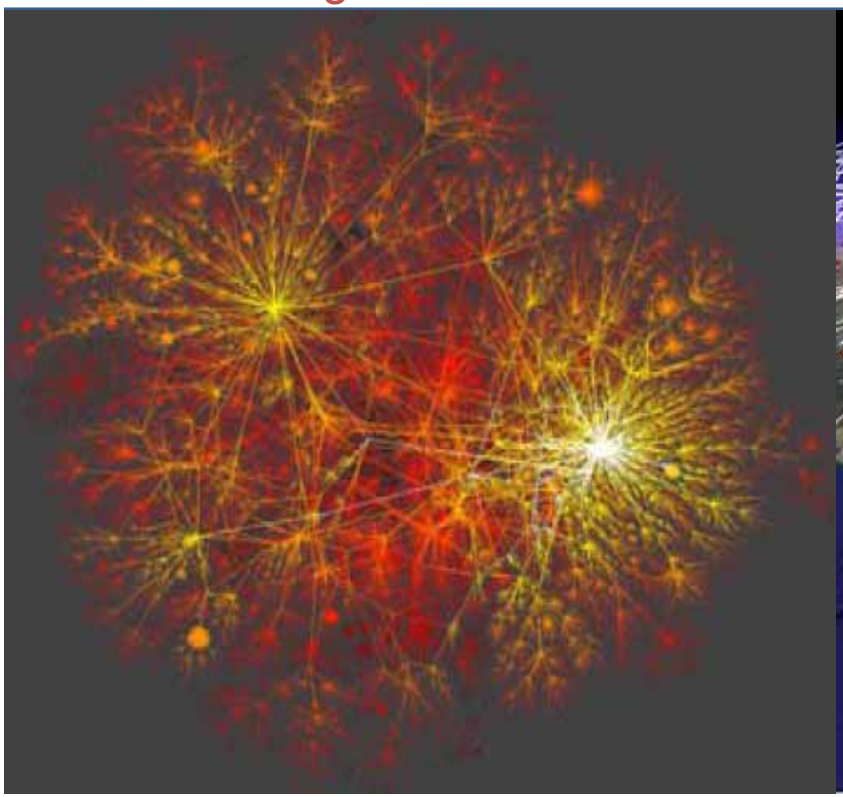
The Internet

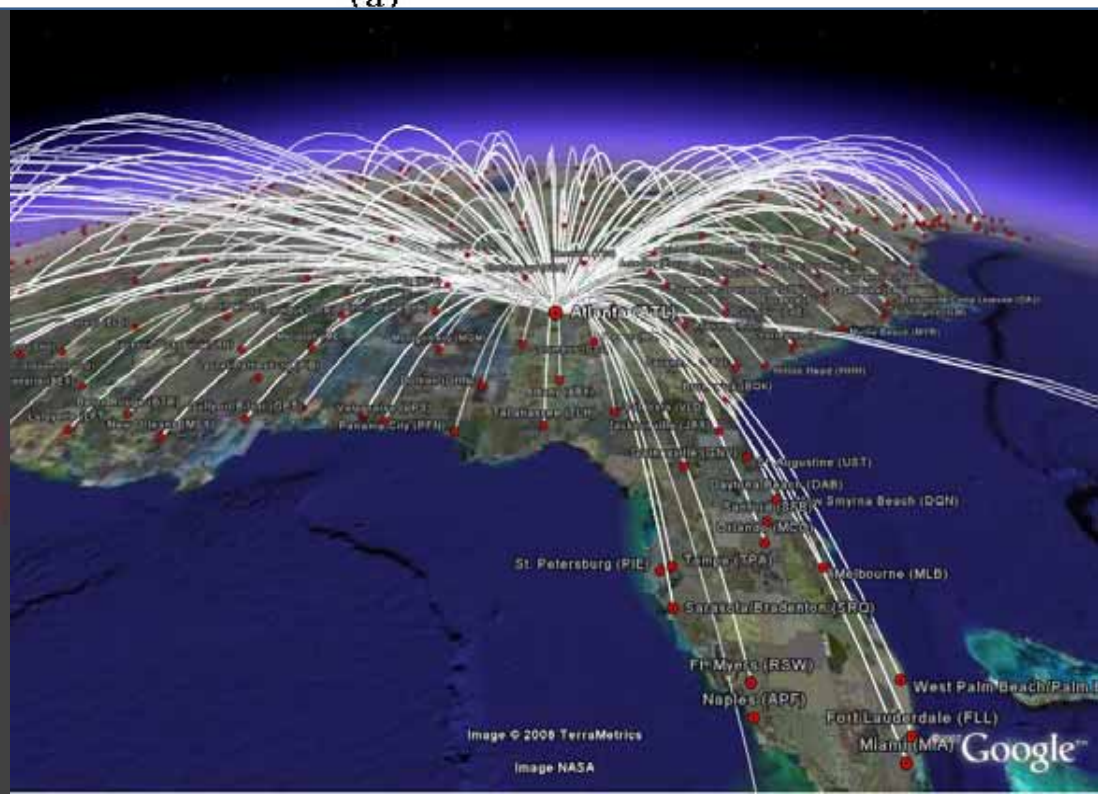


J.J. Finnigan, Complex science for a complex world

(a)

Clusters of galaxies





Distributed Adaptive Control for Multi-Agent Systems

Reynolds, Computer Graphics 1987

Flocking

Nature and Biological Systems are Resilient !

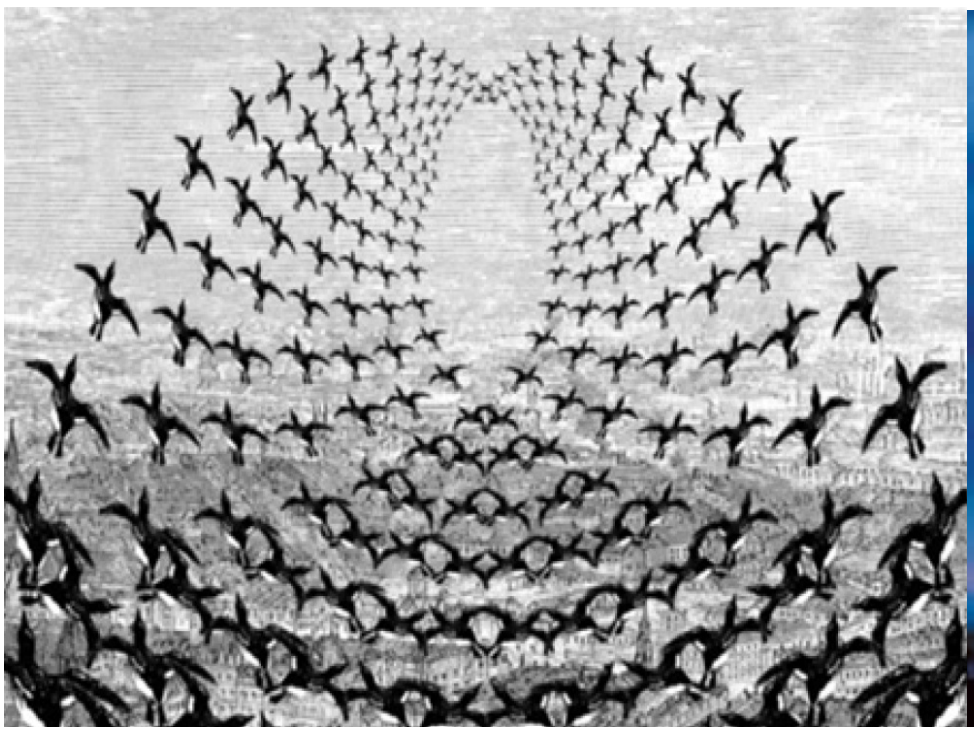
Reynolds' Rules:

Alignment : align headings

$$\dot{\theta}_i = \sum_{j \in N_i} a_{ij}(\theta_j - \theta_i)$$

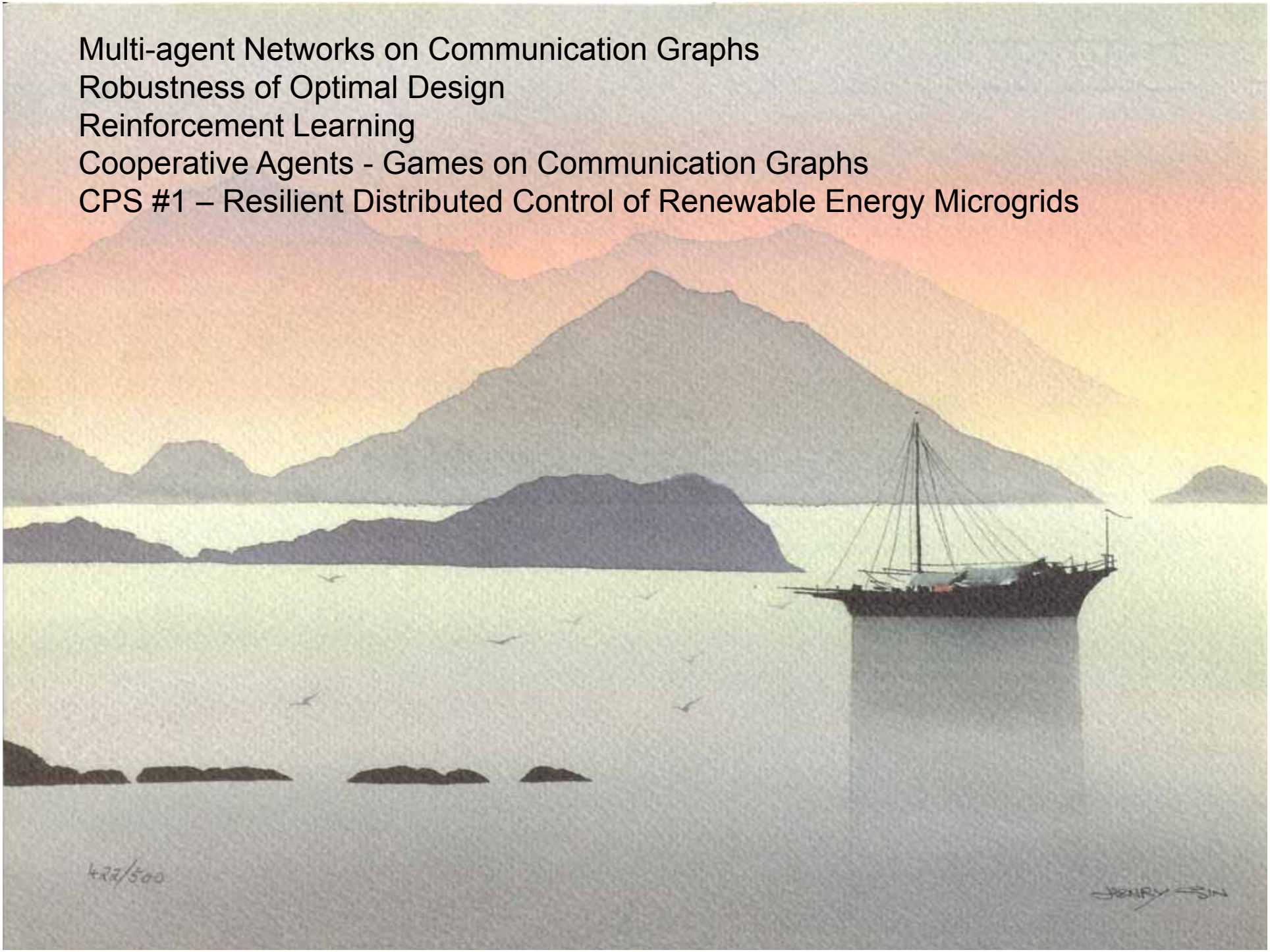
Cohesion : steer towards average position of neighbors- towards c.g.

Separation : steer to maintain separation from neighbors



Resilient Control in Cooperative Multi-agent Networks
Modeling and Control of Adversarial Networked Teams
CPS Applications-
- Distributed Control of Renewable Energy Microgrids
- Shared Learning in Human-Robot Interactions

Multi-agent Networks on Communication Graphs
Robustness of Optimal Design
Reinforcement Learning
Cooperative Agents - Games on Communication Graphs
CPS #1 – Resilient Distributed Control of Renewable Energy Microgrids



F.L. Lewis, H. Zhang, A. Das, K. Hengster-Movric, *Cooperative Control of Multi-Agent Systems: Optimal Design and Adaptive Control*, Springer-Verlag, 2014

Key Point

Lyapunov Functions and Performance Indices
Must depend on graph topology

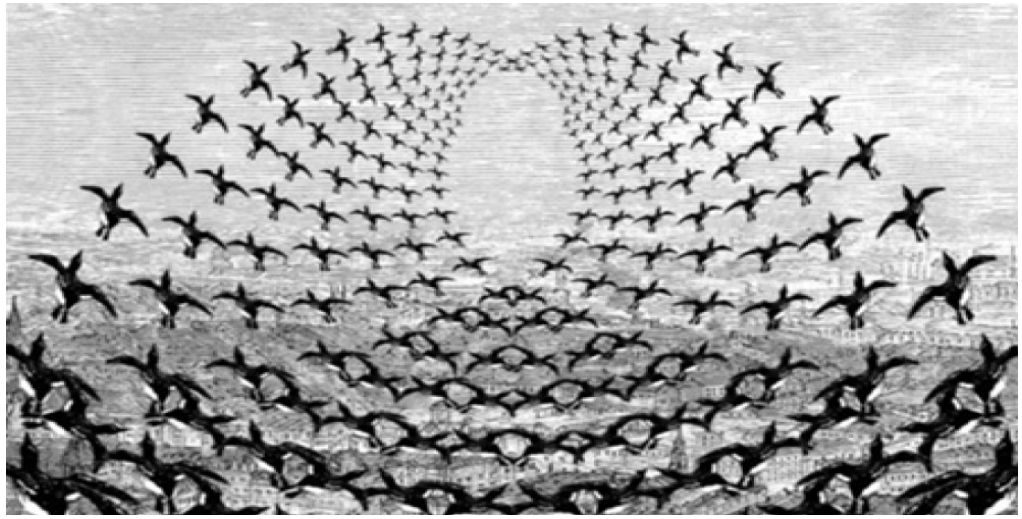
H. Zhang, F.L. Lewis, and Z. Qu, "Lyapunov, Adaptive, and Optimal Design Techniques for Cooperative Systems on Directed Communication Graphs," IEEE Trans. Industrial Electronics, vol. 59, no. 7, pp. 3026-3041, July 2012.

Hongwei Zhang, F.L. Lewis, and Abhijit Das
"Optimal design for synchronization of cooperative systems: state feedback, observer and output feedback," IEEE Trans. Automatic Control, vol. 56, no. 8, pp. 1948-1952, August 2011.

Synchronized Motion of Biological Groups

Fish school

Nature and Biological Systems are Resilient



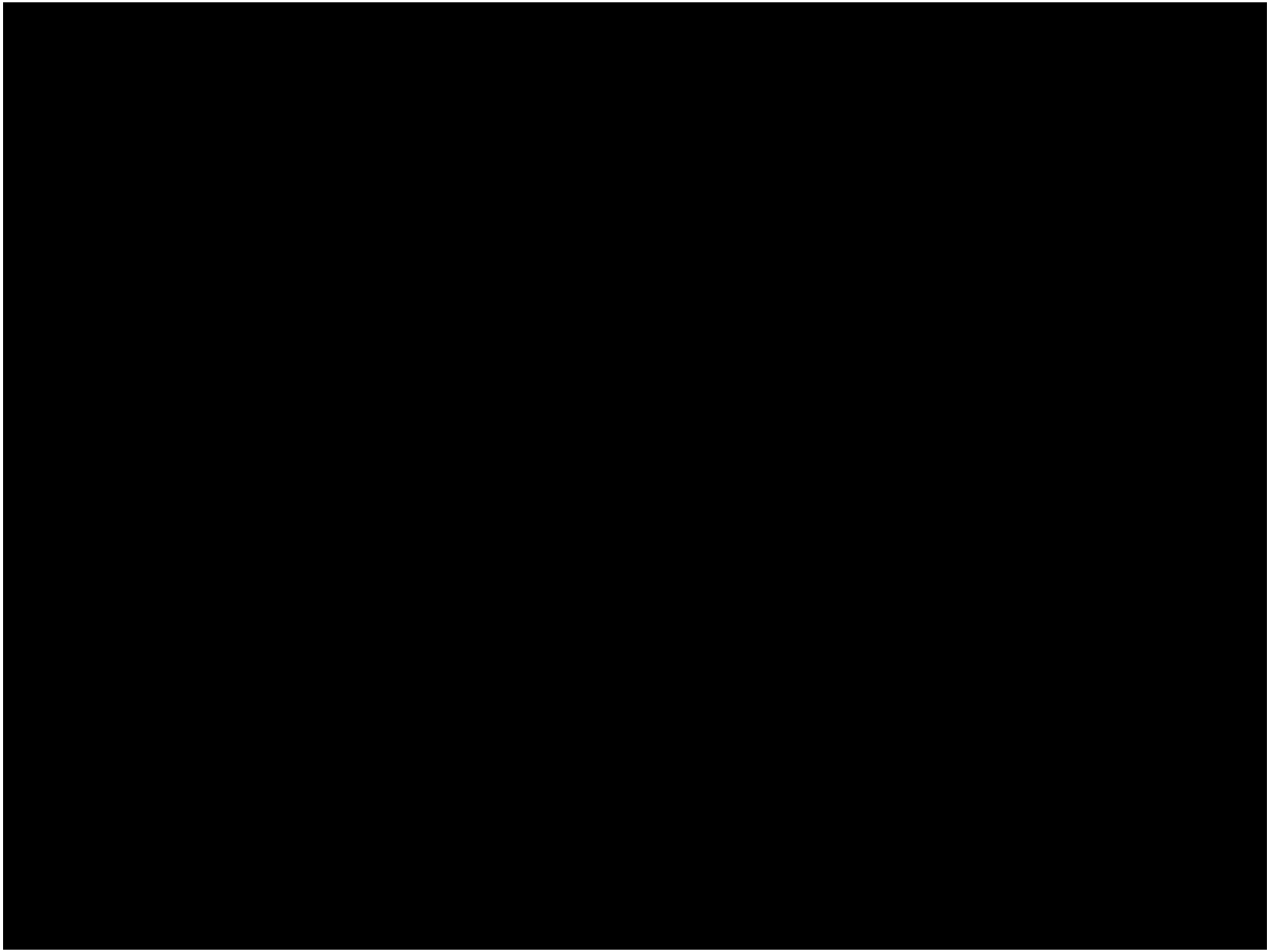
Birds flock

Locusts swarm

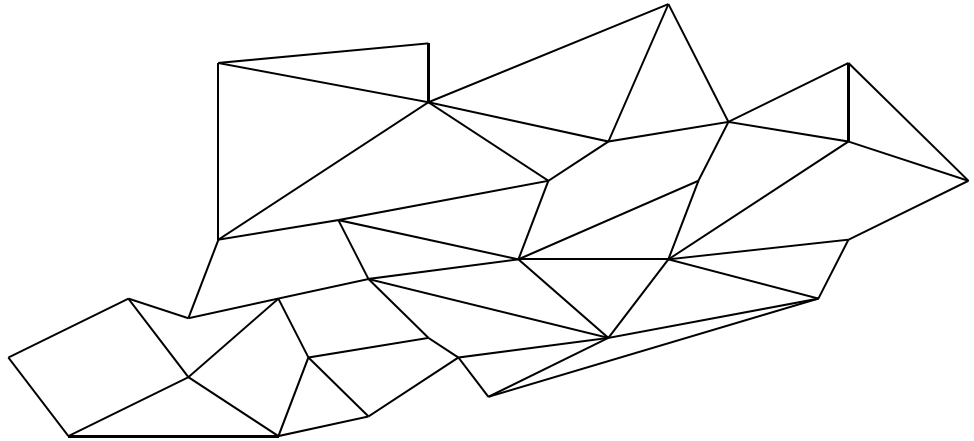
Fireflies synchronize

The Power of Synchronization

Coupled Oscillators
Diurnal Rhythm



Communication Graph

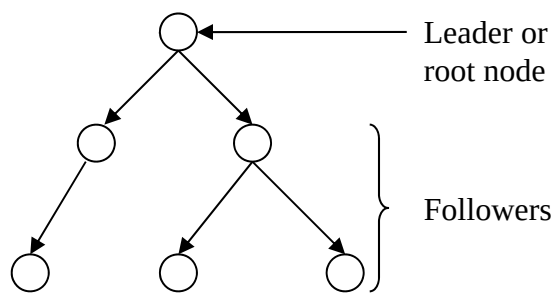


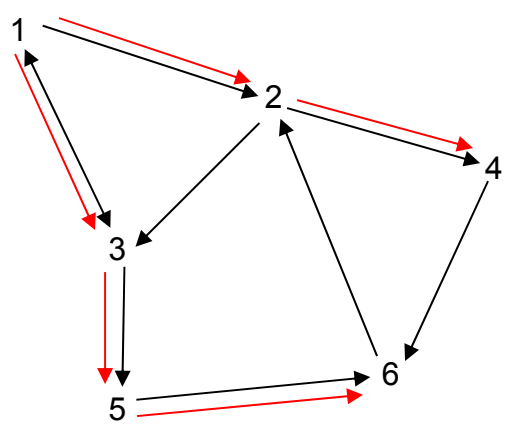
Strongly connected if for all nodes *i* and *j* there is a path from *i* to *j*.

Diameter= length of longest path between two nodes

Volume = sum of in-degrees $Vol = \sum_{i=1}^{N} d_i$

Tree- every node has in-degree=1





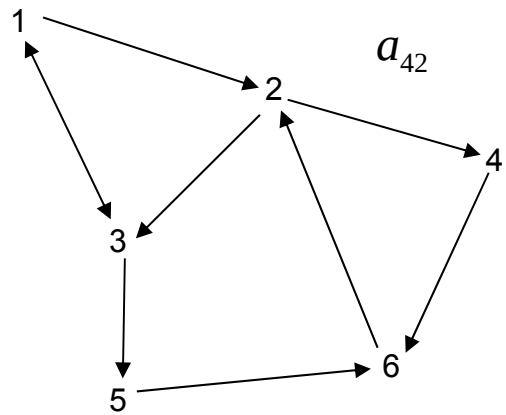
Spanning tree
Root node

Algebraic Graph Theory

Communication Graph

(V,E)

N nodes



Adjacency matrix

$$A = [a_{ij}]$$

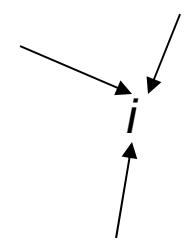
$$a_{ij} > 0 \; if \; (v_j, v_i) \in E$$

$$if \;\; j \in N_i$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

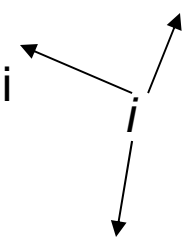
$$d_i = \sum_{j=1}^{N} a_{ij}$$ Row sum= in-degree

N_i In-neighbors of node i



$$d_i^o = \sum_{j=1}^{N} a_{ji}$$ Col sum= out-degree

N_o Out-neighbors of node i



John Baras - social standing

Graph Laplacian Matrix

Adjacency matrix

$$A = [a_{ij}]$$

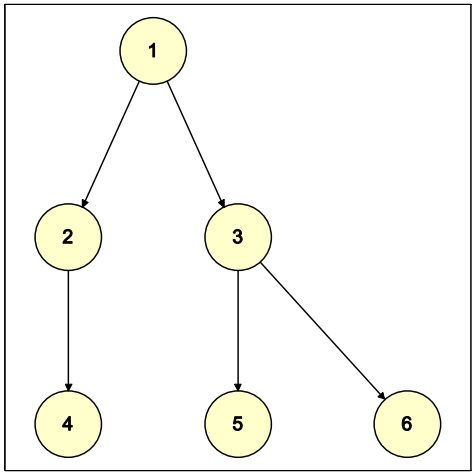
In-Degree Matrix

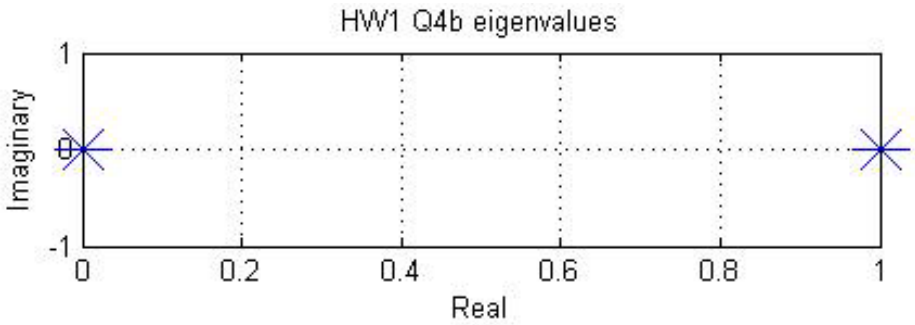
$$D = \begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_N \end{bmatrix}$$

L=D-A = graph Laplacian matrix

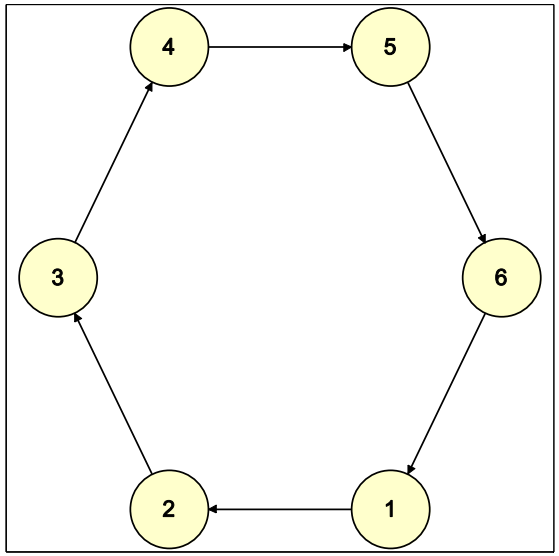
Graph Eigenvalues for Different Communication Topologies

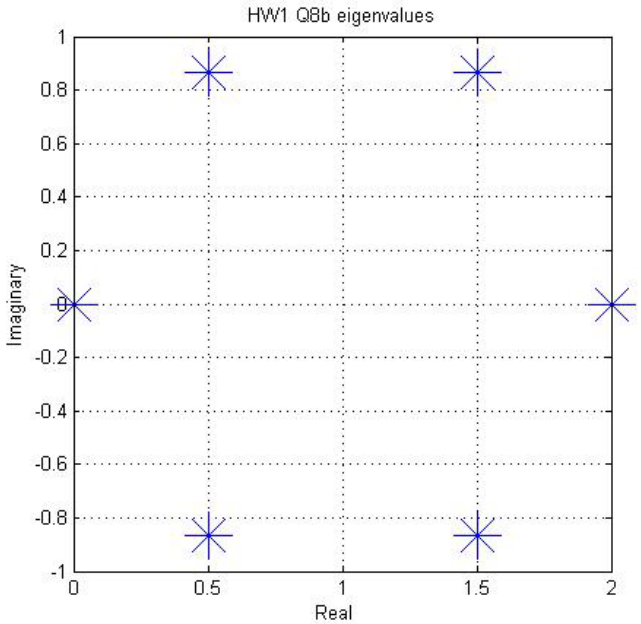
Directed Tree-
Chain of
command





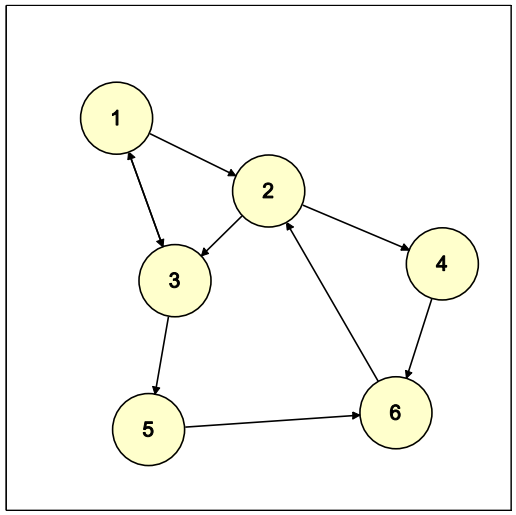
Directed Ring-
Gossip network
OSCILLATIONS

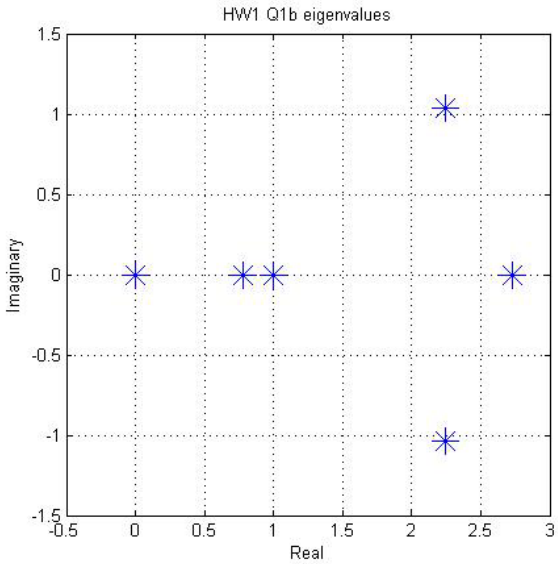




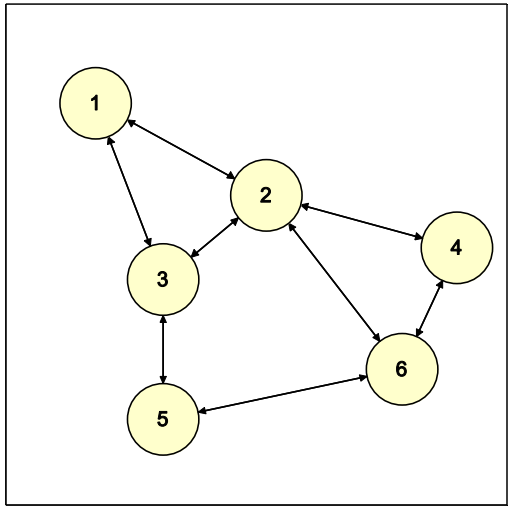
Graph Eigenvalues for Different Communication Topologies

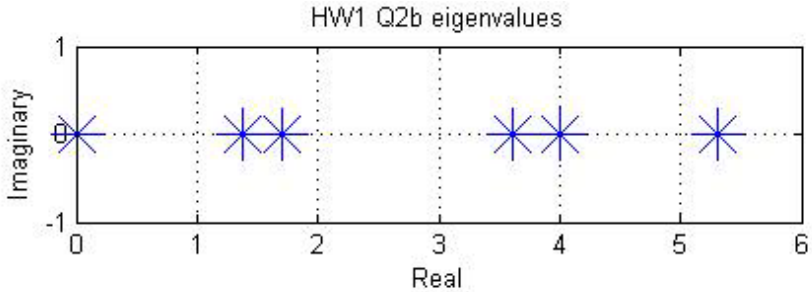
Directed graph-



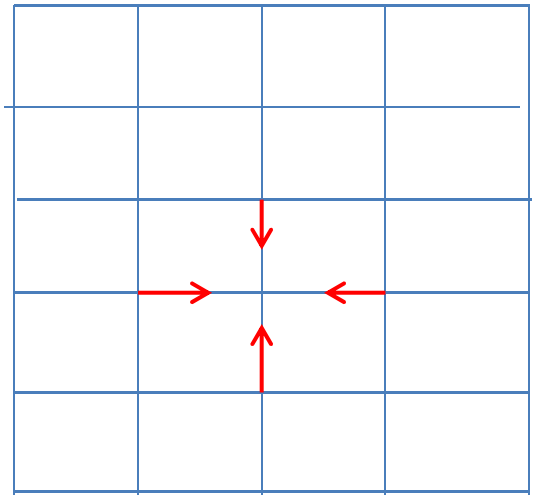


Undirected graph-

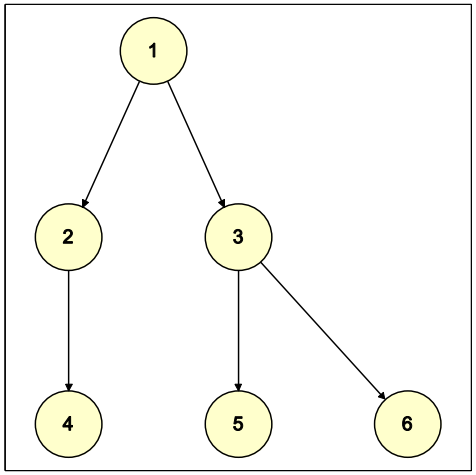


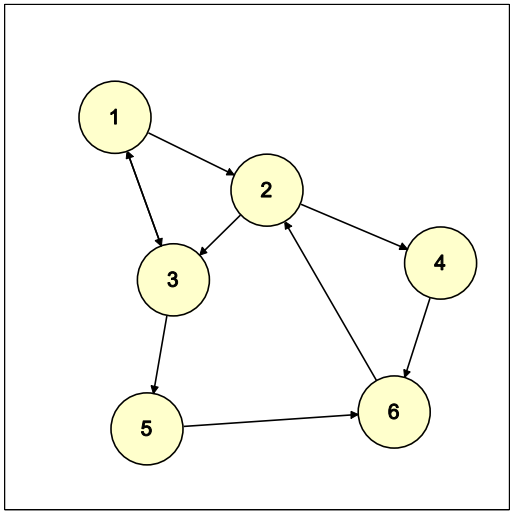


Synchronization on Good Graphs



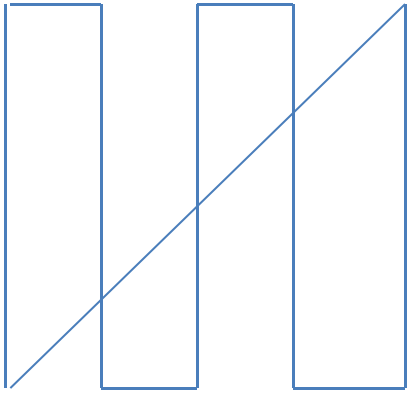
Mesh graph
4 neighbors



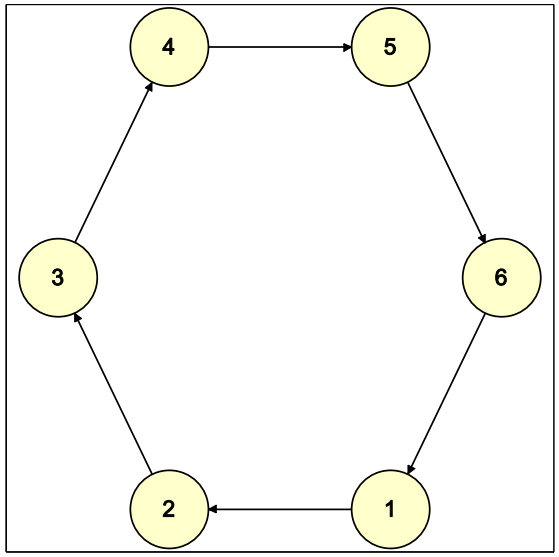


Chris Elliott fast video

Synchronization on Gossip Rings

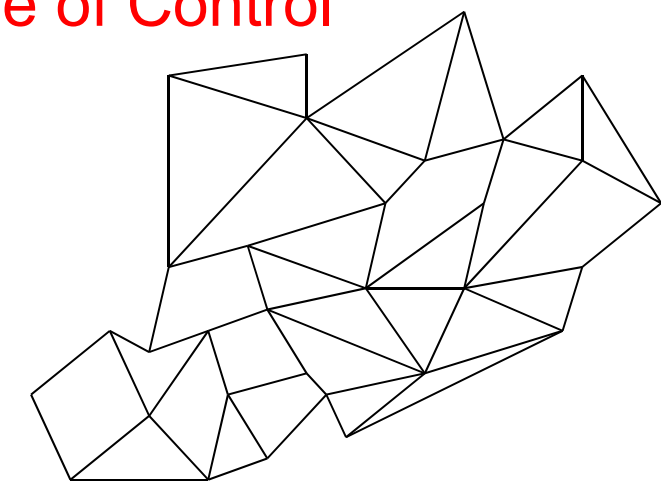


Ring graph
or cycle



Chris Elliott weird video

Dynamic Graph- the Distributed Structure of Control

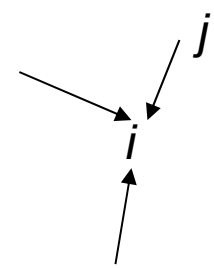


Each node has an associated state $\dot{x}_i = u_i$

Standard local voting protocol $u_i = \sum_{j \in N_i} a_{ij}(x_j - x_i)$

$$u_i = -x_i \sum_{j \in N_i} a_{ij} + \sum_{j \in N_i} a_{ij} x_j = -d_i x_i + \begin{bmatrix} a_{i1} & \cdots & a_{iN} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$

x



$$u = \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix} \qquad D = \begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_N \end{bmatrix} \qquad A = [a_{ij}]$$

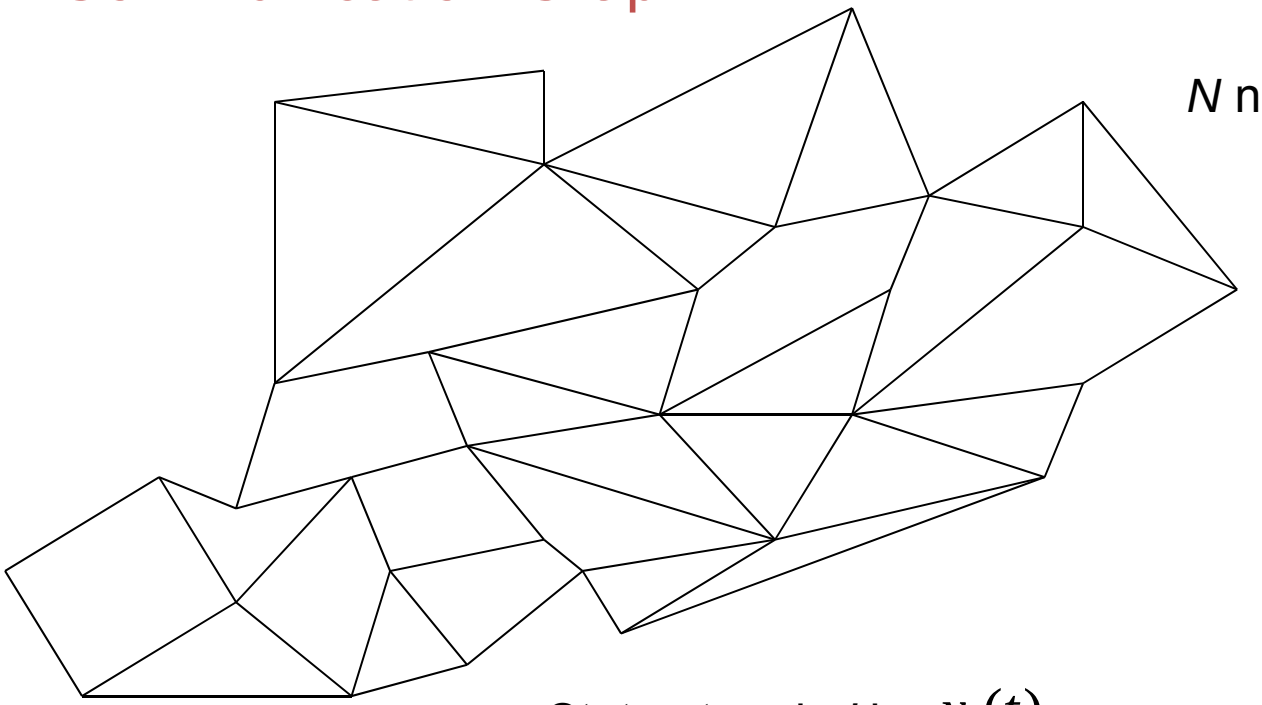
$$u = -Dx + Ax = -(D - A)x = -Lx$$

L=D-A = graph Laplacian matrix

$\dot{x} = -Lx$ Closed-loop dynamics

If x is an n-vector then $\dot{x} = -(L \otimes I_n)x$

Communication Graph



N nodes or agents

State at node *i* is $x_i(t)$

Synchronization problem $x_i(t) - x_j(t) \to 0$

Formation Control
Synchronization of Rigid-body Spacecraft Dynamics
Synchronization of Trust Opinions in Multi-agent Networks
Diurnal Rhythm Synchronization in Nature

Controlled Consensus: Cooperative Tracker

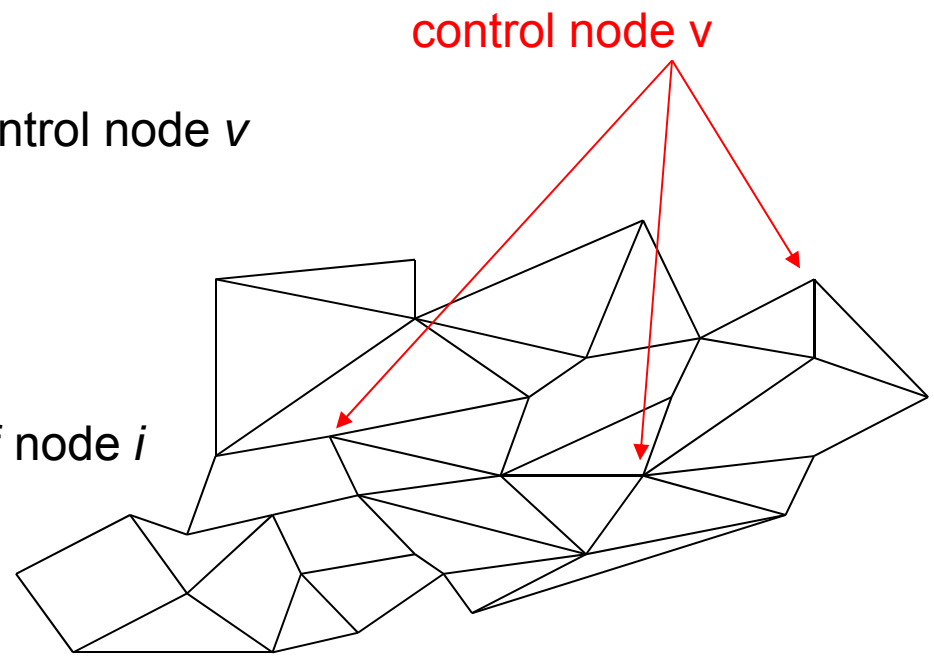
Node state $\dot{x}_i = u_i$

Distributed Local voting protocol with control node *v*

$$u_i = \sum_{j \in N_i} a_{ij}(x_j - x_i) \quad + b_i(v - x_i)$$

Local Neighborhood Tracking Error

$b_i \neq 0$ If control v is in the neighborhood of node *i*



Global state

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$

$$\dot{x} = -(L + B)x + B\underline{1}v \qquad B = diag\{b_i\}$$

Theorem. Let graph have a spanning tree and $b_i \neq 0$ for at least one root node.
Then L+B is nonsingular with all e-vals positive
and -(L+B) is asymptotically stable

Example- Non Resilience in Communication Networks

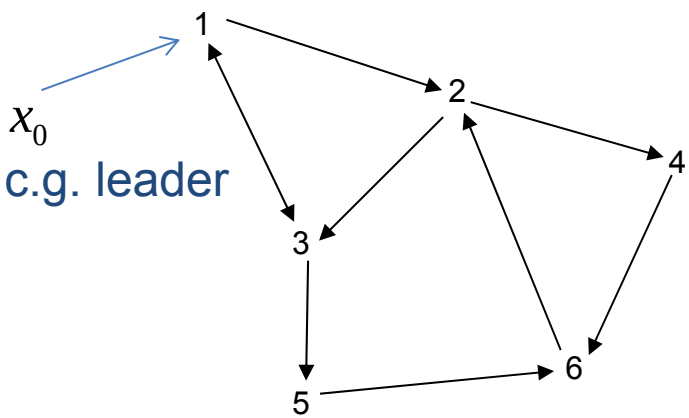
Agent Dynamics and Local Feedback Design

$$\dot{x}_i = Ax_i + Bu_i$$

$$u_i = -Kx_i$$

$$A = \begin{bmatrix} -2 & -1 \\ 2 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad K = \begin{bmatrix} 0.5 & -0.5 \end{bmatrix}$$

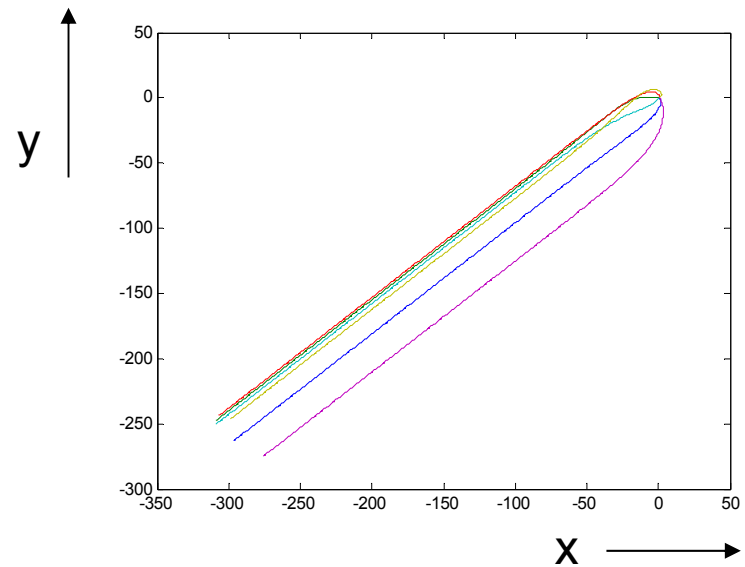
Couple 6 agents with communication graph



Local neighborhood tracking error

$$\boxed{\varepsilon_i = \sum_{j \in N_i} e_{ij}(x_j - x_i) \quad + g_i(x_0 - x_i)}$$

$$u_i = -K\varepsilon_i$$



Nodes synchronize to consensus heading

Example- Non Resilience in Communication Networks

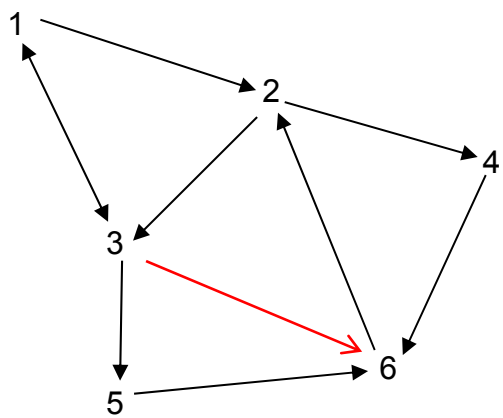
Agent Dynamics and local Feedback design

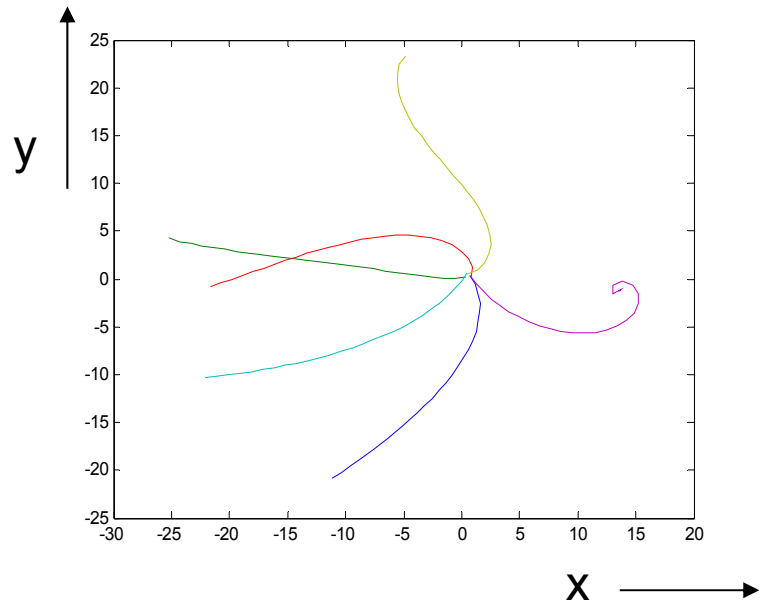
$$\dot{x}_i = Ax_i + Bu_i$$

$$u_i = -Kx_i$$

$$A = \begin{bmatrix} -2 & -1 \\ 2 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad K = \begin{bmatrix} 0.5 & -0.5 \end{bmatrix}$$

ADD another comm. Link- more information flow





Local neighborhood tracking error

$$\varepsilon_i = \sum_{j \in N_i} e_{ij}(x_j - x_i) \quad + g_i(x_0 - x_i)$$

$$u_i = K\varepsilon_i$$

Causes Unstable Formation!
WHY?

Trust-based Control

Trust Propagation and Consensus – Network Security

Inspired by social behavior in flocks, herds, teams

Foundation work by John Baras

Define ξ_{ij} as the trust that node i has for node j

$\xi_{ij} \in [-1,1]$

-1 ..... 0 1

Distrust no opinion complete trust

Define trust vector of node i as

$$\xi_i = \begin{bmatrix} \xi_{i1} \\ \xi_{i2} \\ \xi_{i3} \\ \xi_{i4} \\ \xi_{i5} \\ \xi_{i6} \end{bmatrix}$$

Trust node i has for node 3

N vector

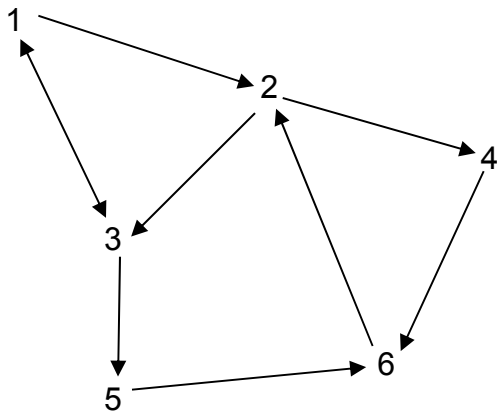
$$\xi \equiv \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_N \end{bmatrix} \in R^{N^2}$$

Standard local voting protocol $\dot{\xi}_i = u_i = \sum_{j \in N_i} a_{ij} (\underbrace{\xi_j - \xi_i})$

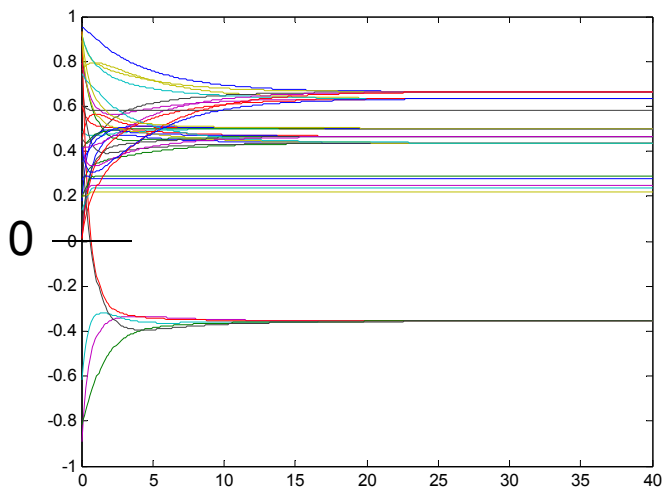
Difference of opinion with neighbors

Closed-loop trust dynamics $\dot{\xi} = -(L \otimes I_N)\xi$

Trust Propagation & Consensus



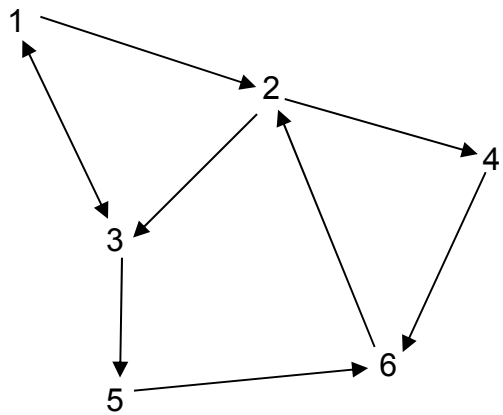
Nodes 1, 2, 4 initially distrust node 5
initial trusts are negative



Other nodes agree that
node 5 has negative trust

Convergence of trust

Trust-Based Control: Swarms/Formations



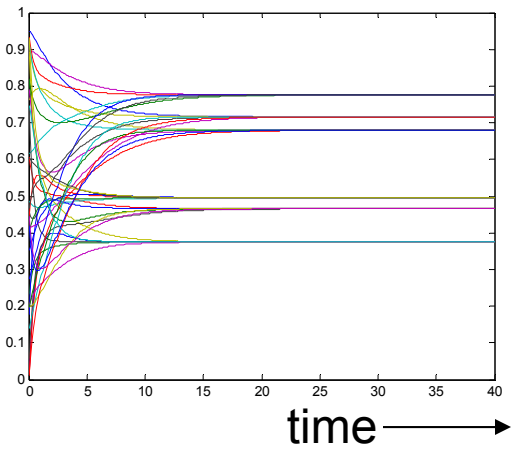
Trust dynamics $\dot{\xi}_i = \sum_{j \in N_i} a_{ij}(\xi_j - \xi_i)$

Motion dynamics

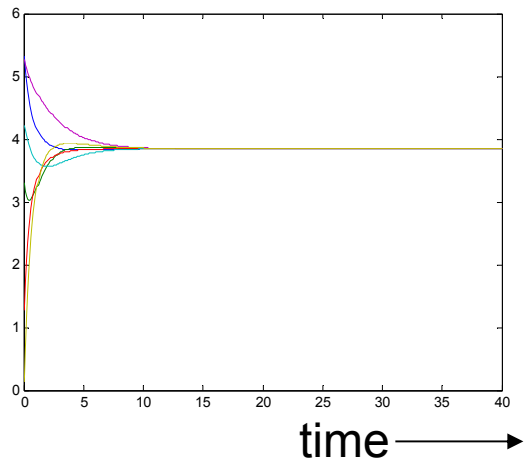
$$\dot{\theta}_i = \sum_{j \in N_i^c} \xi_{ij} a_{ij}(\theta_j - \theta_i)$$ heading angle

$$\dot{x}_i = V \cos\theta_i$$

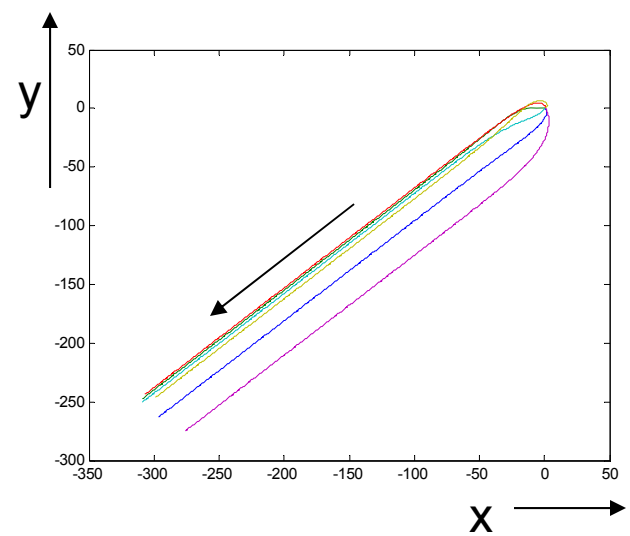
$$\dot{y}_i = V \sin\theta_i$$



Convergence of trust



Convergence of headings

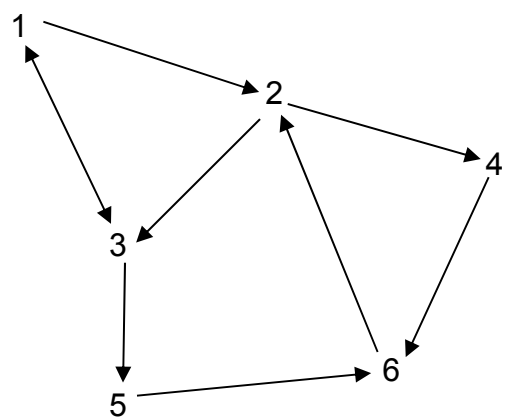


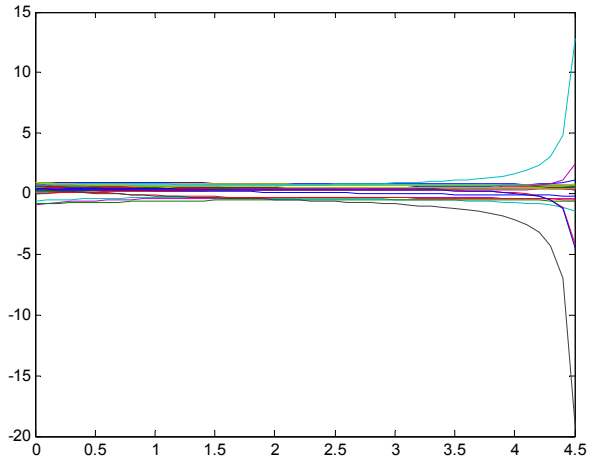
Nodes converge to consensus heading

Trust-Based Control: Swarms/Formations

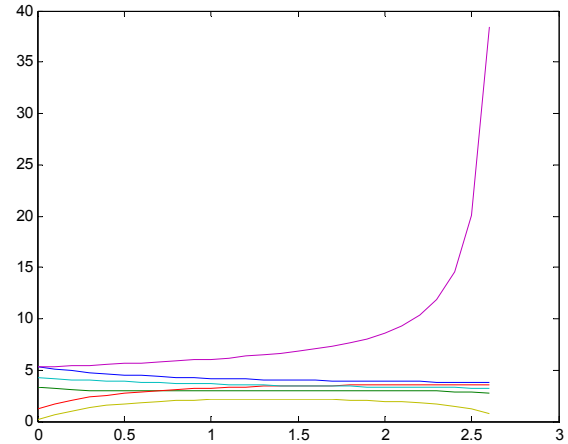
Malicious Node

$$\dot{\theta}_i = \sum_{j \in N_i^c} \xi_{ij} a_{ij} (\theta_j - \theta_i)$$



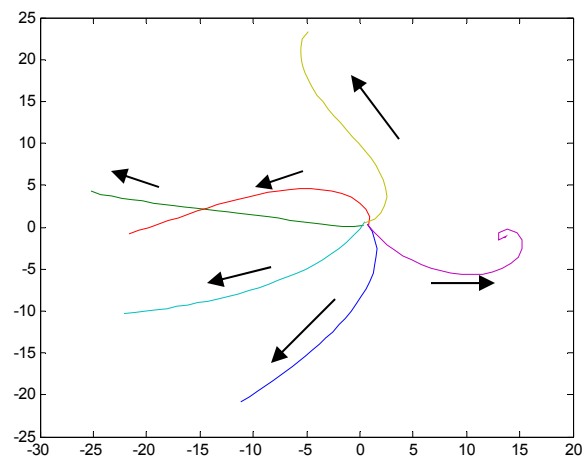


Divergence of trust



Divergence of headings

Node 5 injects
negative trust values



Causes Unstable Formation

Internal attack

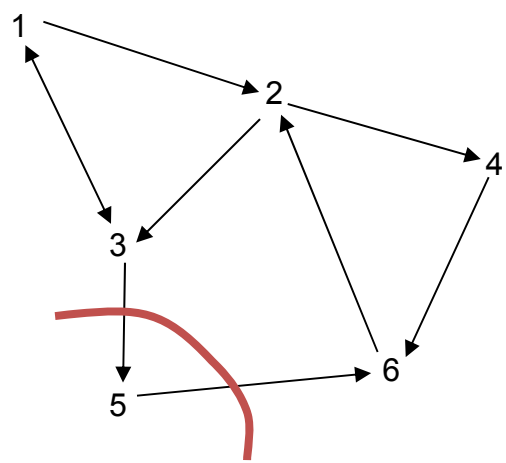
Malicious node puts out bad trust values

- i.e. false *information*
- c.f. virus propagation

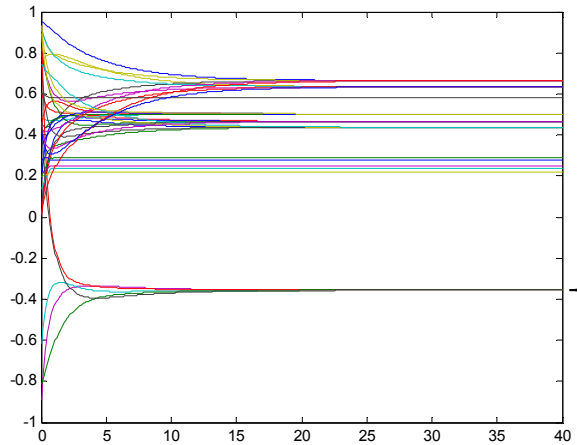
Trust-Based Control: Swarms/Formations
CUT OUT Malicious Node

$$\dot{\theta}_i = \sum_{j \in N_i^c} \xi_{ij} a_{ij} (\theta_j - \theta_i)$$ heading angle

Work by Sajal Das



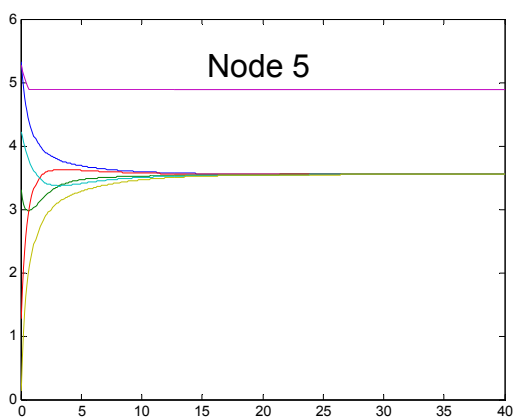
Node 5 injects
negative trust values



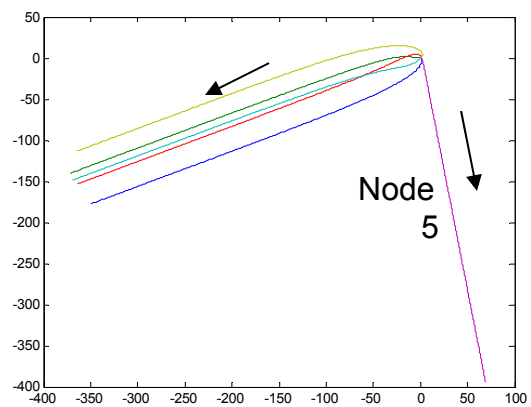
Other nodes agree that
node 5 has negative trust

Convergence of trust

If node 3 distrusts node 5,
Cut out node 5

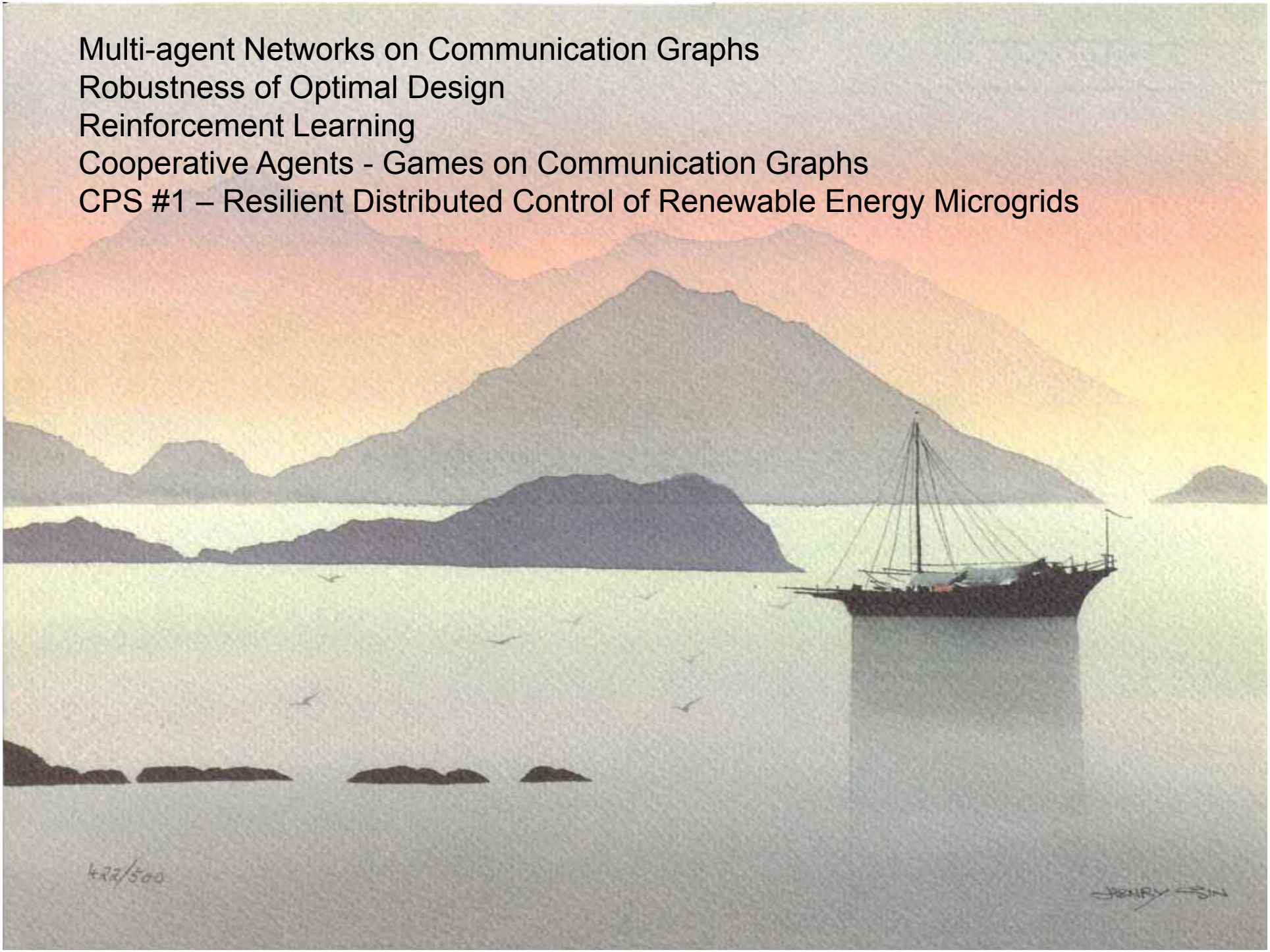


Convergence of headings



Restabilizes Formation

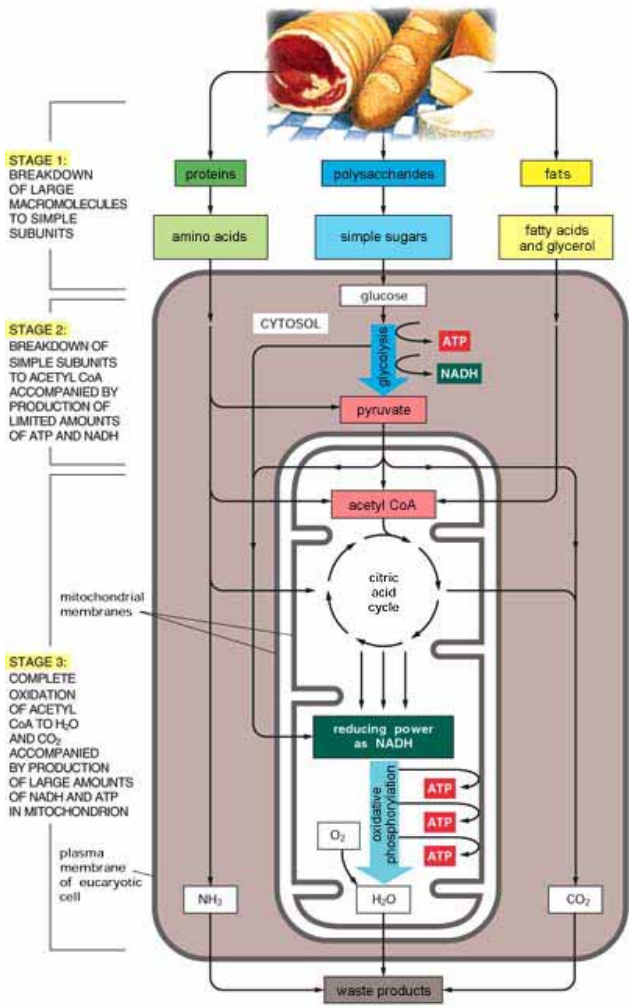
Multi-agent Networks on Communication Graphs
Robustness of Optimal Design
Reinforcement Learning
Cooperative Agents - Games on Communication Graphs
CPS #1 – Resilient Distributed Control of Renewable Energy Microgrids



Why are Biological Systems Resilient?

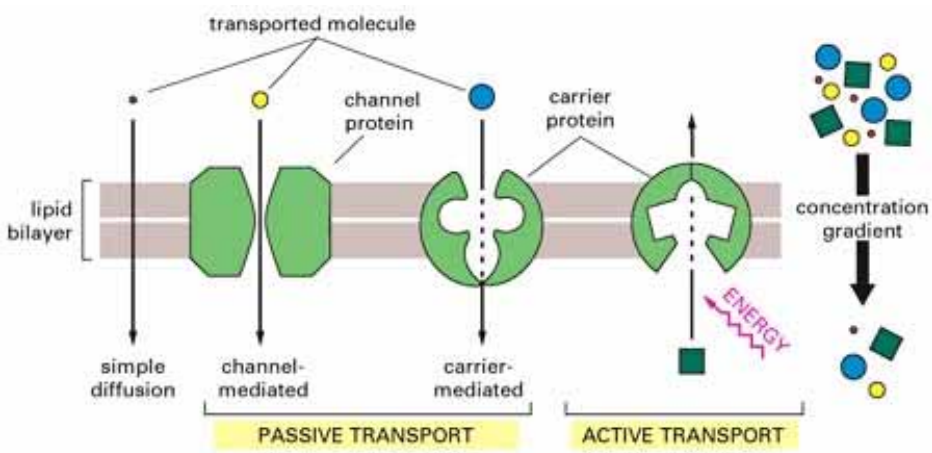
Optimality in Biological Systems

Cell Homeostasis



Cellular Metabolism

The individual cell is a complex feedback control system. It pumps ions across the cell membrane to maintain homeostatis, and has only limited energy to do so.



Permeability control of the cell membrane

http://www.accessexcellence.org/RC/VL/GG/index.html

Why are Biological Systems Resilient?

Optimality in Biological Systems

Every living organism improves its control actions based on rewards received from the environment

The resources available to living organisms are usually meager. Nature uses optimal control.

Optimality Provides an Organizational Principle for Behavior

Charles Darwin showed that Optimal Control over long timescales Is responsible for Natural Selection of Species

Reinforcement Learning

1. Apply a control. Evaluate the benefit of that control.
2. Improve the control policy.

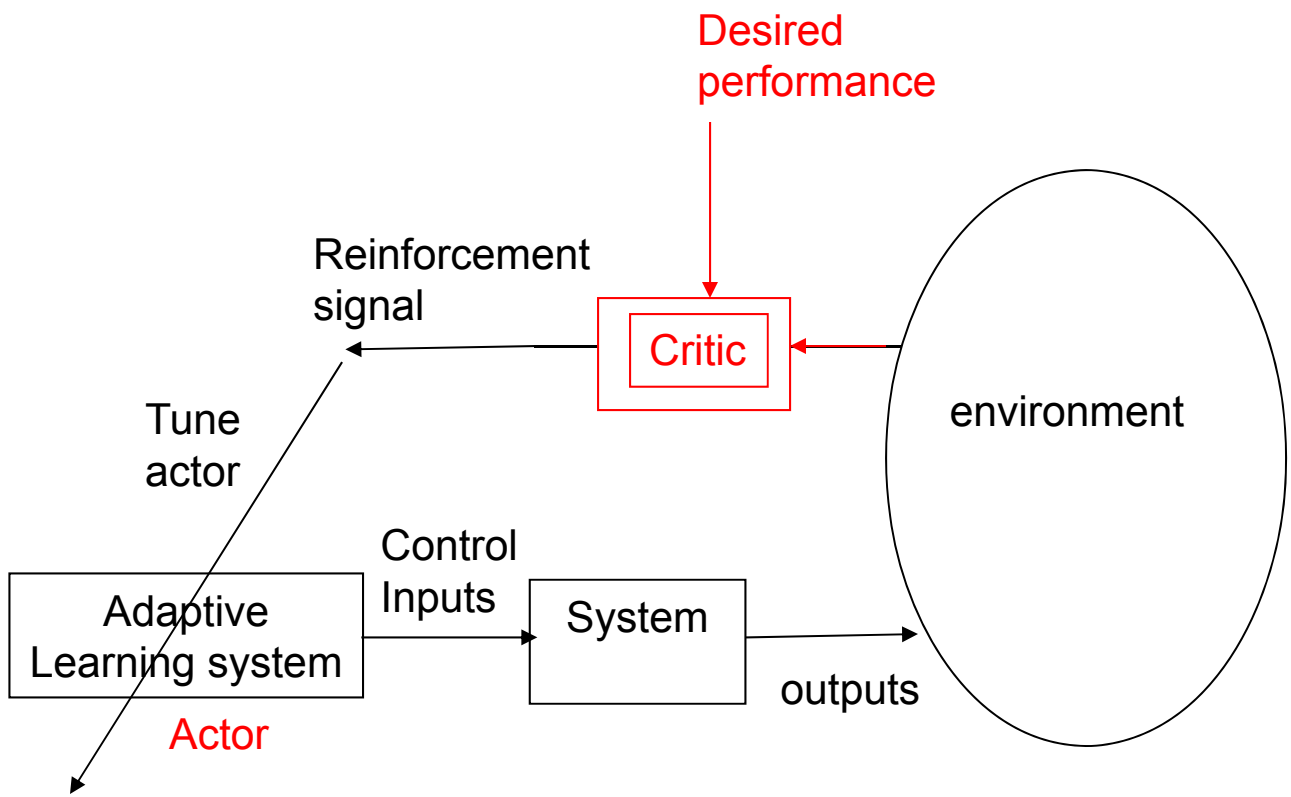
RL finds optimal policies by evaluating the effects of suboptimal policies

Different methods of learning

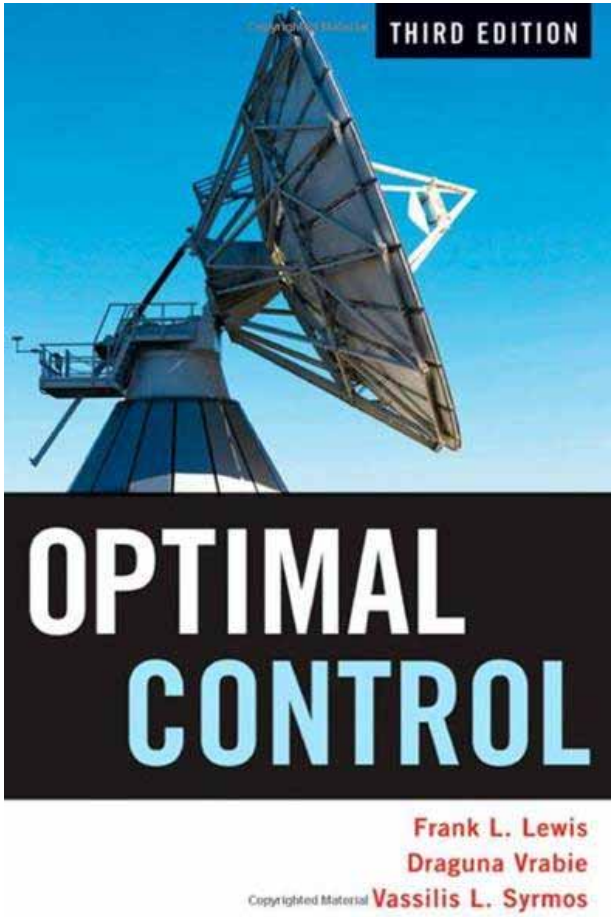
Reinforcement learning
Ivan Pavlov 1890s

We want OPTIMAL performance
ADP- Approximate Dynamic Programming

Actor-Critic Learning



Sutton & Barto book



Books

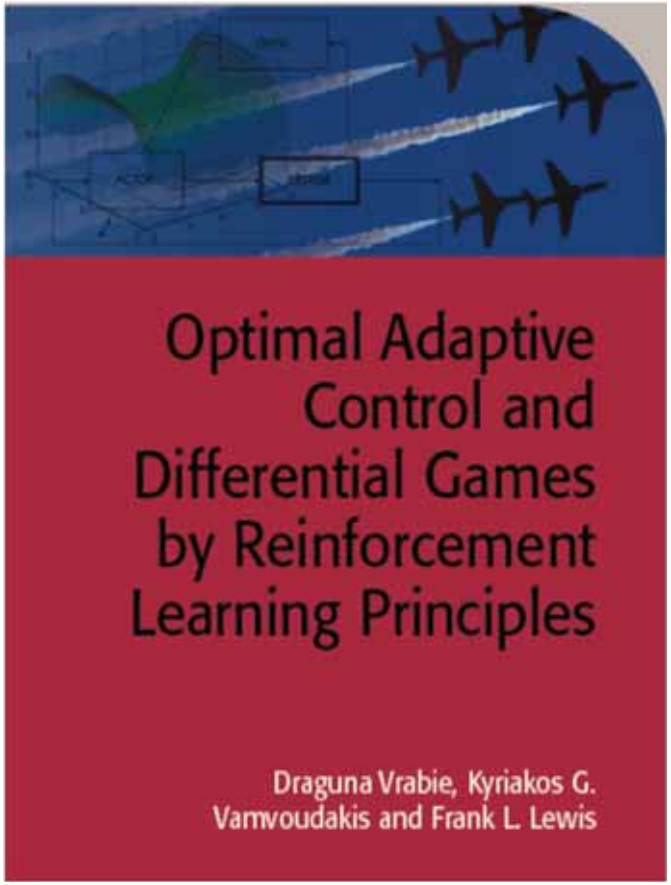
F.L. Lewis, D. Vrabie, and V. Syrmos, *Optimal Control*, third edition, John Wiley and Sons, New York, 2012.

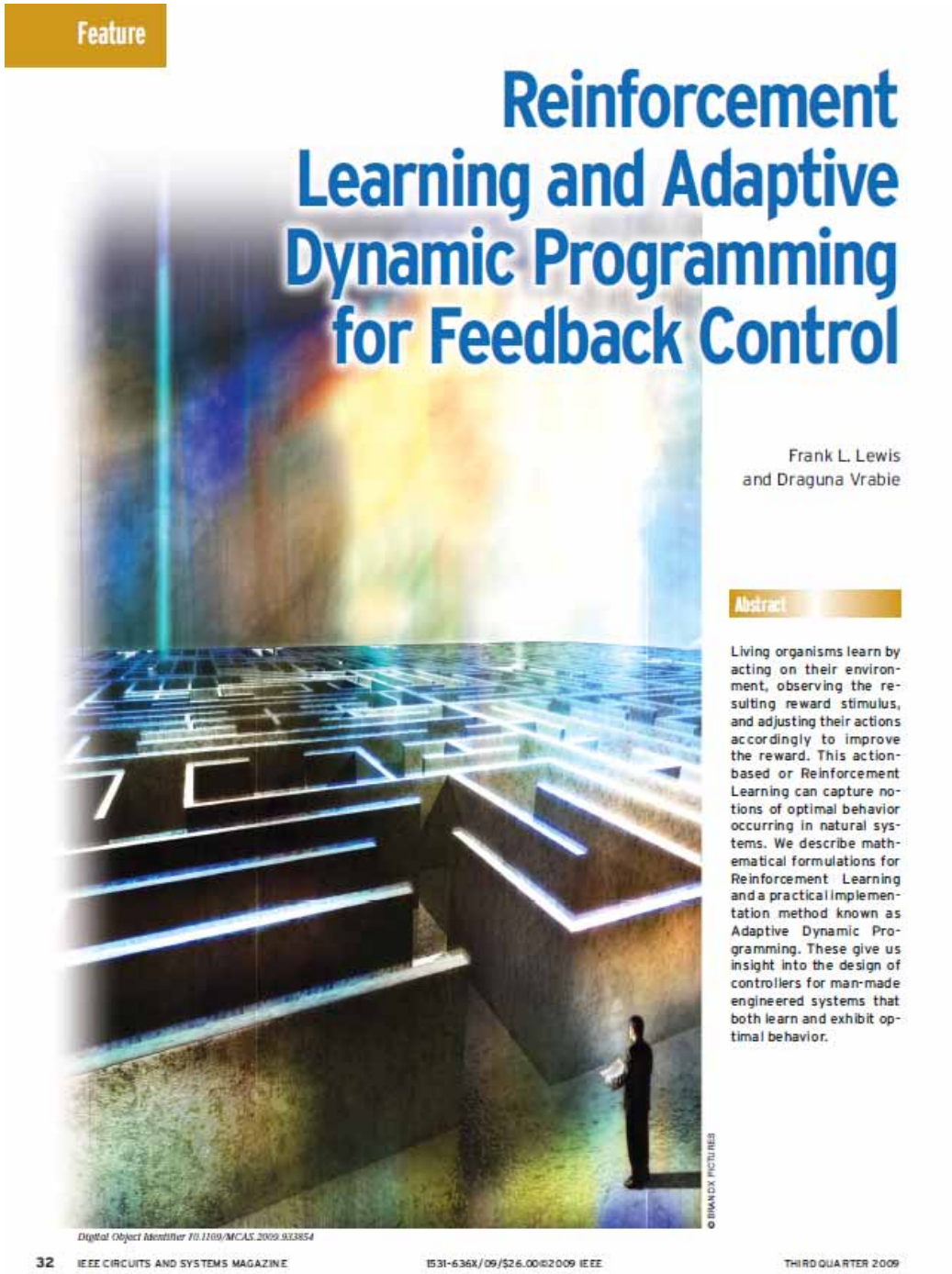
New Chapters on:

Reinforcement Learning

Differential Games

D. Vrabie, K. Vamvoudakis, and F.L. Lewis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*, IET Press, 2012.





Feature

Reinforcement Learning and Adaptive Dynamic Programming for Feedback Control

Frank L. Lewis
and Draguna Vrabie

Abstract

Living organisms learn by acting on their environment, observing the resulting reward stimulus, and adjusting their actions accordingly to improve the reward. This action-based or Reinforcement Learning can capture notions of optimal behavior occurring in natural systems. We describe mathematical formulations for Reinforcement Learning and a practical implementation method known as Adaptive Dynamic Programming. These give us insight into the design of controllers for man-made engineered systems that both learn and exhibit optimal behavior.

© BRIAN DIX PICTURES

Digital Object Identifier 10.1109/MCAS.2009.933854

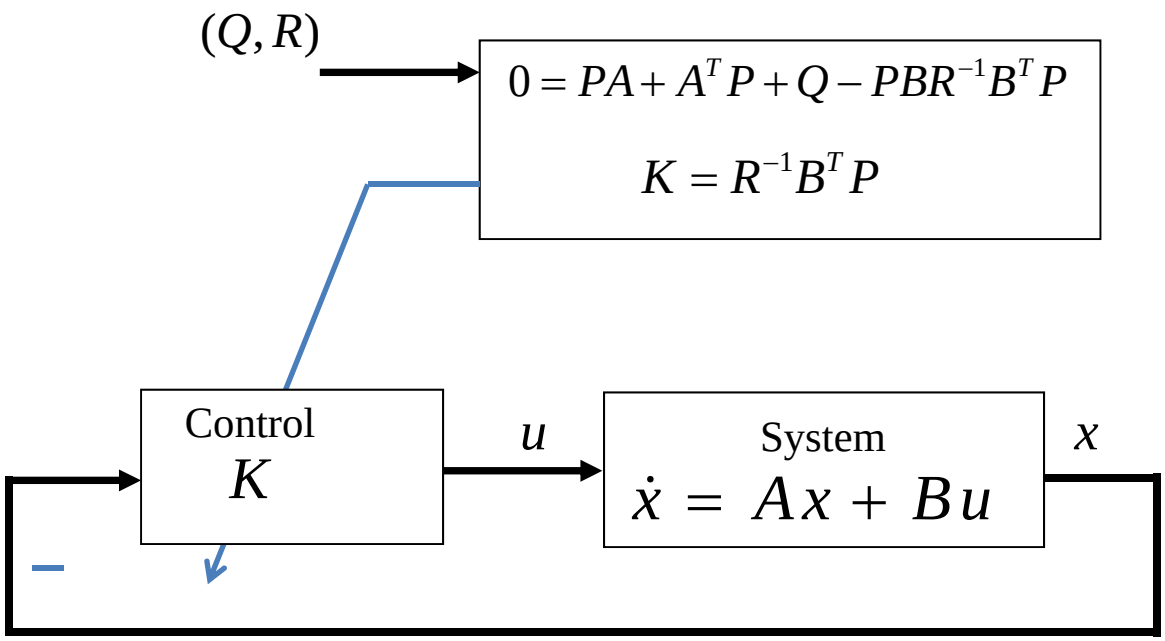
 1531-636X/09/$26.00©2009 IEEE

F.L. Lewis and D. Vrabie, “Reinforcement learning and adaptive dynamic programming for feedback control,” IEEE Circuits & Systems Magazine, Invited Feature Article, pp. 32-50, Third Quarter 2009.

IEEE Control Systems Magazine, F. Lewis, D. Vrabie, and K. Vamvoudakis, “Reinforcement learning and feedback Control,” Dec. 2012

Optimal Control- The Linear Quadratic Regulator (LQR)

User prescribed optimization criterion $$V(x(t)) = \int_t^{\infty} (x^T Q x + u^T R u)\, d\tau$$



FORMAL DESIGN PROCEDURE
- Off-line Design Loop
- Using ARE

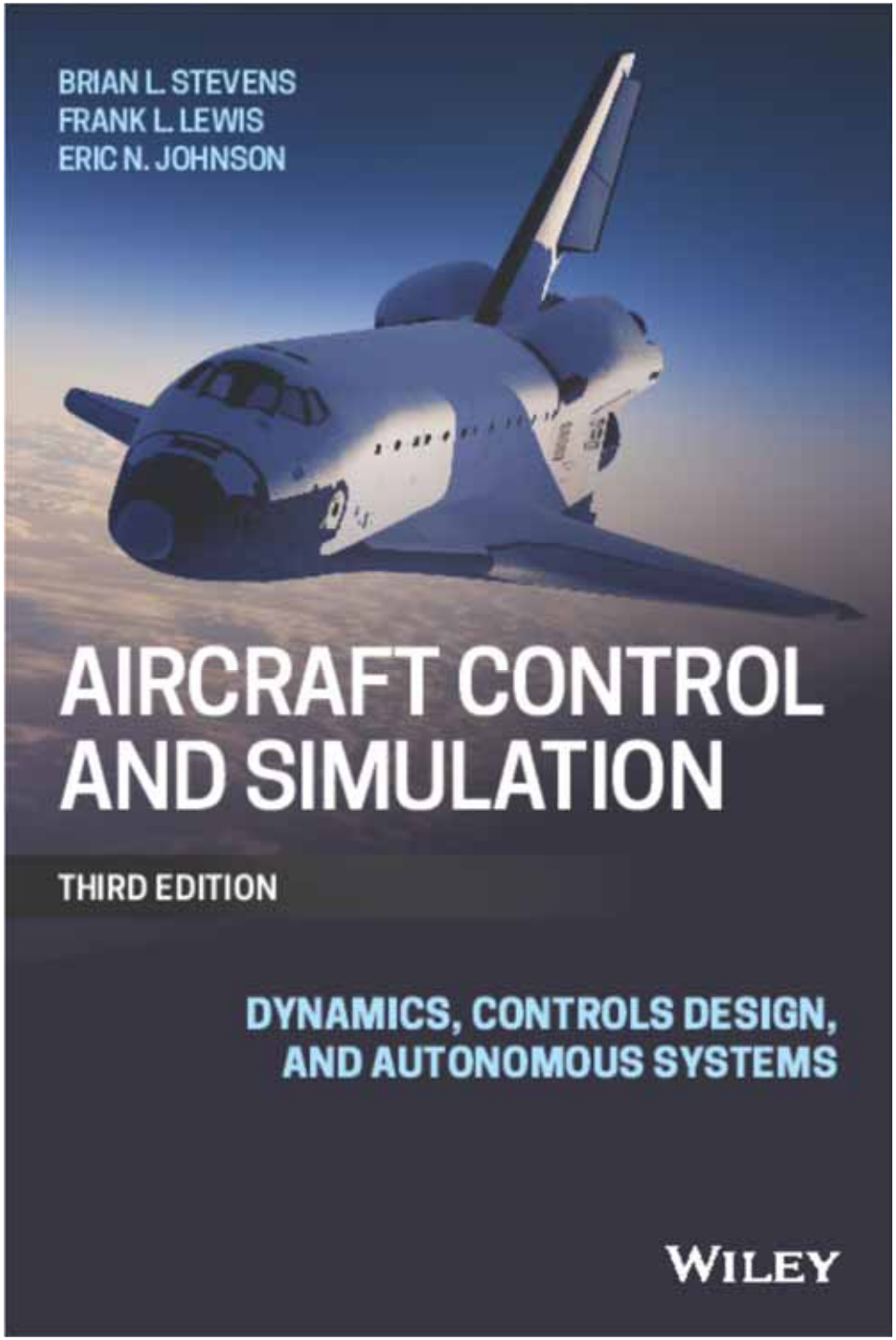
On-line real-time Control Loop

OPTIMAL CONTROL IS
- RESILIENT
- ROBUST
- HAS GUARANTEED STABILITY MARGINS

An Offline Design Procedure
that requires Knowledge of system dynamics model (A,B)

System modeling is expensive, time consuming, and inaccurate

Aircraft Autopilots -
Linear Quadratic Regulator
Linear Quadratic Gaussian LTR



Applications at Boeing Defense Space & Security
Kevin Wise and Eugene Lavretsky

Highly reliable adaptive uncertainty approximation compensators for flight control applications:

unmanned aircraft – Phantom Ray

Tailkit adaptive control systems for
Joint Direct Attack Munition (JDAM) munitions:
Mk-82, Mk-84, and Laser-guided variants
Fielded for defense in Iraq and Afghanistan.

We want to find optimal control solutions

Online in real-time

Using adaptive control techniques

Without knowing the full dynamics

For nonlinear systems and general performance indices

Adaptive Control Structures for Resilience:

A. Optimal control B. Zero-sum games C. Non zero-sum games

---

1. System dynamics
2. Value/cost function
3. Bellman equation
4. HJ solution equation (Riccati eq.)
5. Policy iteration – gives the structure we need

Adaptive Control is online and works for unknown systems.
Generally not Optimal

Optimal Control is off-line,
and needs to know the system dynamics to solve design eqs.

We want ONLINE DIRECT ADAPTIVE OPTIMAL Control
For any performance cost of our own choosing

Reinforcement Learning turns out to be the key to this!

RL brings together Optimal Control and Adaptive Control

Derivation of Nonlinear Optimal Regulator

Nonlinear System dynamics $\dot{x} = f(x,u) = f(x) + g(x)u$

Cost/value $V(x(t)) = \int_t^\infty r(x,u)\,dt = \int_t^\infty (Q(x) + u^T R u)\,dt$

Leibniz gives
Differential equivalent

Bellman Equation, in terms of the Hamiltonian function

$$H(x, \frac{\partial V}{\partial x}, u) = \dot{V} + r(x,u) = \left(\frac{\partial V}{\partial x}\right)^T \dot{x} + r(x,u) = \left(\frac{\partial V}{\partial x}\right)^T \left(f(x) + g(x)u\right) + r(x,u) = 0$$

Stationarity condition $\frac{\partial H}{\partial u} = 0$

Stationary Control Policy $u = h(x) = -\tfrac{1}{2} R^{-1} g^T(x) \frac{\partial V}{\partial x}$

Hamilton-Jacobi-Bellmen Equation

$$0 = \left(\frac{dV^*}{dx}\right)^T f + Q(x) - \tfrac{1}{4}\left(\frac{dV^*}{dx}\right)^T g R^{-1} g^T \frac{dV^*}{dx}$$

Off-line solution
HJB hard to solve. May not have smooth solution.
Dynamics must be known

Integral Reinforcement Learning

To find online methods for optimal control

Iterate on these two equations

Nonlinear System dynamics $\dot{x} = f(x,u) = f(x) + g(x)u$

Cost/value $V(x(t)) = \int_t^\infty r(x,u)\,dt = \int_t^\infty (Q(x) + u^T R u)\,dt$

$$V(x(t)) = \int_t^\infty r(x,u)\,d\tau = \int_t^{t+T} r(x,u)\,d\tau + \int_{t+T}^\infty r(x,u)\,d\tau$$

Bellman Equation, in terms of the Hamiltonian function

$$V(x(t)) = \int_t^{t+T} r(x,u)\,d\tau \quad + \quad V(x(t+T)), \quad V(0) = 0$$

Good Bellman Equation

Stationarity condition $\frac{\partial H}{\partial u} = 0$

Stationary Control Policy $\boxed{u = h(x) = -\tfrac{1}{2} R^{-1} g^T(x) \frac{\partial V}{\partial x}}$

Integral Reinforcement Learning (IRL)- Draguna Vrabie

IRL Policy iteration

Policy evaluation- IRL Bellman Equation

Cost update $$\underline{V_k}(x(t)) = \int_t^{t+T} r(x,u_k)\,dt \quad + \quad \underline{V_k}(x(t+T))$$

CT Bellman eq.

f(x) and *g(x)* do not appear

Equivalent to $$0 = \left(\frac{\partial V}{\partial x}\right)^T f(x,u) + r(x,u) \equiv H(x,\frac{\partial V}{\partial x},u)$$

Solves Bellman eq. (nonlinear Lyapunov eq.) without knowing system dynamics

Policy improvement

Control gain update $$u_{k+1} = h_{k+1}(x) = -\tfrac{1}{2} R^{-1} g^T(x) \frac{\partial V_k}{\partial x}$$

g(x) needed for control update

Initial stabilizing control is needed

Converges to solution to HJB eq. $$0 = \left(\frac{dV^*}{dx}\right)^T f + Q(x) - \tfrac{1}{4}\left(\frac{dV^*}{dx}\right)^T g R^{-1} g^T \frac{dV^*}{dx}$$

D. Vrabie proved convergence to the optimal value and control
Automatica 2009, Neural Networks 2009

Real-time Implementation
Approximate Dynamic Programming

Value Function Approximation (VFA) to Solve Bellman Equation

$$V_k(x(t)) = \int_t^{t+T} \left(Q(x) + u_k{}^T R u_k \right) dt + V_k(x(t+T))$$

Approximate value by Weierstrass Approximator Network $V = W^T \phi(x)$

$$W_k^T \phi(x(t)) = \int_t^{t+T} \left(Q(x) + u_k{}^T R u_k \right) dt + W_k^T \phi(x(t+T))$$

$$W_k^T \underbrace{\left[\phi(x(t)) - \phi(x(t+T)) \right]}_{\text{regression vector}} = \int_t^{t+T} \underbrace{\left(Q(x) + u_k{}^T R u_k \right) dt}_{\text{Reinforcement on time interval } [t,\ t+T]}$$

Scalar equation
with vector unknowns

Same form as standard System ID problems

Now use RLS along the trajectory to get new weights W_k

Then find updated FB

$$u_{k+1} = h_{k+1}(x) = -\tfrac{1}{2} R^{-1} g^T(x) \frac{\partial V_k}{\partial x} = -\tfrac{1}{2} R^{-1} g^T(x) \left[\frac{\partial \phi(x(t))}{\partial x(t)} \right]^T W_k$$

Direct Optimal Adaptive Control for Partially Unknown CT Systems

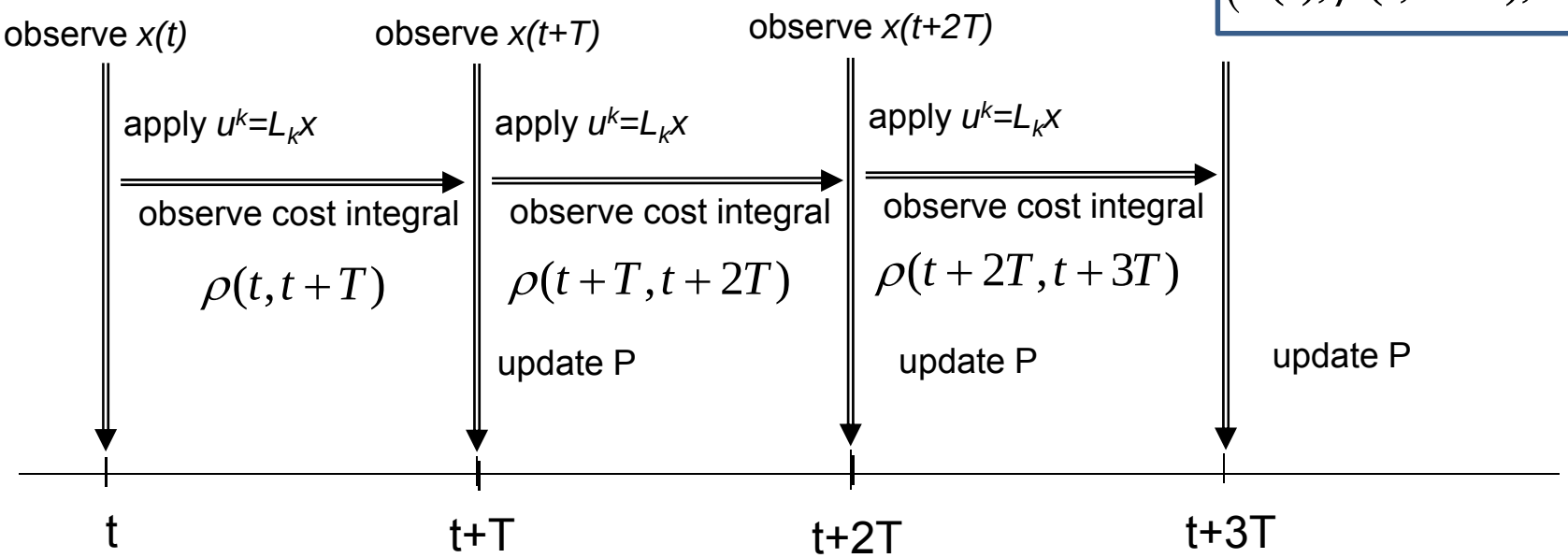
Integral Reinforcement Learning (IRL)

Solve Bellman Equation - Solves Lyapunov eq. without knowing dynamics

$$W_k^T\left[\phi(x(t))-\phi(x(t+T))\right]=\int_t^{t+T} x(\tau)^T\left(Q+K_k^T R K_k\right)x(\tau)\,d\tau=\rho(t,t+T)$$

Data set at time *[t,t+T)*

$$\left(x(t),\rho(t,t+T),x(t+T)\right)$$



Do RLS until convergence to P_k

Or use batch least-squares

A is not needed anywhere

update control gain

$$K_{k+1}=R^{-1}B^T P_k$$

This is a data-based approach that uses measurements of *x(t), u(t)*
Instead of the plant dynamical model.

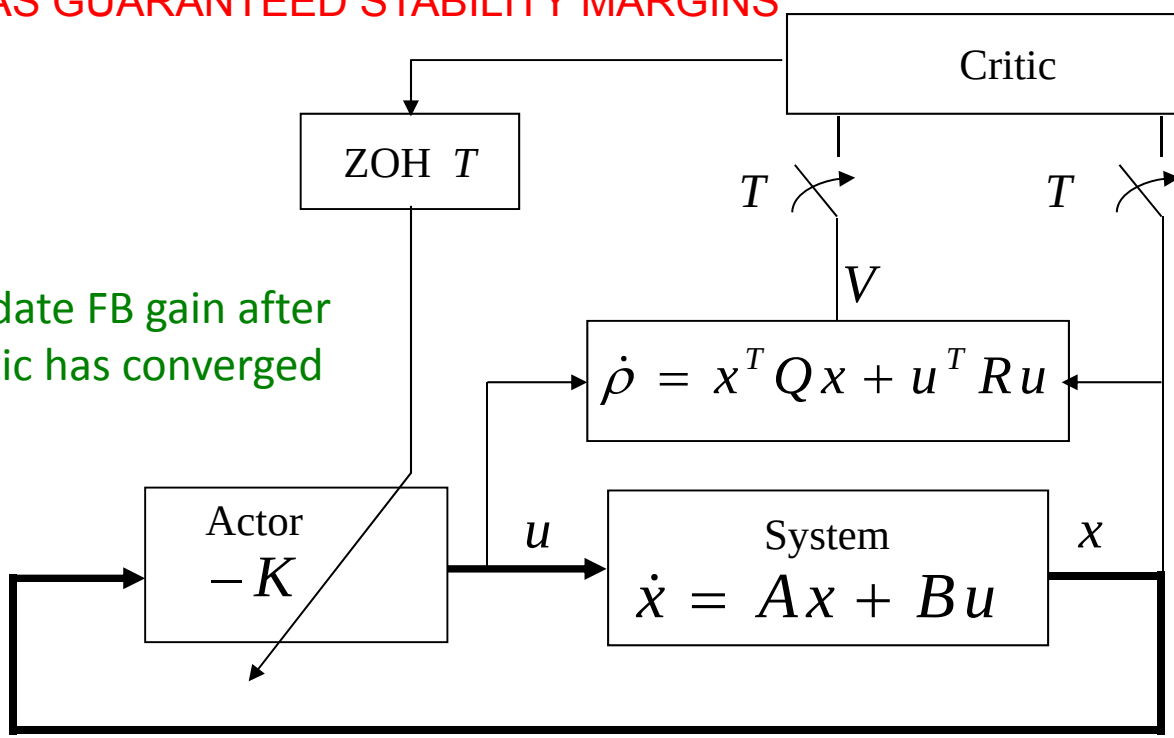
Draguna Vrabie

Direct Optimal Adaptive Controller

Solves Riccati Equation Online without knowing A matrix

OPTIMAL CONTROL IS
- RESILIENT
- ROBUST
- HAS GUARANTEED STABILITY MARGINS

Run RLS or use batch L.S.
To identify value of current control



Dynamic
Control
System
w/ MEMORY

Update FB gain after
Critic has converged

A hybrid continuous/discrete dynamic controller
whose internal state is the observed cost over the interval

Reinforcement interval T can be selected on line on the fly – can change

Optimal Adaptive IRL for CT systems

D. Vrabie, 2009

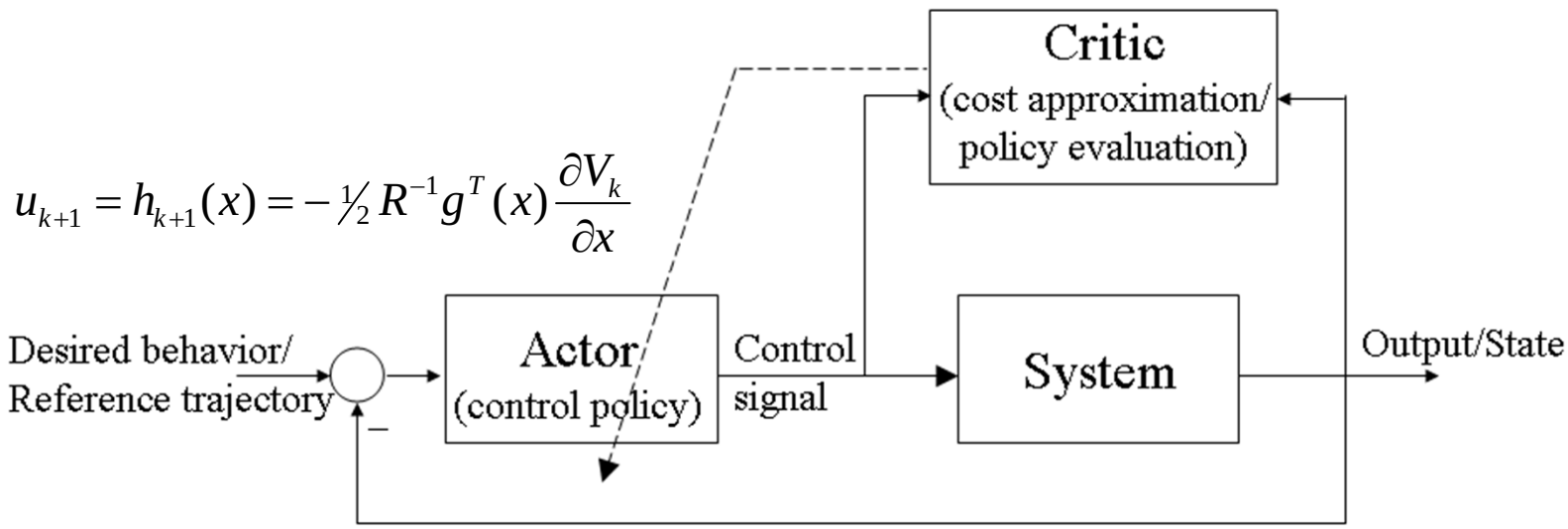
Actor / Critic structure for CT Systems

Reinforcement learning

$$V_k(x(t)) = \int_t^{t+T} r(x,u_k)\,dt \quad + \quad V_k(x(t+T))$$

Theta waves 4-8 Hz

$$u_{k+1} = h_{k+1}(x) = -\tfrac{1}{2} R^{-1} g^T(x) \frac{\partial V_k}{\partial x}$$



Motor control 200 Hz

A new structure of adaptive controllers

Oscillation is a fundamental property of neural tissue

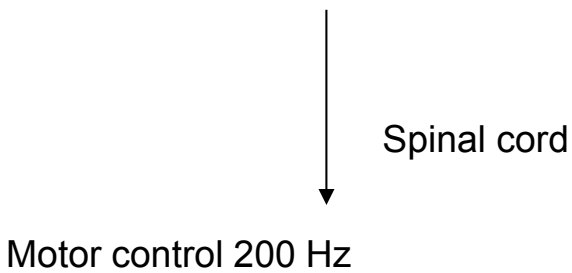
Brain has multiple adaptive clocks with different timescales

gamma rhythms 30-100 Hz, hippocampus and neocortex
high cognitive activity.

- consolidation of memory
- spatial mapping of the environment – place cells

The high frequency processing is due to the large amounts of sensorial data to be processed

theta rhythm, Hippocampus, Thalamus, 4-10 Hz
sensory processing, memory and voluntary control of movement.



D. Vrabie, F.L. Lewis, D. Levine, “Neural Network-Based Adaptive Optimal Controller- A Continuous-Time Formulation -,” Proc. Int. Conf. Intelligent Control, Shanghai, Sept. 2008.

D. Vrabie and F.L. Lewis, “Neural network approach to continuous-time direct adaptive optimal control for partially-unknown nonlinear systems,” Neural Networks, vol. 22, no. 3, pp. 237-246, Apr. 2009.

Deliberative evaluation

Limbic system

control

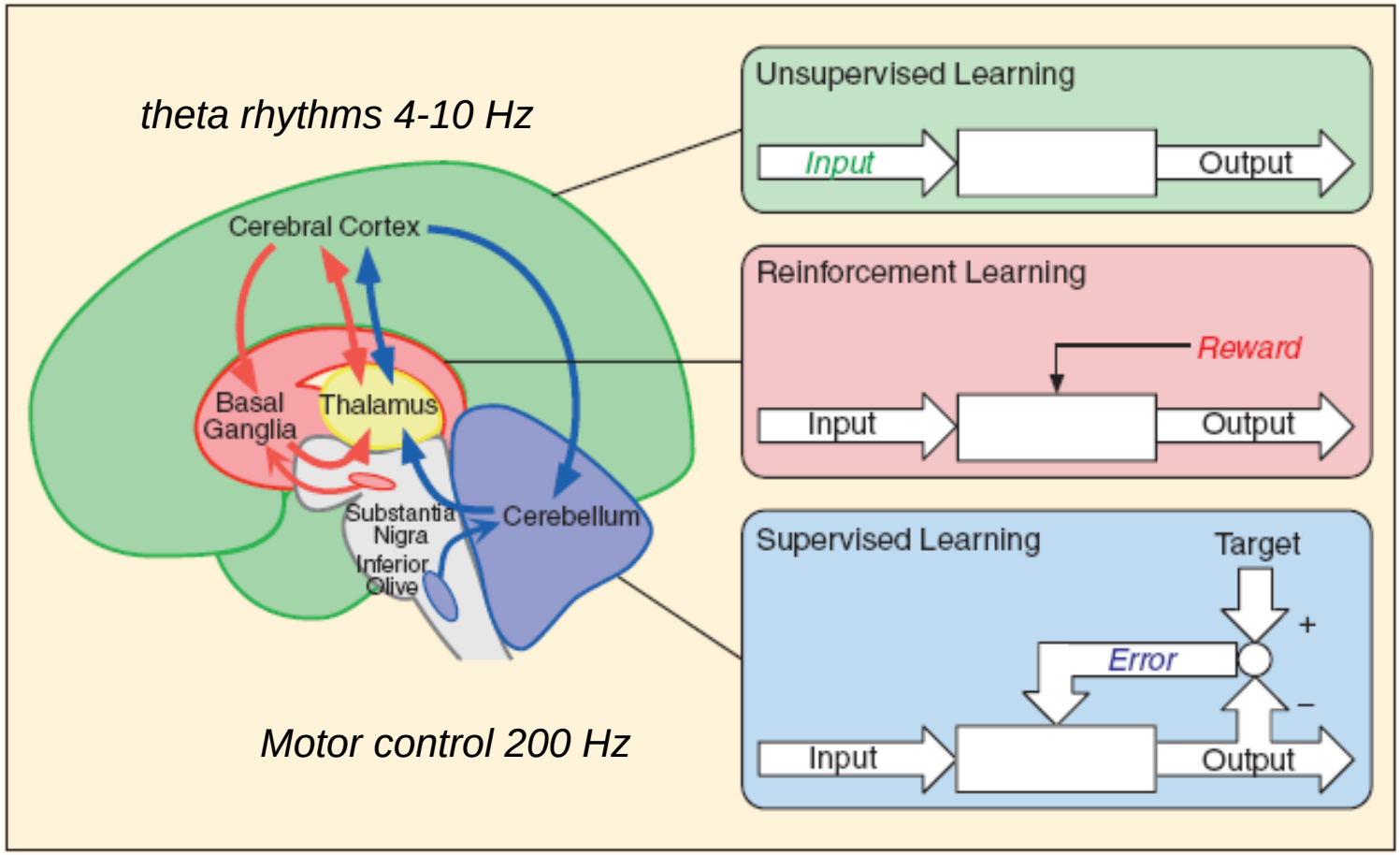


Figure 1. *Learning-oriented specialization of the cerebellum, the basal ganglia, and the cerebral cortex [1], [2]. The cerebellum is specialized for supervised learning based on the error signal encoded in the climbing fibers from the inferior olive. The basal ganglia are specialized for reinforcement learning based on the reward signal encoded in the dopaminergic fibers from the substantia nigra. The cerebral cortex is specialized for unsupervised learning based on the statistical properties of the input signal.*

Doya, Kimura, Kawato 2001

OPTIMALITY AND RESILIENCE OF BIOLOGICAL SYSTEMS

Simulation 1- F-16 aircraft pitch rate controller

$$\dot{x} = \begin{bmatrix} -1.01887 & 0.90506 & -0.00215 \\ 0.82225 & -1.07741 & -0.17555 \\ 0 & 0 & -1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

Stevens and Lewis 2003

$$x = [\alpha \quad q \quad \delta_e]$$

$$Q = I, \quad R = I$$

ARE $\quad 0 = PA + A^T P + Q - PBR^{-1}B^T P$

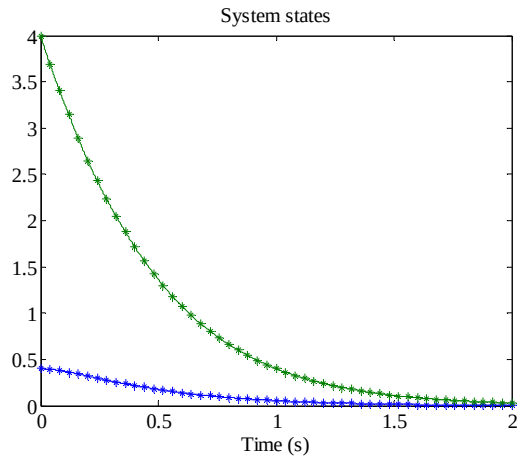
Select quadratic NN basis set for VFA

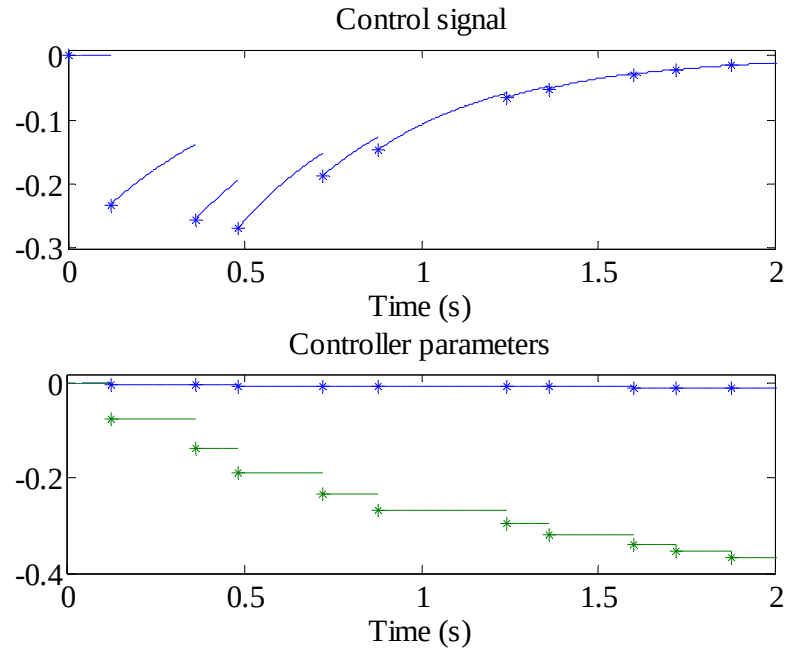
Exact solution $\quad W_1^* = [p_{11} \quad 2p_{12} \quad 2p_{13} \quad p_{22} \quad 2p_{23} \quad p_{33}]^T$

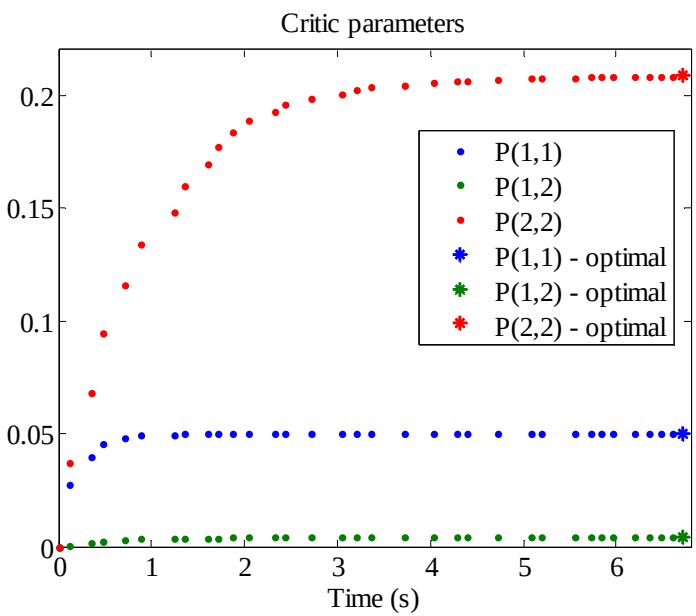
$$= [1.4245 \quad 1.1682 \quad -0.1352 \quad 1.4349 \quad -0.1501 \quad 0.4329]^T$$

Simulations on: F-16 autopilot

A matrix not needed







Converge to SS Riccati equation soln

Solves ARE online without knowing A

$$0 = PA + A^T P + Q - PBR^{-1}B^T P$$

A. Al-Tamimi, D. Vrabie, Youyi Wang

Simulation 2: Load Frequency Control of Electric Power system

$$\dot{x} = Ax + Bu$$

$$x(t) = [\Delta f(t) \quad \Delta P_g(t) \quad \Delta X_g(t) \quad \Delta E(t)]^T$$

Frequency
Generator output
Governor position
Integral control

$$A = \begin{bmatrix} -1/T_p & K_p/T_p & 0 & 0 \\ 0 & -1/T_T & 1/T_T & 0 \\ -1/RT_G & 0 & -1/T_G & -1/T_G \\ K_E & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1/T_G \\ 0 \end{bmatrix}$$

ARE

$$0 = PA + A^T P + Q - PBR^{-1}B^T P$$

ARE solution using full dynamics model (A,B)

$$P_{ARE} = \begin{bmatrix} 0.4750 & 0.4766 & 0.0601 & 0.4751 \\ 0.4766 & 0.7831 & 0.1237 & 0.3829 \\ 0.0601 & 0.1237 & 0.0513 & 0.0298 \\ 0.4751 & 0.3829 & 0.0298 & 2.3370 \end{bmatrix}.$$

$$0 = PA + A^T P + Q - PBR^{-1}B^T P$$

Solves ARE online without knowing A

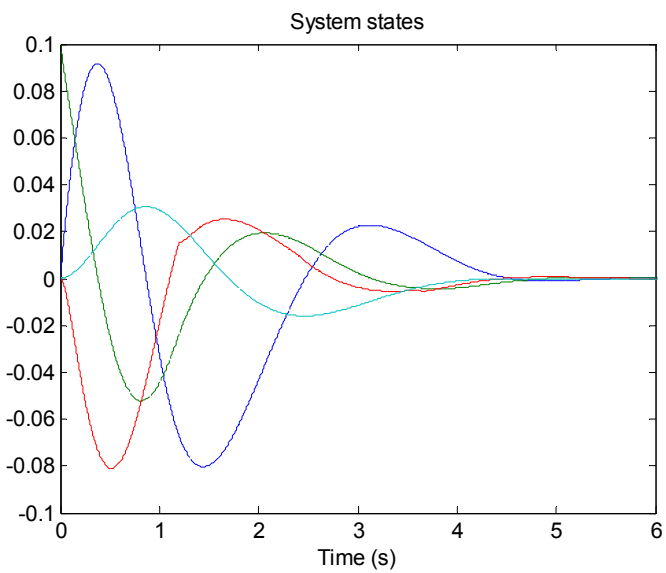
$$P_{ARE} = \begin{bmatrix} 0.4750 & 0.4766 & 0.0601 & 0.4751 \\ 0.4766 & 0.7831 & 0.1237 & 0.3829 \\ 0.0601 & 0.1237 & 0.0513 & 0.0298 \\ 0.4751 & 0.3829 & 0.0298 & 2.3370 \end{bmatrix}.$$

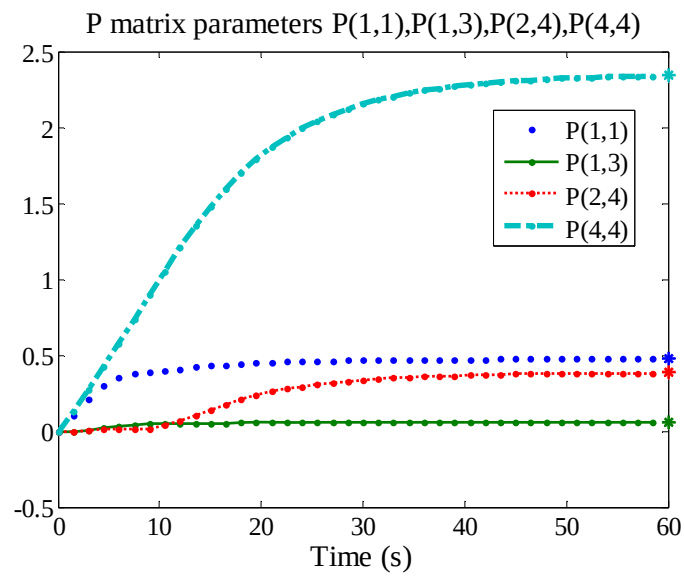
$$P_{critic\ NN} = \begin{bmatrix} 0.4802 & 0.4768 & 0.0603 & 0.4754 \\ 0.4768 & 0.7887 & 0.1239 & 0.3834 \\ 0.0603 & 0.1239 & 0.0567 & 0.0300 \\ 0.4754 & 0.3843 & 0.0300 & 2.3433 \end{bmatrix}.$$

IRL period of T= 0.1s.

Fifteen data points $\left(x(t), x(t+T), \rho(t:t+T)\right)$

Hence, the value estimate was updated every 1.5s.





The Power of Optimal Design

Once you can do optimal design that minimizes a performance index, many sorts of designs are immediately possible.

Minimum energy

$$J = \frac{1}{2}\int_0^\infty x^T Q x + u^T R u \, dt$$

Minimum fuel

$$J = \frac{1}{2}\int_0^\infty x^T Q x + \rho |u| \, dt$$

Minimum time

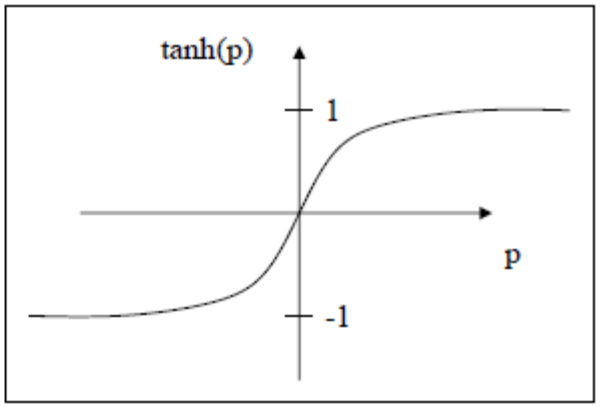
$$J = \int_0^T 1 \, dt = T$$

Constrained control inputs

$$J = \frac{1}{2}\int_0^\infty \left(Q(x) + \int_0^u \sigma^{-1}(v) dv \right) dt$$

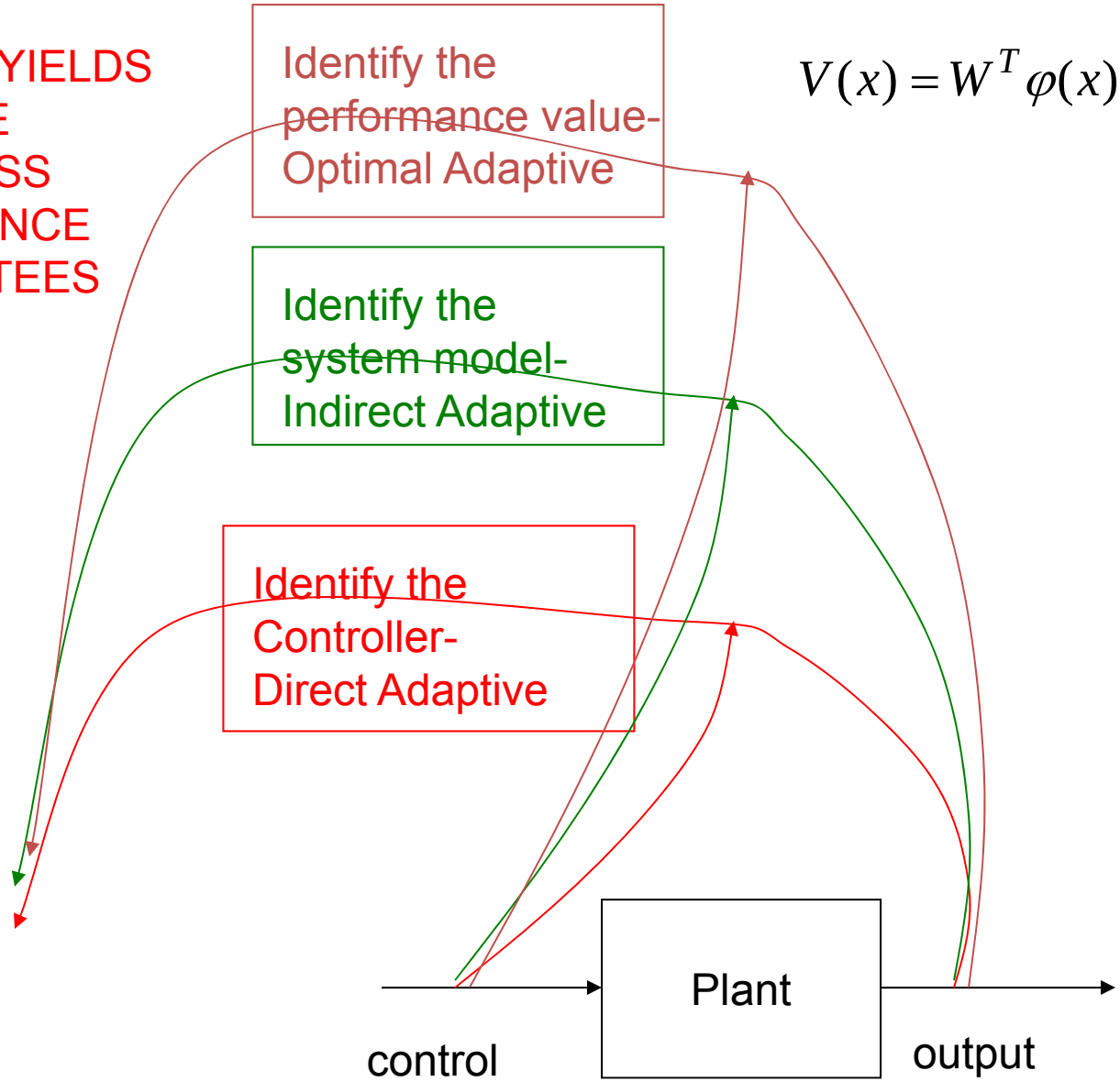
Approximate minimum time with smooth control inputs

$$J = \frac{1}{2}\int_0^\infty \left(\tanh(x^T Q x) + \rho \int_0^u \sigma^{-1}(v) dv \right) dt$$



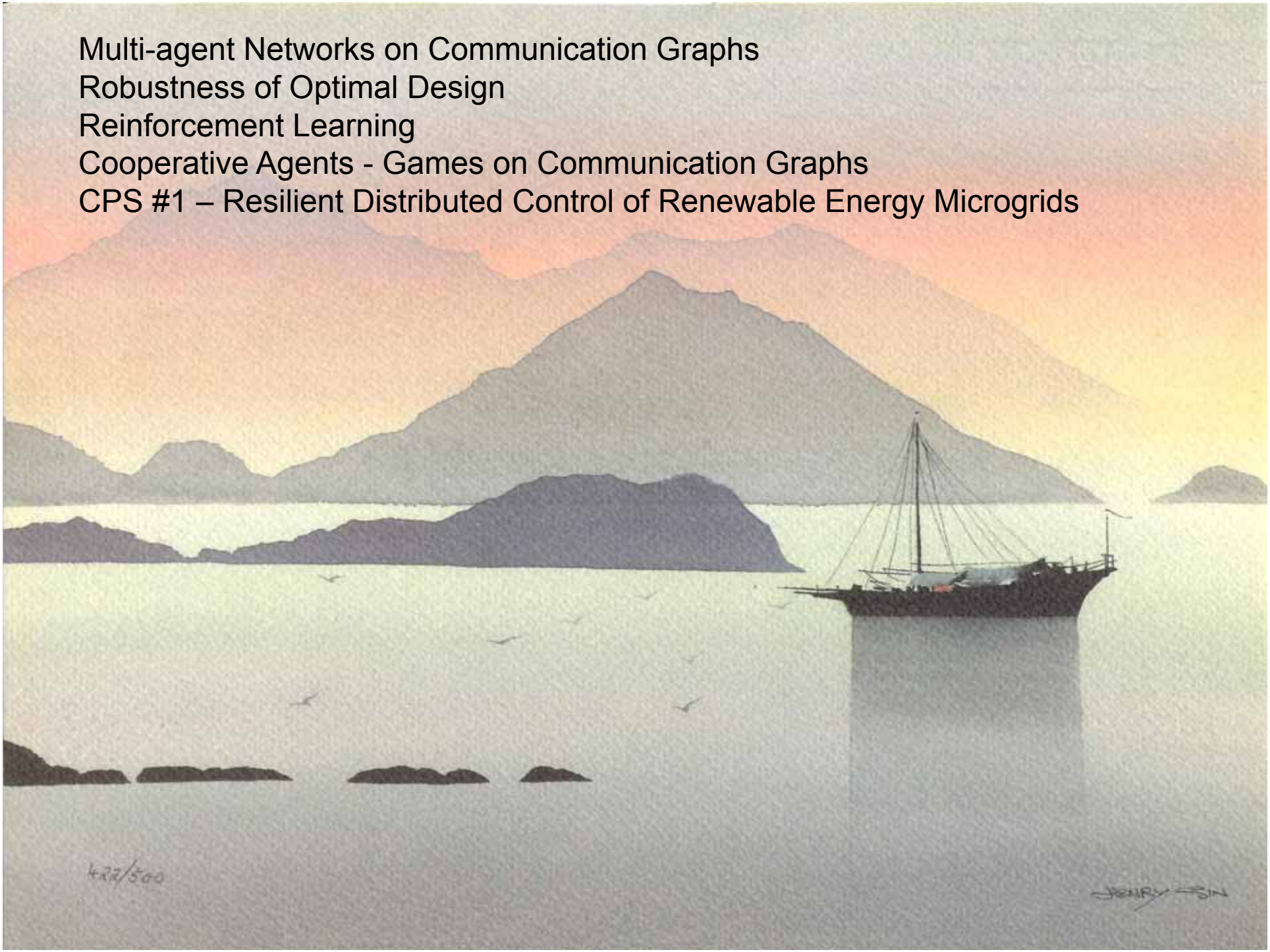
Adaptive Control

OPTIMALITY YIELDS
RESILIENCE
ROBUSTNESS
PERFORMANCE
GUARANTEES

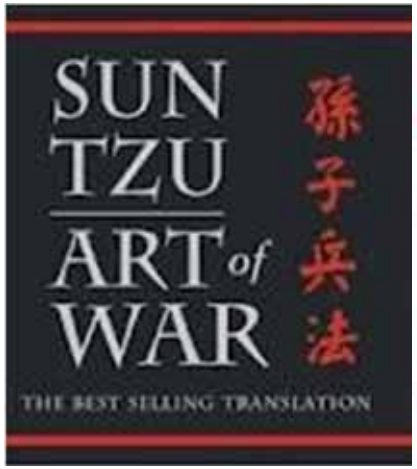


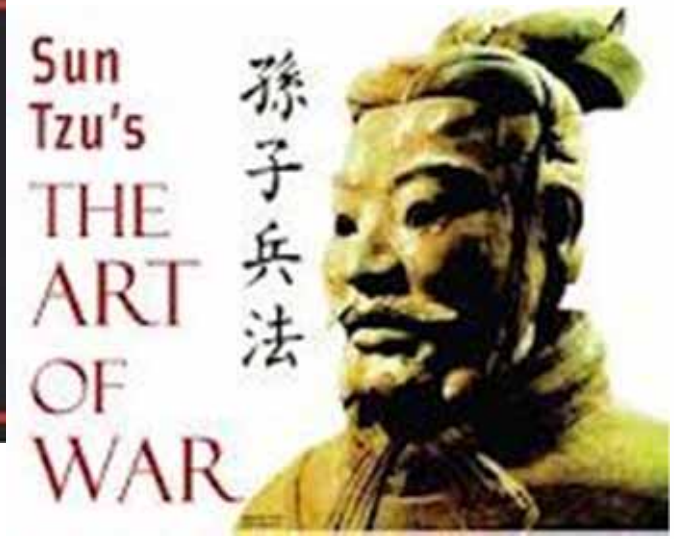
$$V(x) = W^T \varphi(x)$$

Multi-agent Networks on Communication Graphs
Robustness of Optimal Design
Reinforcement Learning
Cooperative Agents - Games on Communication Graphs
CPS #1 – Resilient Distributed Control of Renewable Energy Microgrids



Games on Communication Graphs





500 BC

孙**子兵法**

Sun Tz bin fa

Speed up Multi-agent Decision-making by Imposing Sparse Efficient Communication Structures

Fast Decision Using
New Concept of Graphical Games to limit each agent's decision horizon

Bounded Rationality -
- Faster decisions based on local information
- Under certain conditions, the local decisions are globally optimal
- Nash Equilibrium on Graphs is defined

Automatica 50 (2014) 3038–3053

Contents lists available at ScienceDirect

Automatica

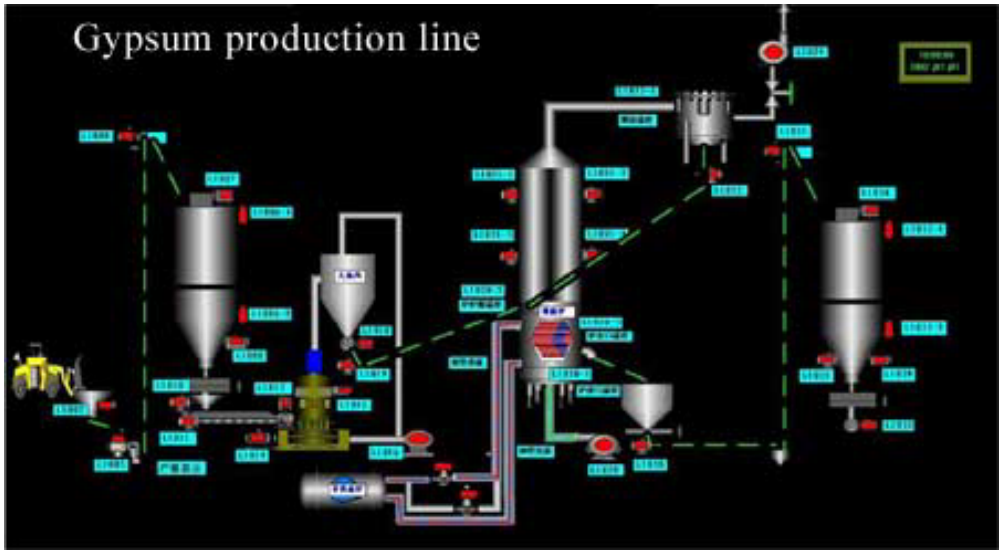
journal homepage: www.elsevier.com/locate/automatica

Multi-agent discrete-time graphical games and reinforcement learning solutions☆

Mohammed I. Abouheaf [a,1], Frank L. Lewis [b], Kyriakos G. Vamvoudakis [c], Sofie Haesaert [d], Robert Babuska [e]

Manufacturing as the Interactions of Multiple Agents

Each machine has it own dynamics and cost function
Neighboring machines influence each other most strongly
There are local optimization requirements as well as global necessities



Each process has its own dynamics $\dot{\delta}_i = A\delta_i + (d_i + g_i)B_i u_i - \sum_{j \in N_i} e_{ij} B_j u_j$

And cost function $J_i(\delta_i(0), u_i, u_{-i}) = \frac{1}{2}\int_0^\infty (\delta_i^T Q_{ii} \delta_i + u_i^T R_{ii} u_i + \sum_{j \in N_i} u_j^T R_{ij} u_j)\, dt$

Each process helps other processes achieve optimality and efficiency

F.L. Lewis, H. Zhang, A. Das, K. Hengster-Movric, *Cooperative Control of Multi-Agent Systems: Optimal Design and Adaptive Control*, Springer-Verlag, 2014

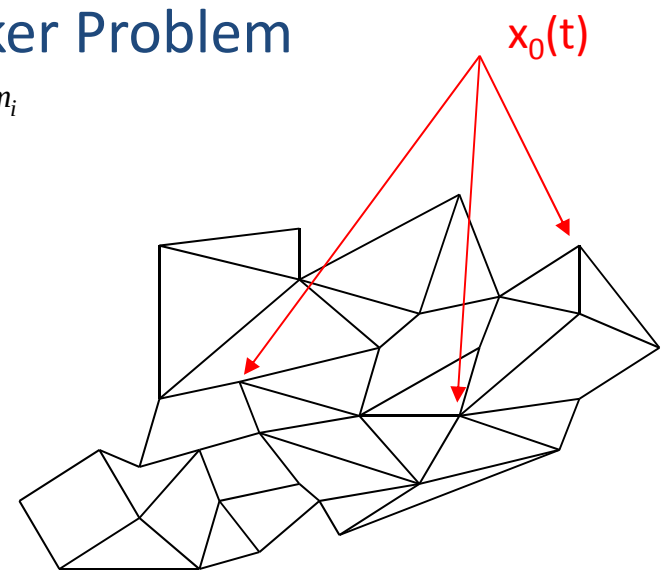
Key Point

Lyapunov Functions and Performance Indices
Must depend on graph topology

H. Zhang, F.L. Lewis, and Z. Qu, "Lyapunov, Adaptive, and Optimal Design Techniques for Cooperative Systems on Directed Communication Graphs," IEEE Trans. Industrial Electronics, vol. 59, no. 7, pp. 3026-3041, July 2012.

Hongwei Zhang, F.L. Lewis, and Abhijit Das
"Optimal design for synchronization of cooperative systems: state feedback, observer and output feedback," IEEE Trans. Automatic Control, vol. 56, no. 8, pp. 1948-1952, August 2011.

Graphical Games
Synchronization- Cooperative Tracker Problem



Node dynamics $\dot{x}_i = Ax_i + B_i u_i, \quad x_i(t) \in \mathbb{R}^n, \quad u_i(t) \in \mathbb{R}^{m_i}$

Target generator dynamics $\dot{x}_0 = Ax_0$

Synchronization problem $x_i(t) \to x_0(t), \forall i$

Local neighborhood tracking error (Lihua Xie)

$$\delta_i = \sum_{j \in N_i} e_{ij}(x_i - x_j) \quad + g_i(x_i - x_0),$$

Local nbhd. tracking error dynamics

$$\dot{\delta}_i = A\delta_i + (d_i + g_i)B_i u_i - \sum_{j \in N_i} e_{ij} B_j u_j$$

Local agent dynamics driven by neighbors' controls

Impose Optimality to get Resilience and Robustness

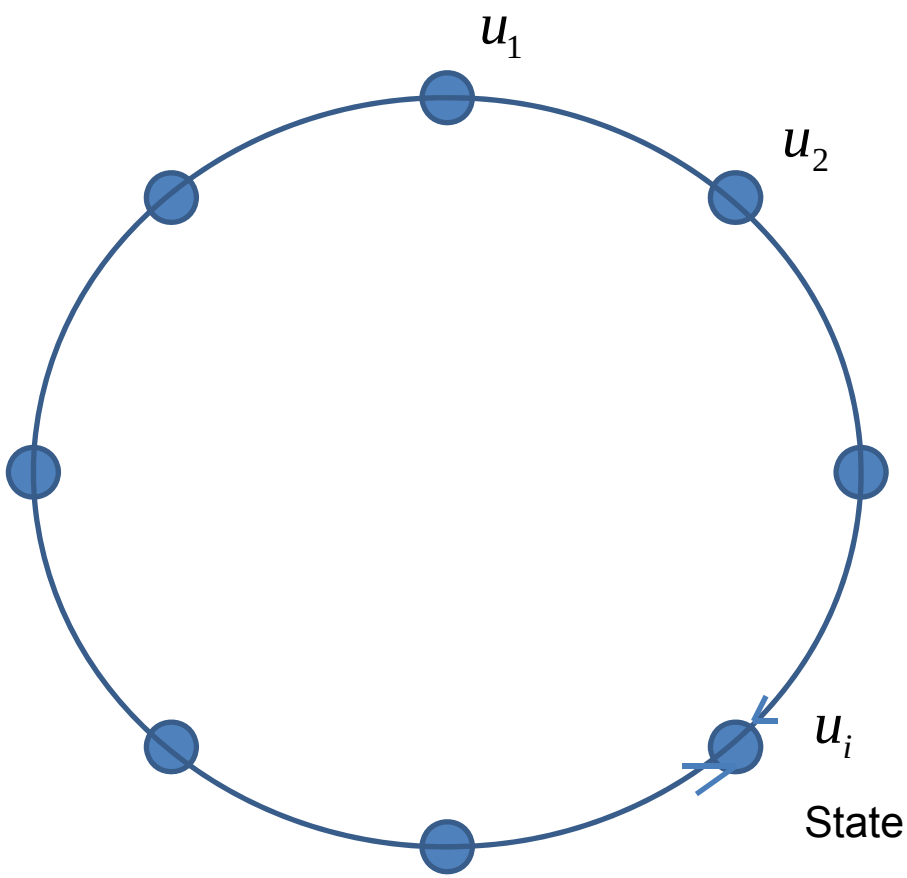
Define Local nbhd. performance index

$$J_i(\delta_i(0), u_i, u_{-i}) = \tfrac{1}{2}\int_0^\infty (\delta_i^T Q_{ii}\delta_i + u_i^T R_{ii} u_i + \sum_{j \in N_i} u_j^T R_{ij} u_j)\, dt \quad \equiv \tfrac{1}{2}\int_0^\infty L_i(\delta_i(t), u_i(t), u_{-i}(t))\, dt$$

Values driven by neighbors' controls

K.G. Vamvoudakis, F.L. Lewis, and G.R. Hudas, "Multi-Agent Differential Graphical Games: online adaptive learning solution for synchronization with optimality," Automatica, vol. 48, no. 8, pp. 1598-1611, Aug. 2012.

New Differential Graphical Game



Local Dynamics
Local Value Function
Only depends on
graph neighbors

u_i Control action of player *i*

State dynamics of agent *i*

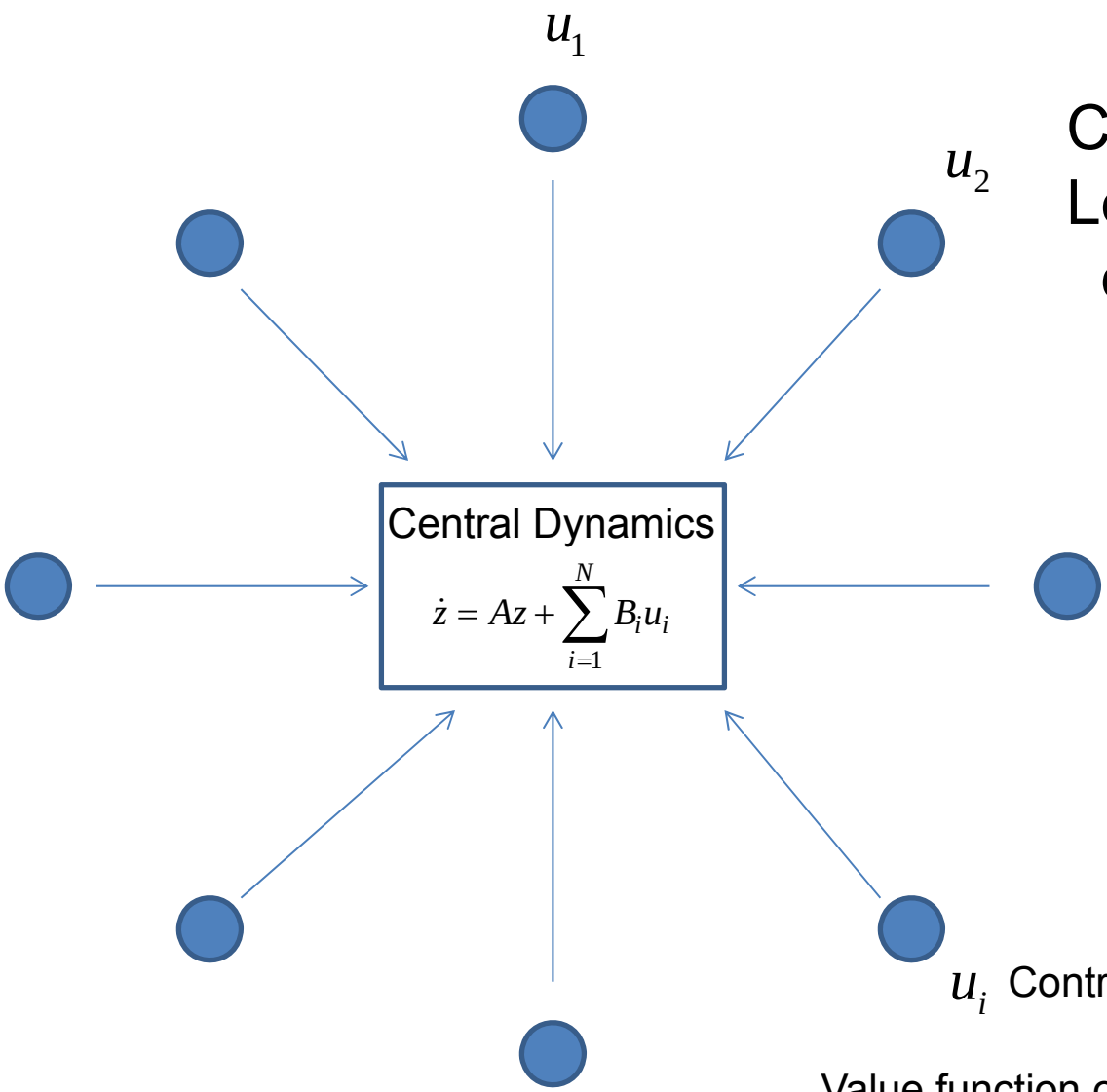
$$\dot{\delta}_i = A\delta_i + (d_i + g_i)B_i u_i - \sum_{j \in N_i} e_{ij} B_j u_j$$

Value function of player *i*

$$J_i(\delta_i(0), u_i, u_{-i}) = \tfrac{1}{2}\int_0^\infty (\delta_i^T Q_{ii}\delta_i + u_i^T R_{ii} u_i + \sum_{j \in N_i} u_j^T R_{ij} u_j)\, dt$$

OPTIMAL CONTROL IS
RESILIENT
ROBUST
HAS GUARANTEED STABILITY MARGINS

Standard Multi-Agent Differential Game



Central Dynamics
Local Value Function
depends on ALL
other control actions

u_i Control action of player *i*

Value function of player *i*

$$J_i(z(0), u_i, u_{-i}) = \tfrac{1}{2}\int_0^\infty (z^T Q z + \sum_{j=1}^{N} u_j^T R_{ij} u_j)\, dt$$

Team Interest vs. Self Interest

Cooperation vs. Collaboration

The objective functions of each player can be written as a *team average* term plus a *conflict of interest* term:

$$J_1 = \tfrac{1}{3}(J_1 + J_2 + J_3) + \tfrac{1}{3}(J_1 - J_2) + \tfrac{1}{3}(J_1 - J_3) \equiv J_{team} + J_1^{coi}$$

$$J_2 = \tfrac{1}{3}(J_1 + J_2 + J_3) + \tfrac{1}{3}(J_2 - J_1) + \tfrac{1}{3}(J_2 - J_3) \equiv J_{team} + J_2^{coi}$$

$$J_3 = \tfrac{1}{3}(J_1 + J_2 + J_3) + \tfrac{1}{3}(J_3 - J_1) + \tfrac{1}{3}(J_3 - J_2) \equiv J_{team} + J_3^{coi}$$

For N-players

$$J_i = \tfrac{1}{N}\sum_{j=1}^{N} J_j + \tfrac{1}{N}\sum_{j=1}^{N}(J_i - J_j) \equiv J_{team} + J_i^{coi}, \quad i = 1, N$$

For *N-player zero-sum games*, the first term is zero, i.e. the players have no goals in common.

New Definition of Nash Equilibrium for Graphical Games

To restore the Symmetry of Nash equilibrium

Def: Interactive Nash equilibrium

$\{u_1^*, u_2^*, \ldots, u_N^*\}$ are in Interactive Nash equilibrium if

1. $J_i^* \triangleq J_i(u_i^*, u_{G-i}^*) \le J_i(u_i, u_{G-i}^*), \ \forall i \in N$ i.e. they are in Nash equilibrium

2. There exists a policy u_j such that

$$J_i(u_j, u_{G-j}^*) \ne J_i(u_j^*, u_{G-j}^*), \quad \forall i, j \in N$$

That is, every player can find a policy that changes the value of every other player.

Theorem 3. Let *(A,B$_i$)* be reachable for all *i*.
Let agent *i* be in local best response

$$J_i(u_i^*, u_{-i}) \le J_i(u_i, u_{-i}), \quad \forall i$$

Then $\{u_1^*, u_2^*, \ldots, u_N^*\}$ are in global Interactive Nash iff the graph is strongly connected.

Graphical Game Solution Using Integral Reinforcement Learning

Value function

$$V_i(\delta_i(t)) = \tfrac{1}{2}\int_t^{\infty} (\delta_i^T Q_{ii}\delta_i + u_i^T R_{ii} u_i + \sum_{j\in N_i} u_j^T R_{ij} u_j)\, dt$$

Differential equivalent (Leibniz formula) is Bellman's Equation

$$H_i(\delta_i, \frac{\partial V_i}{\partial \delta_i}, u_i, u_{-i}) \equiv \frac{\partial V_i}{\partial \delta_i}^T \left(A\delta_i + (d_i + g_i)B_i u_i - \sum_{j\in N_i} e_{ij} B_j u_j \right) + \tfrac{1}{2}\delta_i^T Q_{ii}\delta_i + \tfrac{1}{2} u_i^T R_{ii} u_i + \tfrac{1}{2}\sum_{j\in N_i} u_j^T R_{ij} u_j = 0$$

Stationarity Condition

$$0 = \frac{\partial H_i}{\partial u_i} \quad \Rightarrow \quad u_i = -(d_i + g_i) R_{ii}^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i}$$

To find online methods for GG

Iterate on these two equations

1. Coupled HJ equations

$$\frac{\partial V_i}{\partial \delta_i}^T A_i^c + \tfrac{1}{2}\delta_i^T Q_{ii}\delta_i + \tfrac{1}{2}(d_i + g_i)^2 \frac{\partial V_i}{\partial \delta_i}^T B_i R_{ii}^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i} + \tfrac{1}{2}\sum_{j\in N_i} (d_j + g_j)^2 \frac{\partial V_j}{\partial \delta_j}^T B_j R_{jj}^{-1} R_{ij} R_{jj}^{-1} B_j^T \frac{\partial V_j}{\partial \delta_j} = 0, i \in N$$

$$H_i(\delta_i, \frac{\partial V_i}{\partial \delta_i}, u_i^*, u_{-i}^*) = 0$$

where $$A_i^c = A\delta_i - (d_i + g_i)^2 B_i R_{ii}^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i} + \sum_{j\in N_i} e_{ij}(d_j + g_j) B_j R_{jj}^{-1} B_j^T \frac{\partial V_j}{\partial \delta_j}, i \in N$$

---

2. Best Response HJ Equations – other players have fixed policies u_j

$$0 = H_i(\delta_i, \frac{\partial V_i}{\partial \delta_i}, u_i^*, u_{-i}) \equiv \frac{\partial V_i}{\partial \delta_i}^T A_i^c + \tfrac{1}{2}\delta_i^T Q_{ii}\delta_i + \tfrac{1}{2}(d_i + g_i)^2 \frac{\partial V_i}{\partial \delta_i}^T B_i R_{ii}^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i} + \tfrac{1}{2}\sum_{j\in N_i} u_j^T R_{ij} u_j$$

where $$A_i^c = A\delta_i - (d_i + g_i)^2 B_i R_{ii}^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i} - \sum_{j\in N_i} e_{ij} B_j u_j$$

Online Solution of Graphical Games

Kyriakos Vamvoudakis

Use Reinforcement Learning

POLICY ITERATION

Algorithm 1. Policy Iteration (PI) Solution for N-player distributed games.

Step 0: Start with admissible initial policies $u_i^0, \forall i$.

Step 1: (Policy Evaluation) Solve for V_i^k using (14)

$$H_i(\delta_i, \frac{\partial V_i^k}{\partial \delta_i}, u_i^k, u_{-i}^k) = 0, \forall i = 1, \ldots, N \qquad (38)$$

Step 2: (Policy Improvement) Update the N-tuple of control policies using

$$u_i^{k+1} = \arg\min_{u_i} H_i(\delta_i, \frac{\partial V_i^k}{\partial \delta_i}, u_i, u_{-i}^k), \forall i = 1, \ldots, N$$

which explicitly is

$$u_i^{k+1} = -(d_i + g_i) R_{ii}^{-1} B_i^T \frac{\partial V_i^k}{\partial \delta_i}, \forall i = 1, \ldots, N. \qquad (39)$$

Go to step 1.

On convergence End ■

MULTI-AGENT LEARNING

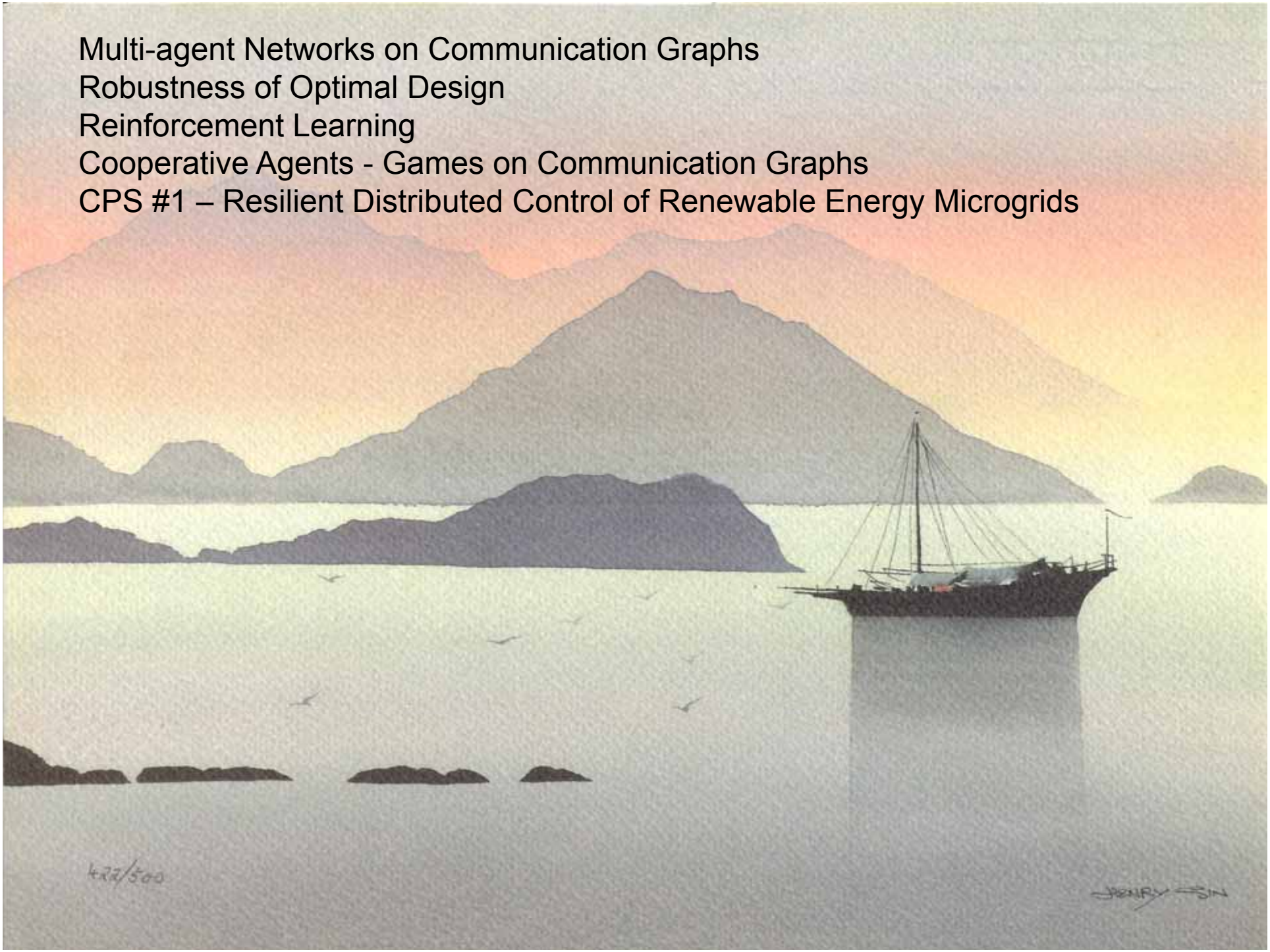
Convergence Results

Theorem 3. Convergence of Policy Iteration algorithm when only i^{th} agent updates its policy and all players u_{-i} in the neighborhood do not change. Given fixed neighbors policies u_{-i}, assume there exists an admissible policy u_i. Assume that agent i performs Algorithm 1 and the its neighbors do not update their control policies. Then the algorithm converges to the best response u_i to policies u_{-i} of the neighbors and to the solution V_i to the best response HJ equation (36).

The next result concerns the case where all nodes update their policies at each step of the algorithm. Define the relative control weighting as $\rho_{ij} = \bar{\sigma}(R_{jj}^{-1} R_{ij})$, where $\bar{\sigma}(R_{jj}^{-1} R_{ij})$ is the maximum singular value of $R_{jj}^{-1} R_{ij}$.

Theorem 4. Convergence of Policy Iteration algorithm when all agents update their policies. Assume all nodes i update their policies at each iteration of PI. Then for small enough edge weights e_{ij} and ρ_{ij}, μ_i converges to the global Nash equilibrium and for all i, and the values converge to the optimal game values $V_i^k \to V_i^*$.

Multi-agent Networks on Communication Graphs
Robustness of Optimal Design
Reinforcement Learning
Cooperative Agents - Games on Communication Graphs
CPS #1 – Resilient Distributed Control of Renewable Energy Microgrids



What is a micro-grid?

- Micro-grid is a small-scale power system that provides the power for a group of consumers.
- Micro-grid enables local power support for local and critical loads.
- Micro-grid has the ability to work in both grid-connected and islanded modes.
- Micro-grid facilitates the integration of Distributed Energy Resources (DER).

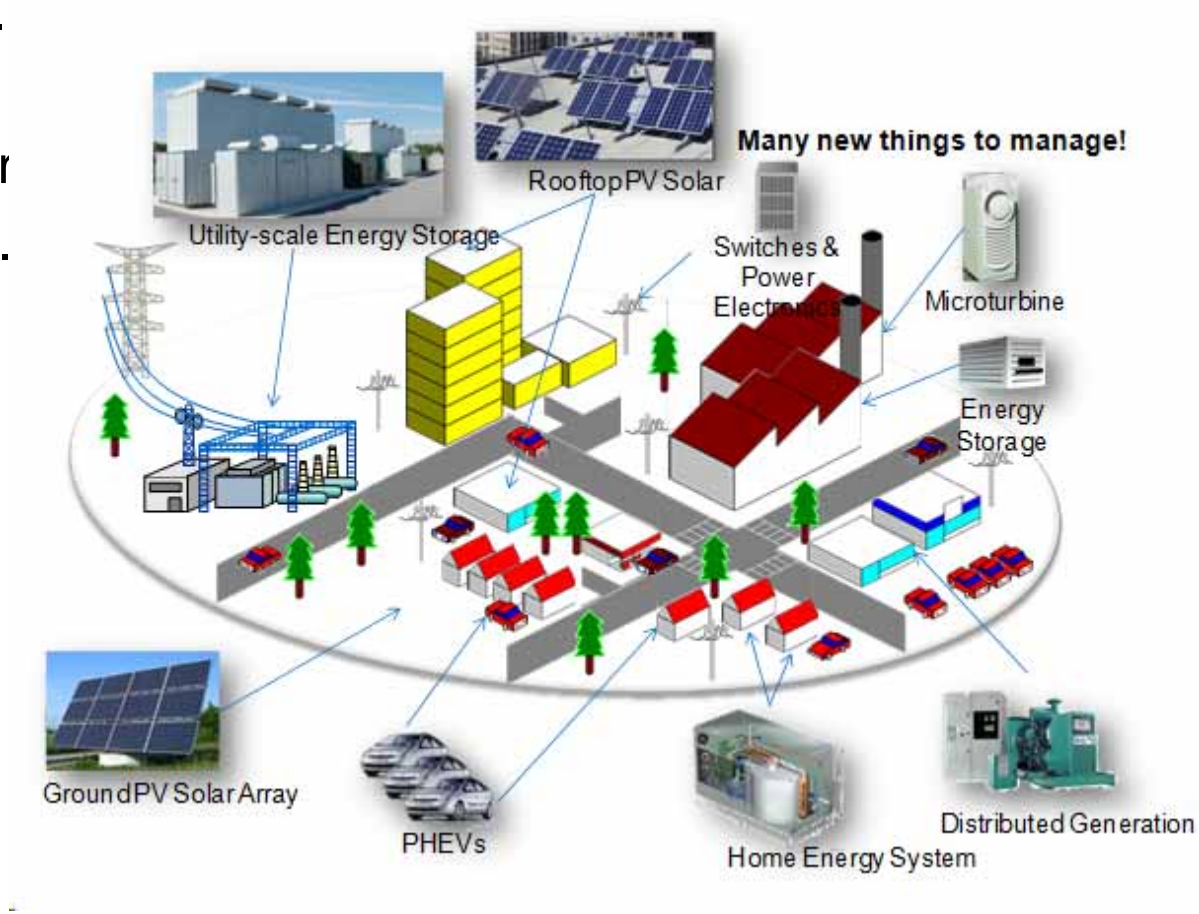
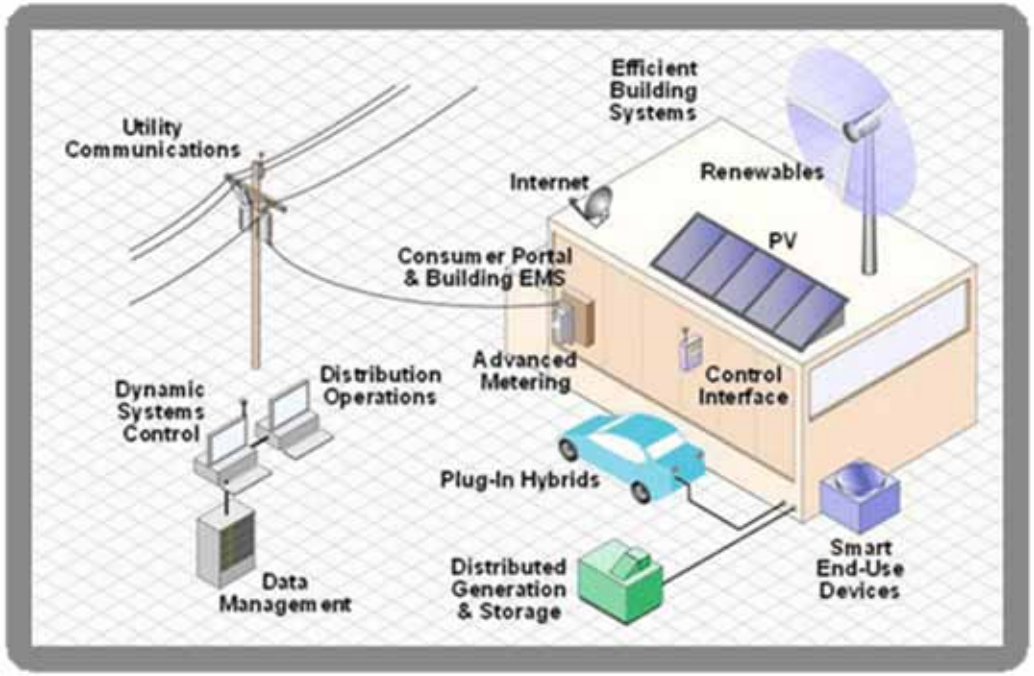


Photo from: http://www.horizonenergygroup.com

An introduction to micro-grids: Micro-grid applications

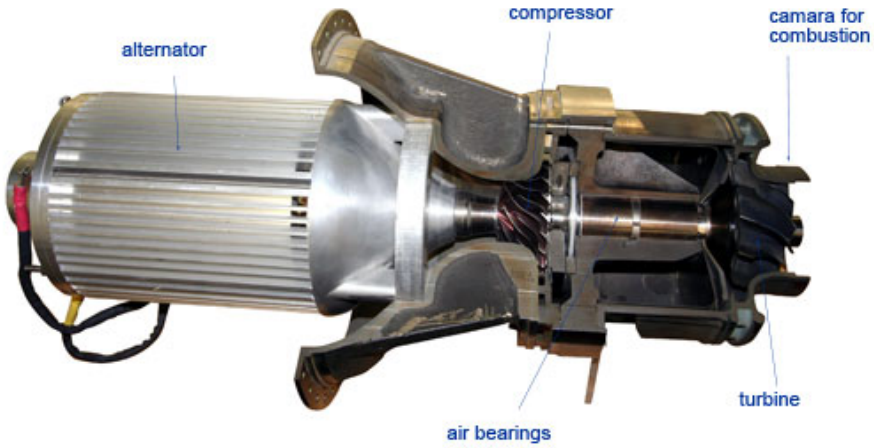
- The main building block of smart-grids
- Rural plants
- Business buildings, hospitals, and factories



Smart-grid photo from: http://www.sustainable-sphere.com

Distributed Generators (DG)
Distributed Energy Resources (DER)

- Non-renewables
  - Internal combustion engine
  - Micro-turbines
  - Fuel cells
- Renewables
  - Photovoltaic
  - Wind
  - Hydroelectric
  - Biomass

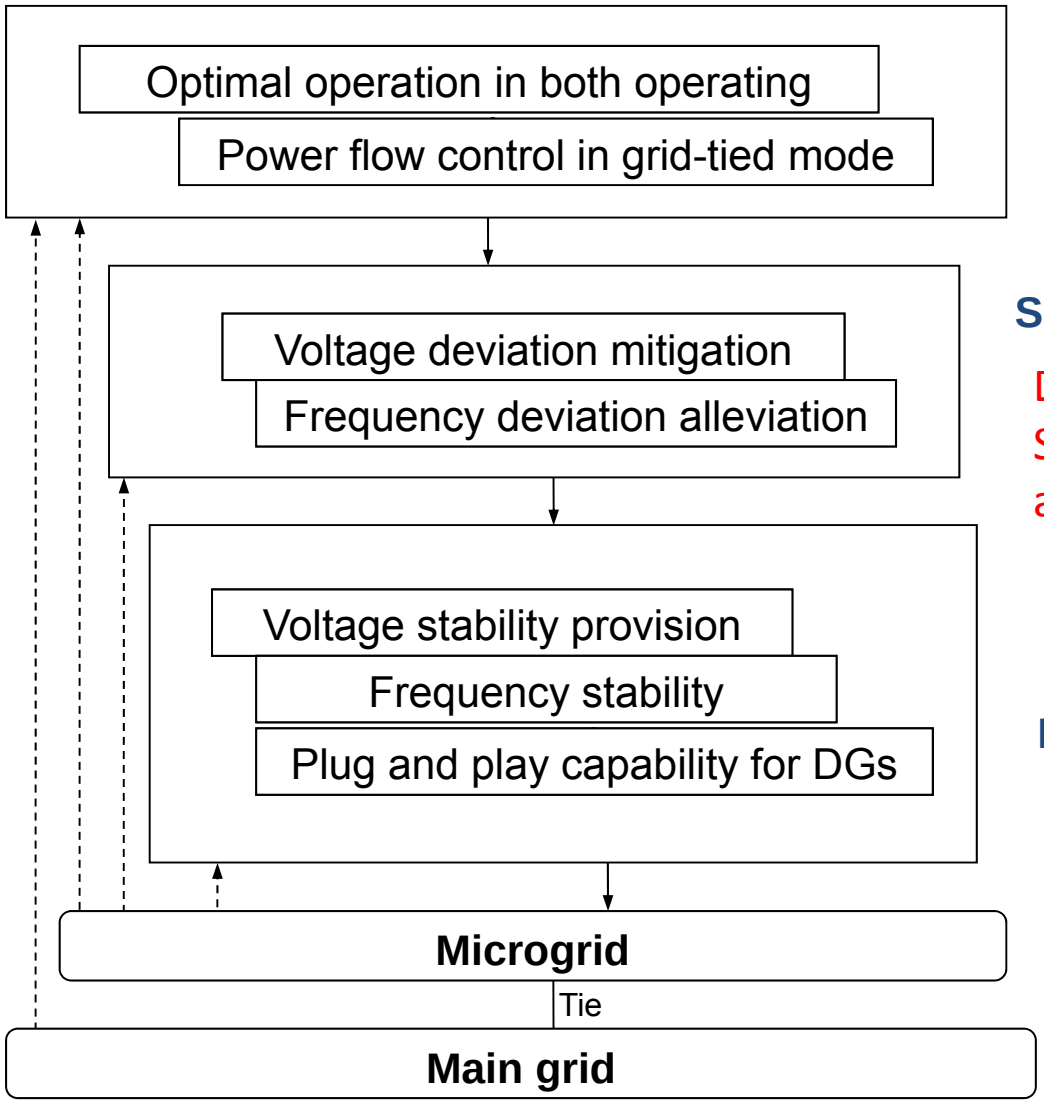


Micro-grid Advantages

- Micro-grid provides high quality and reliable power to the critical consumers
- During main grid disturbances, micro-grid can quickly disconnect form the main grid and provide reliable power for its local loads
- DGs can be simply installed close to the loads which significantly reduces the power transmission line losses
- By using renewable energy resources, a micro-grid reduces CO2 emissions

Bidram, A., & Davoudi, A. (2012). Hierarchical structure of microgrids control system. *IEEE Transactions on Smart Grid*, vol. 3, pp. 1963-1976, Dec 2012.

Micro-grid Hierarchical Control Structure



Tertiary Control

Secondary Control

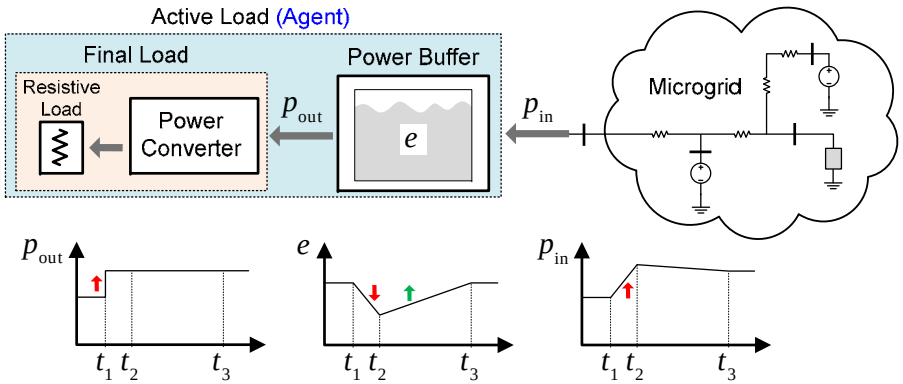
Do coop. ctrl. here to Synchronize frequency and voltage

Primary Control

Maintains Stability

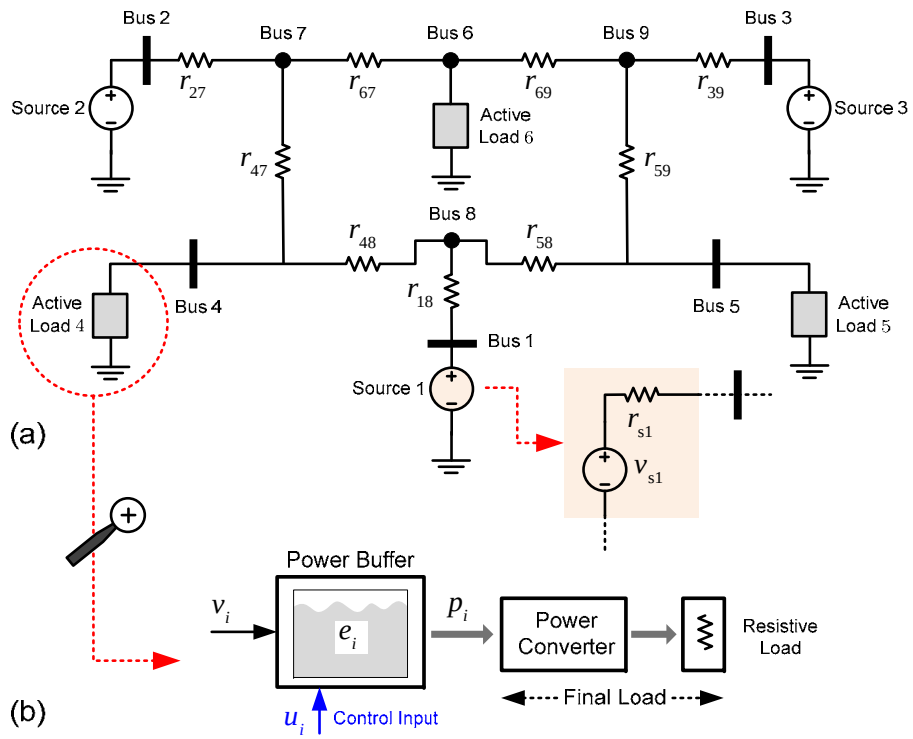
Cooperative Game-theoretic Control of Active Loads in DC Microgrids

Ling-ling Fan, Vahidreza Nasirian, Hamidreza Modares,
Frank L. Lewis, Yong-duan Song, and Ali Davoudi,



Power buffer operation during a step change in power demand.

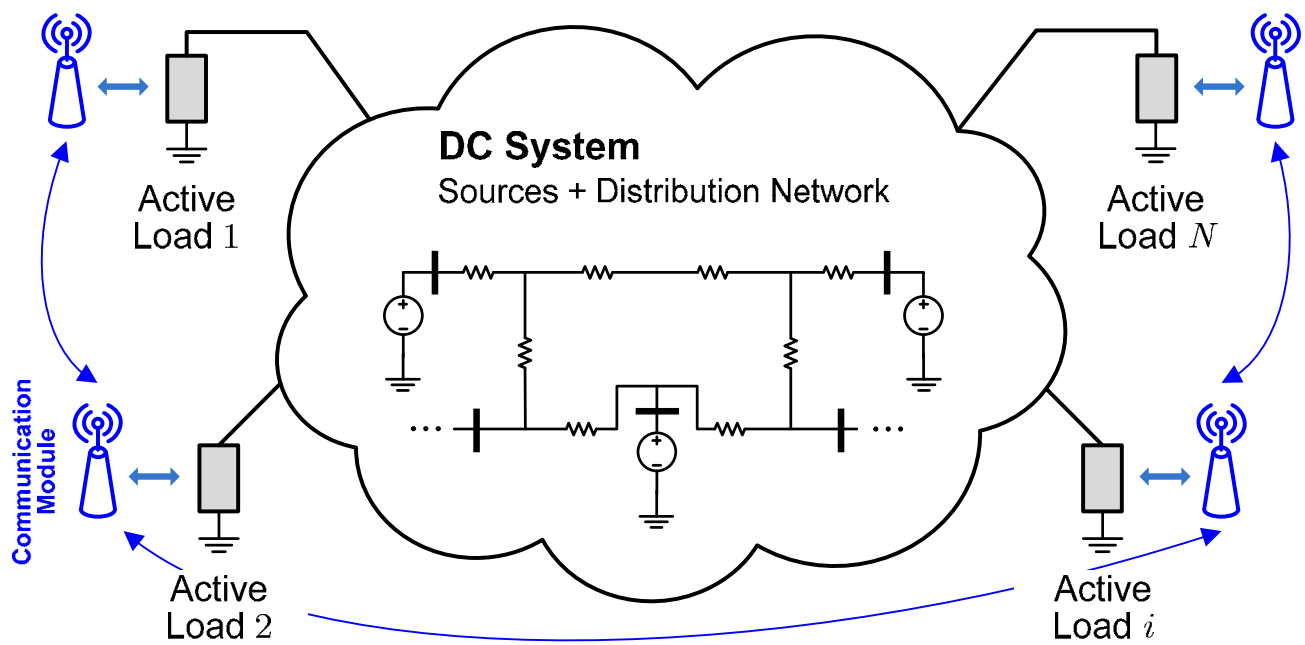
Power buffers in Microgrid Network



Active Load Power Buffer

$$\begin{cases} \dot{e}_i = \dfrac{v_i^2}{r_i} - p_i \\ \dot{r}_i = u_i \end{cases},$$

Stored energy e_i
Input impedance r_i
Bus voltage v_i
Control input u_i
Output power = a disturbance p_i



Solve for bus voltage to get coupled agent dynamics

Vahid Nasirian
Reza Modares

Dr. Ali Davoudi

$$\underbrace{\begin{bmatrix}\dot{e}_i\\ \dot{r}_i\\ \dot{p}_i\end{bmatrix}}_{\dot{\mathrm{x}}_i}=\underbrace{\begin{bmatrix}0 & 2i_i^{\mathrm{q}}\gamma_{ii}-\left(i_i^{\mathrm{q}}\right)^2 & -1\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}}_{\mathrm{A}_i}\underbrace{\begin{bmatrix}e_i\\ r_i\\ p_i\end{bmatrix}}_{\mathrm{x}_i}+\underbrace{\begin{bmatrix}0\\1\\0\end{bmatrix}}_{\mathrm{B}_i}u_i+\underbrace{\begin{bmatrix}0\\0\\1\end{bmatrix}}_{\mathrm{D}_i}w_i+$$

$$+2i_i^{\mathrm{q}}\begin{bmatrix}\sum_{j=M+1(\neq i)}^{M+N}\gamma_{ij}r_j\\ 0\\ 0\end{bmatrix},\qquad i=M+1,\cdots,M+N,$$

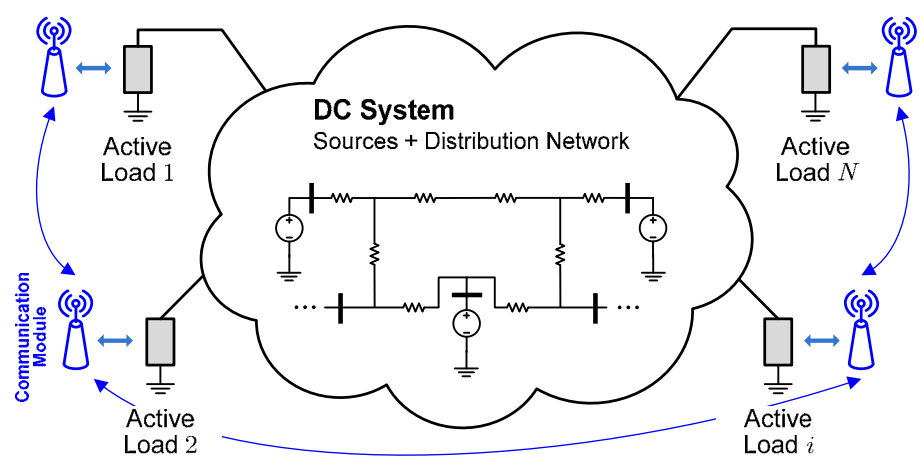
Define coupled performance indices

$$J_i=\int_0^\infty\left(\sum_{j\in\bar{N}_i}\mathrm{x}_j^{\mathsf{T}}\mathrm{Q}_{ij}\mathrm{x}_j+\rho_i u_i^2\right)\mathrm{d}t,\qquad i=M+1,\cdots,M+N,$$

Define Communication Graph

Sparse efficient topology

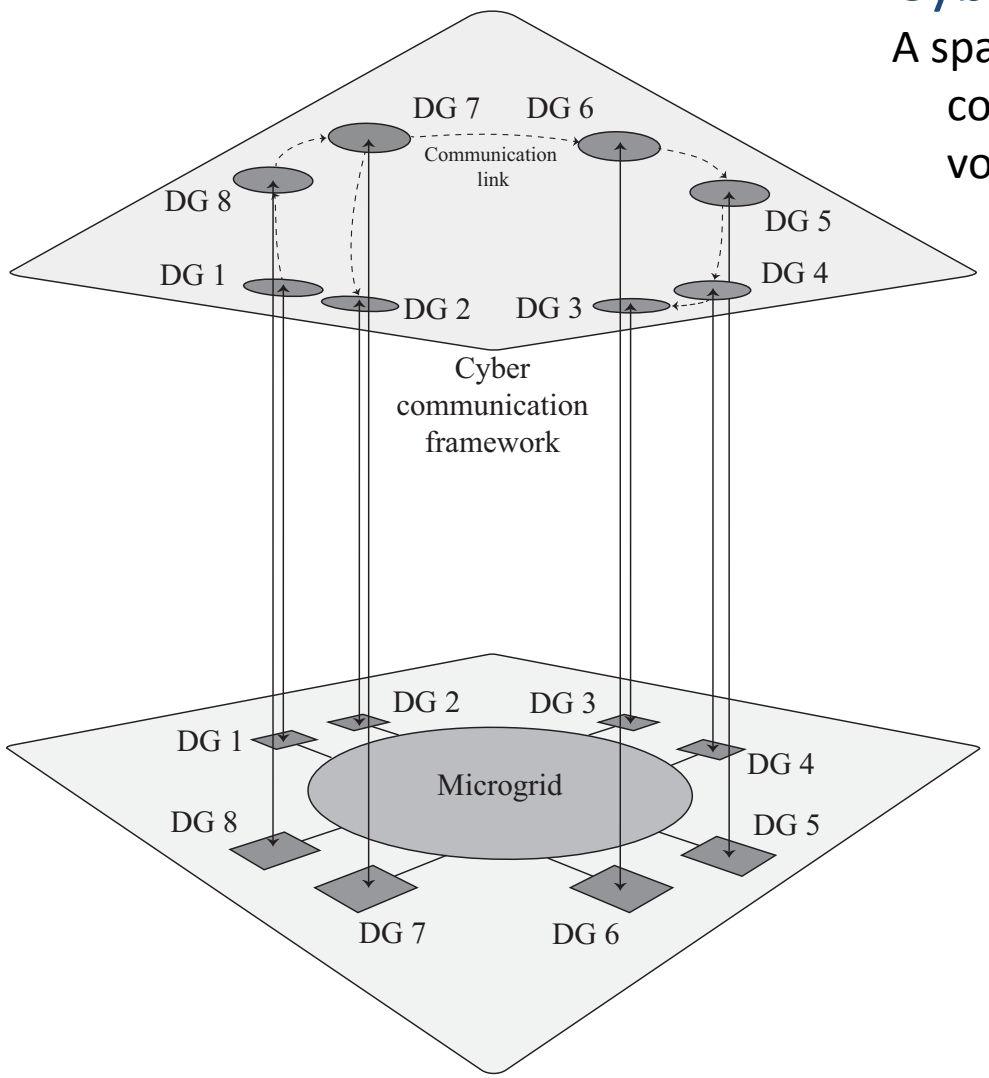
Optimal design provides Resilience and disturbance rejection



Micro-grid secondary control: Distributed CPS structure

Work of Ali Bidram
With Dr. A. Davoudi

Cyber Physical System (CPS)



Cyber layer

A sparse, efficient communication network to allow cooperative control for synchronization of voltage and frequency

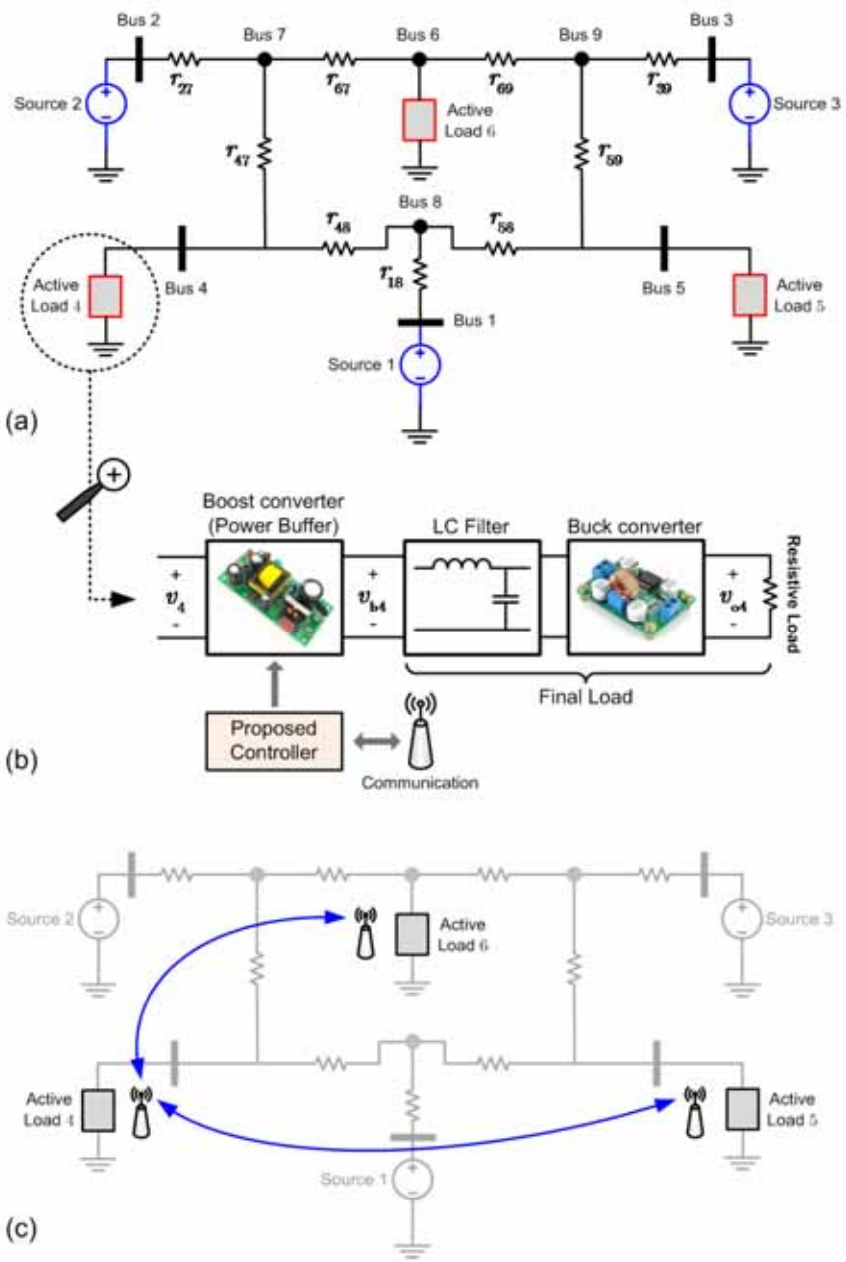
Resilience Because of-
Sparse, Efficient communication network
Optimal Design- Graph Games
Distributed computation – no SPOF

Physical Layer

The interconnect structure of the power grid

Case Study

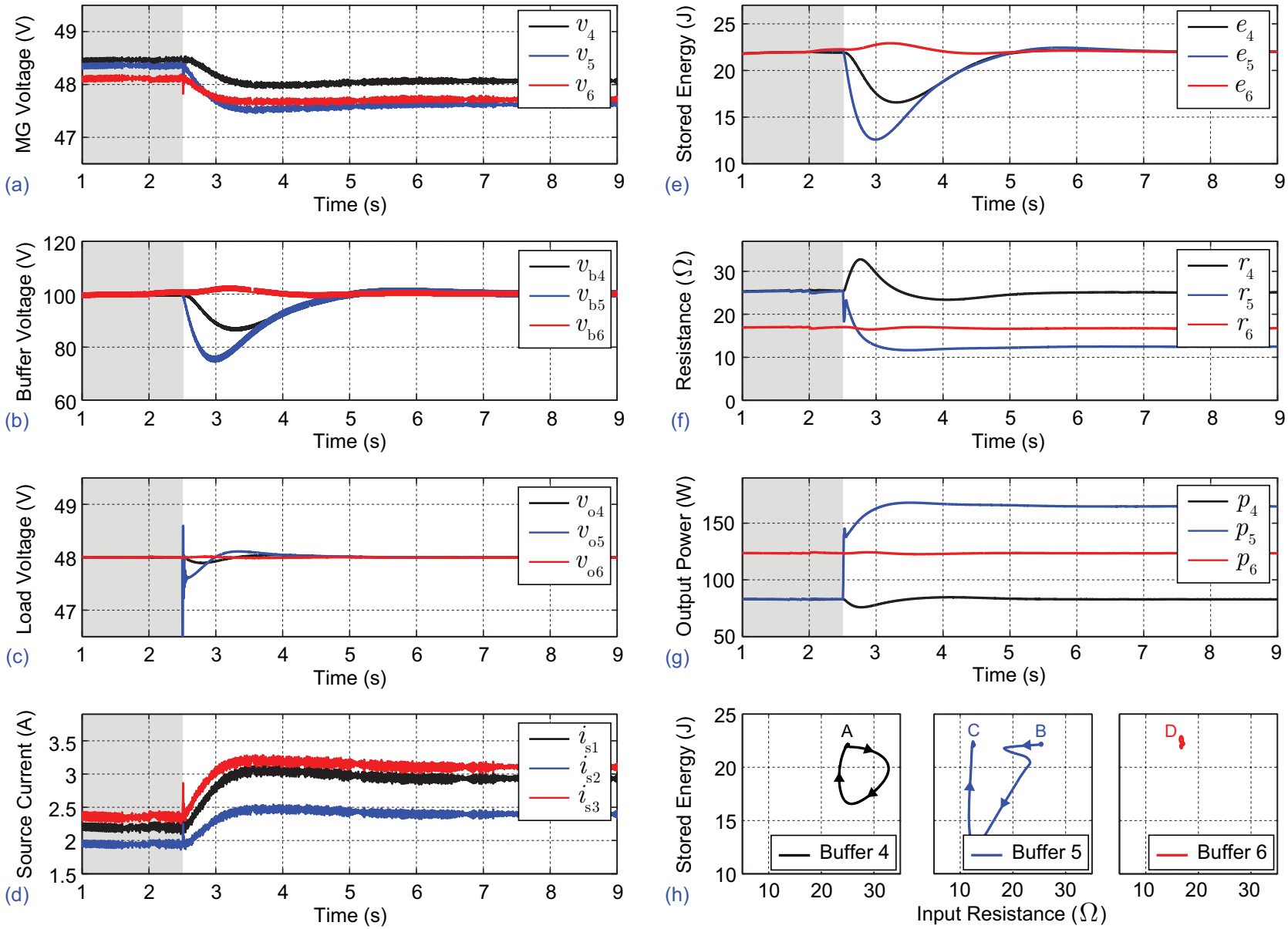
A 48 V DC distribution network with three active loads and three sources



Physical Microgrid Layer

Cyber Communication Layer

Load change at terminal 5

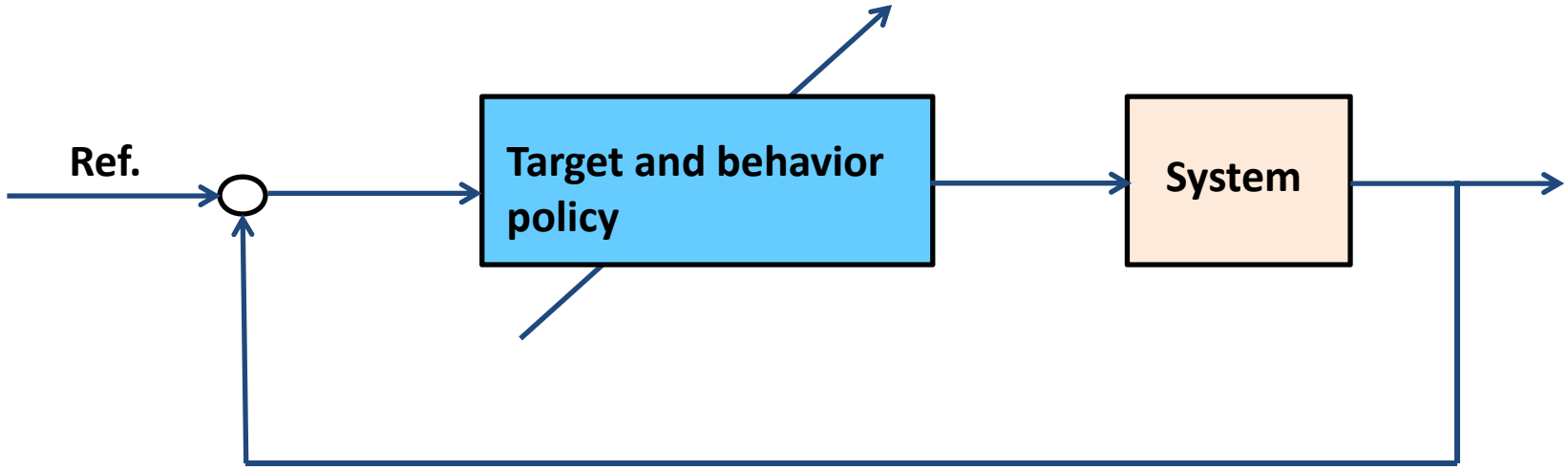


Optimal Leader-Follower Trackers
CPS #2 – Resilient Human-robot Interaction Systems

On-policy RL

Target policy: The policy that we are learning about.

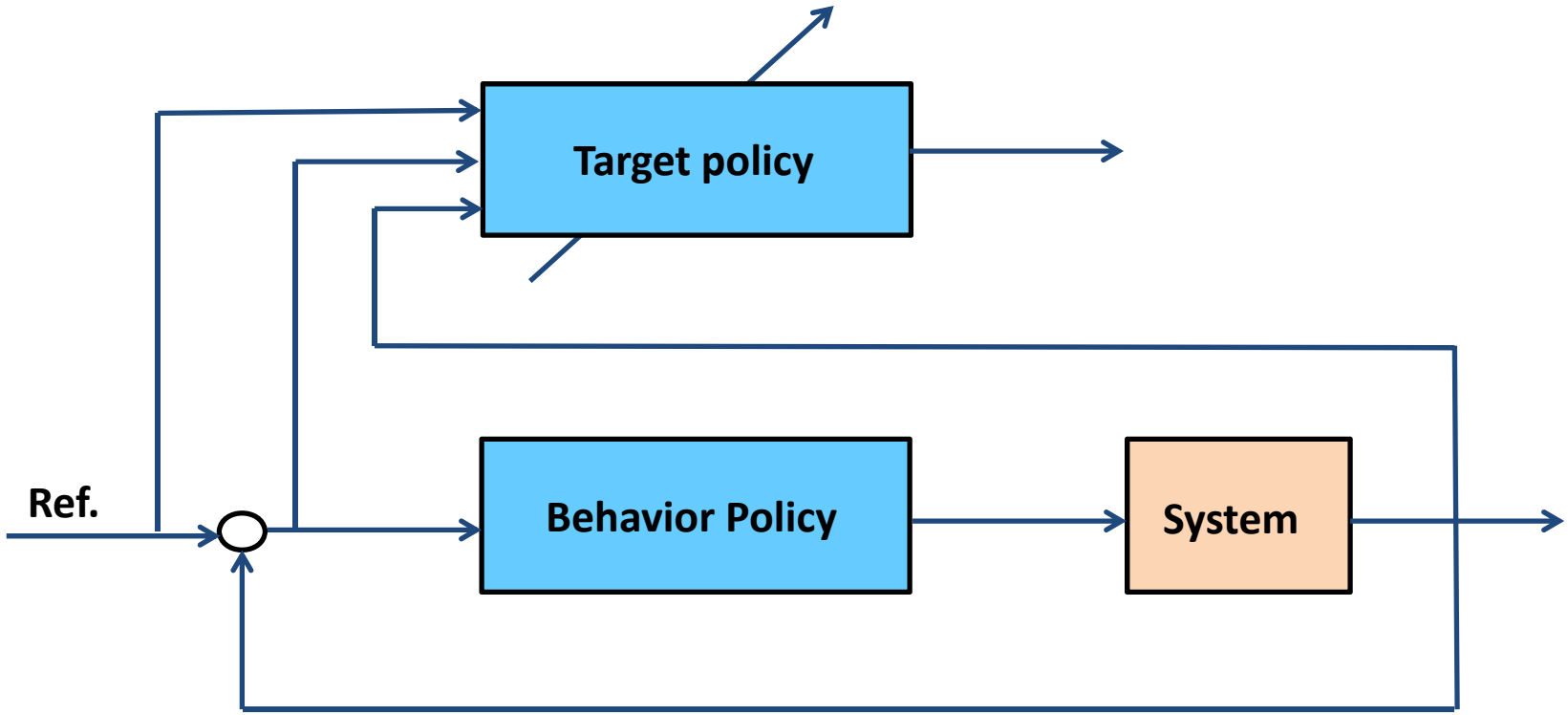
Behavior policy: The policy that generates actions and behavior



Target policy and behavior policy are the same

Off-policy RL

Humans can learn optimal policies while actually applying suboptimal policies



Target policy and behavior policy are different

Off-policy IRL

Yu Jiang & Zhong-Ping Jiang, Automatica 2012

Humans can learn optimal policies while actually applying suboptimal policies

system $\dot{x} = f(x) + g(x)u$

value $J(x) = \int_t^\infty r\left(x(\tau), u(\tau)\right) d\tau$

On-policy IRL

$$J^{[i]}(x(t)) - J^{[i]}(x(t-T)) = -\int_{t-T}^{t} Q(x) d\tau - \int_{t-T}^{t} u^{[i]T} R u^{[i]} d\tau$$

$$u^{[i+1]} = -\frac{1}{2} R^{-1} g^T J_x^{[i]}$$

Must know *g(x)*

Off-policy IRL

$$\dot{x} = f + g u^{[i]} + g(u - u^{[i]})$$

$$J^{[i]}(x(t)) - J^{[i]}(x(t-T)) = -\int_{t-T}^{t} Q(x) d\tau - \int_{t-T}^{t} u^{[i]T} R u^{[i]} d\tau + \underline{2\int_{t-T}^{t} u^{[i+1]T} R(u^{[i]} - u)\, d\tau}$$

DDO

This is a linear equation for $J^{[i]}$ and $u^{[i+1]}$

They can be found simultaneously online using measured data using Kronecker product and VFA

1. Completely unknown system dynamics
2. Can use applied u(t) for –

   disturbance rejection – Z.P. Jiang - R. Song and Lewis, 2015

   robust control – Y. Jiang & Z.P. Jiang, IEEE TCS 2012

   exploring probing noise – without bias ! - J.Y. Lee, J.B. Park, Y.H. Choi 2012

LQ Tracker Problem for Continuous-time Systems

System Dynamics $\dot{x} = Ax + Bu$

$y = Cx$

Assume the reference trajectory is generated by $\dot{y}_d = Fy_d$

Augmented system state: $X(t) = \left[x(t)^T \; y_d(t)^T\right]^T$

Augmented system: $\dot{X} = \begin{bmatrix} A & 0 \\ 0 & F \end{bmatrix} X + \begin{bmatrix} B \\ 0 \end{bmatrix} u \equiv T\,X + B_1 u$

Value function:

$$V\left(X(t)\right) = \frac{1}{2}\int_t^{\infty} e^{-\gamma(\tau - t)}\left[X^T Q_T\, X + u^T R\, u\right] d\tau = \frac{1}{2} X(t)^T P\; X(t)$$

$Q_T = C_1^T Q C_1, \quad C_1 = [C \;\; -I]$

LQT Bellman equation:

$$0 = (T\,X + B_1 u)^T P\, X + X^T P (T\,X + B_1 u) - \gamma X^T P\, X + X^T Q_T\, X + u^T R\, u$$

Online solution to the CT LQT ARE: Off-policy IRL

Humans can learn optimal policies while actually applying suboptimal policies

$$\dot{X} = TX + B_1 u = T_i X + B_1 (K^i X + u), \quad T_i = T - B_1 K^i$$

$$e^{-\gamma T} X(t+T)^T P^i X(t+T) - X(t)^T P^i X(t) = -\int_t^{t+T} \frac{d}{d\tau}(e^{-\gamma(\tau-t)} X^T P^i X)\, d\tau$$

$$= -\int_t^{t+T} e^{-\gamma(\tau-t)} X^T (Q_T + K^{iT} R K^i) X\, d\tau + 2\int_t^{t+T} e^{-\gamma(\tau-t)} (u + K^i X)^T \underbrace{B_1^T P^i}_{R K^{i+1}} X\, d\tau$$

Off-policy IRL Bellman equation

Algorithm. Online Off-policy IRL algorithm for LQT

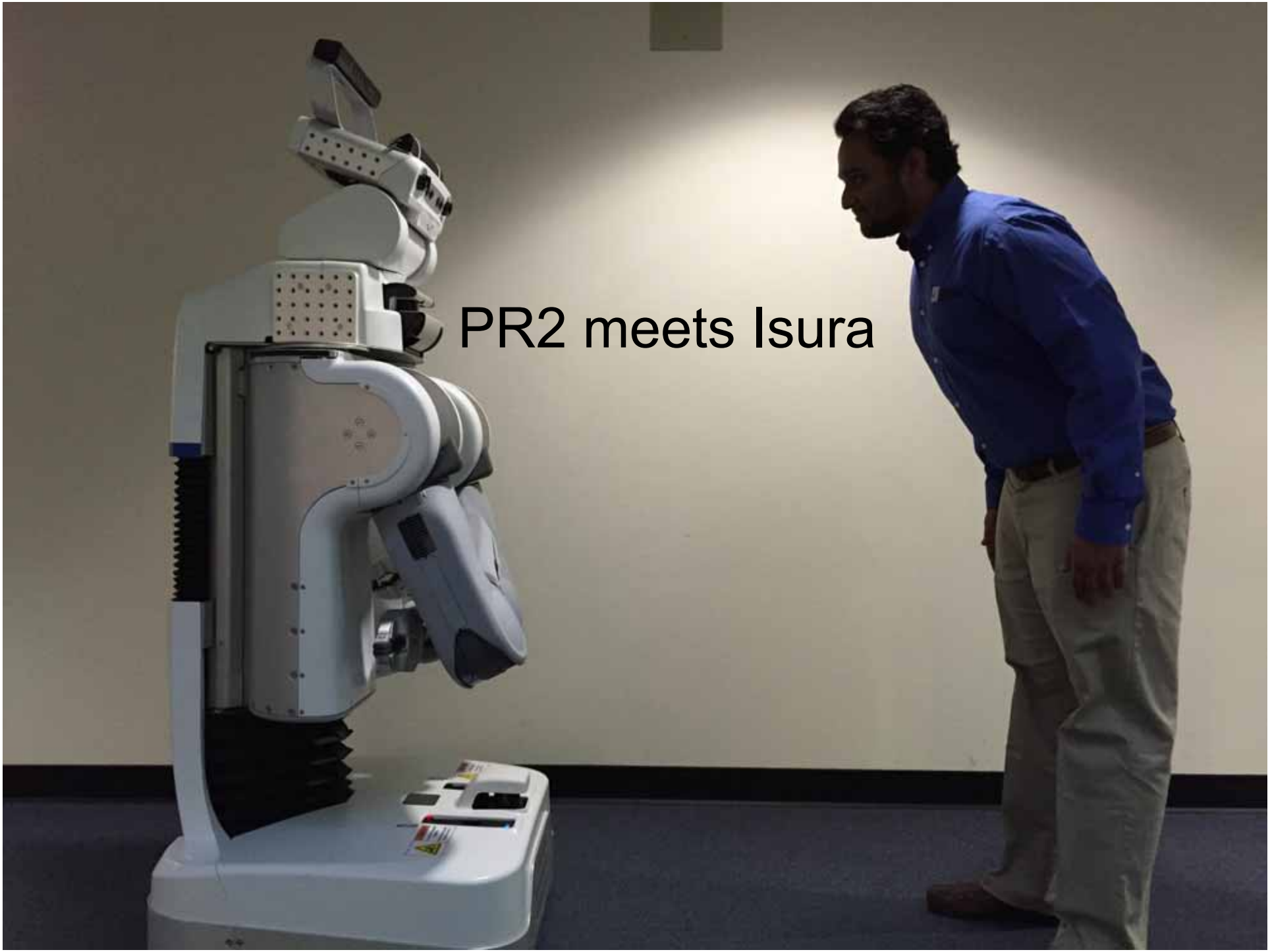
Online step: Apply a fixed control input and collect some data

Offline step: Policy evaluation and improvement using LS on collected data

$$e^{-\gamma T} X(t+T)^T P^i X(t+T) - X(t)^T P^i X(t) = -\int_t^{t+T} e^{-\gamma(\tau-t)} X^T Q_i X\, d\tau +$$

$$2\int_t^{t+T} e^{-\gamma(\tau-t)} (u + K^i X)^T R K^{i+1} X\, d\tau$$ No knowledge of the dynamics is required

Optimal Leader-Follower Trackers
CPS #2 – Resilient Human-robot Interaction Systems


PR2 meets Isura

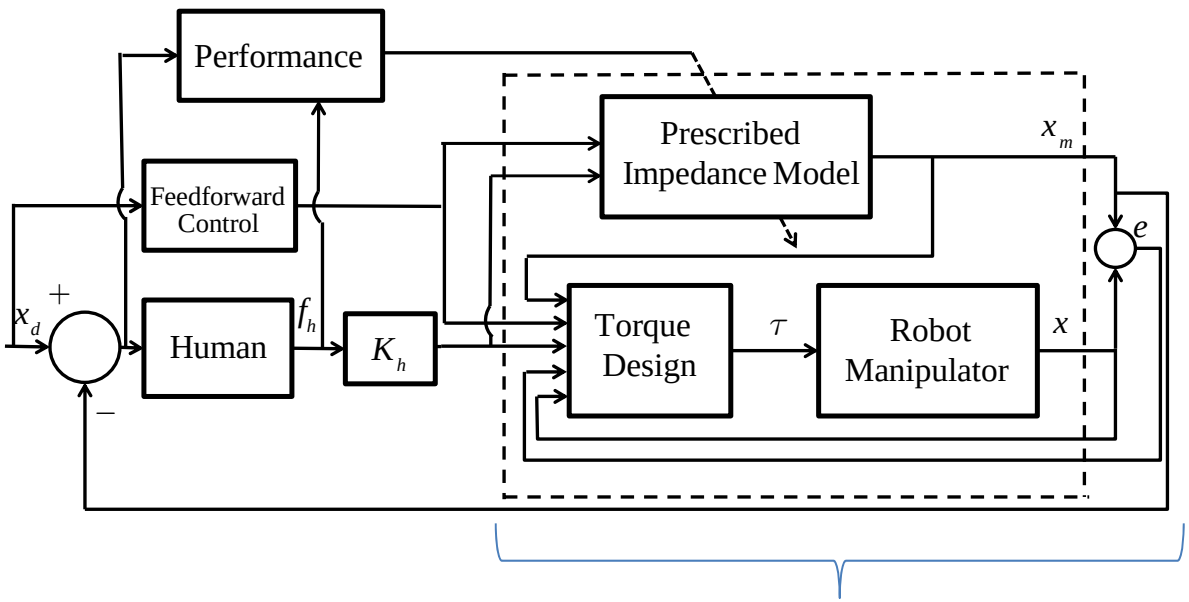
RL for Human-Robot Interactions

H. Modares, I. Ranatunga, F.L. Lewis, and D.O. Popa, "Optimized Assistive Human-robot Interaction using Reinforcement Learning," IEEE Transactions on Cybernetics, to appear 2015.

2-Loop HRI Learning Controller Based on Human Factors Performance Studies
Humans learn 2 components in HRI-

1. **must learn to compensate for nonlinear robot dynamics**
2. **must learn to perform the task**

Our result - The Robot adapts in order to minimize human effort



Robot-specific inner control loop

Task-specific outer-loop control design
Using IRL Optimal Tracking

Implemented on PR2 robot

Human-Robot Interaction Using Reinforcement Learning

IEEE TRANSACTIONS ON CYBERNETICS

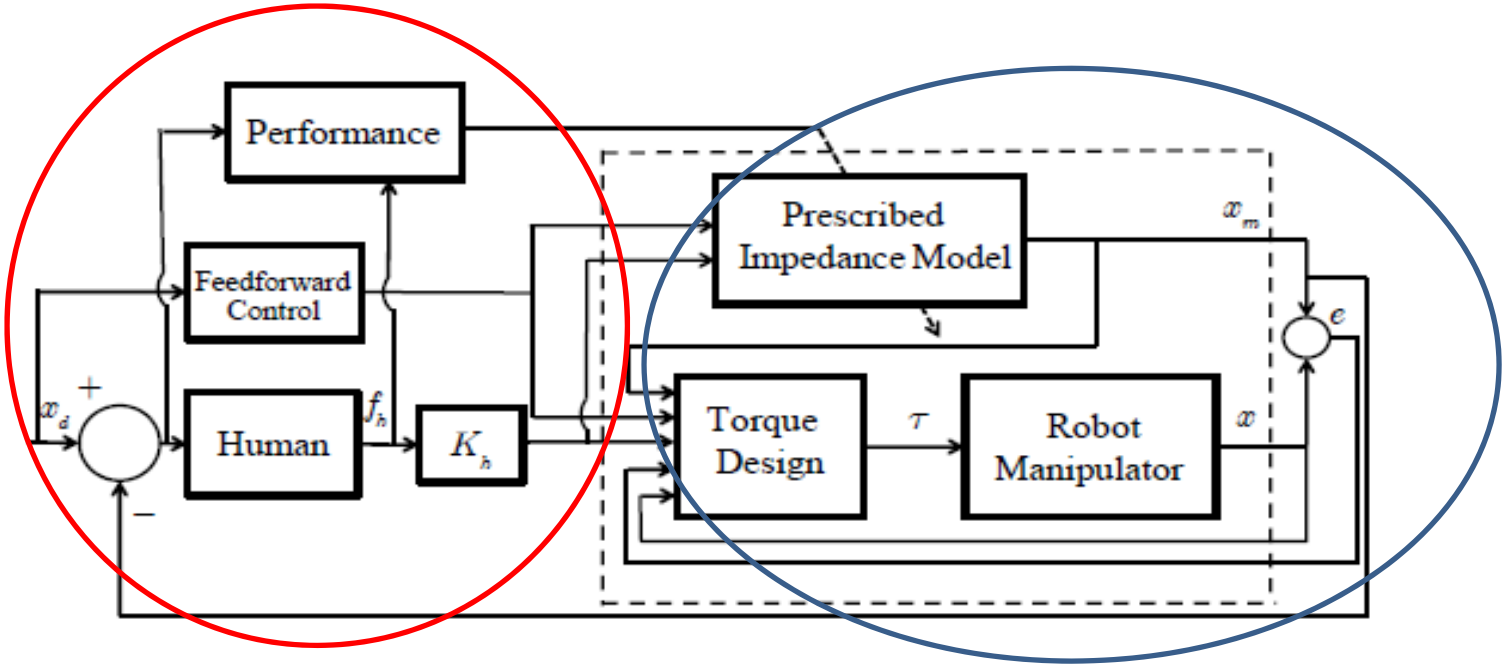
Optimized Assistive Human–Robot Interaction Using Reinforcement Learning

Hamidreza Modares, Isura Ranatunga, *Student Member, IEEE*, Frank L. Lewis, *Fellow, IEEE*, and Dan O. Popa, *Member, IEEE*

2-Loop HRI Learning Controller Based on Human Factors Performance Studies
The Robot adapts in order to minimize human effort

Task control outer loop
using Online Reinforcement Learning

Robot control inner loop
using Neural Adaptive Learning



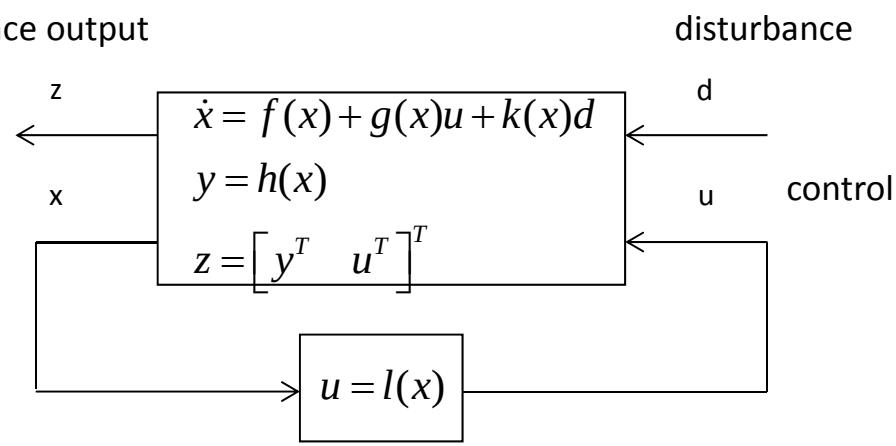
ROS

Modeling and Control of Adversarial Networked Teams
Zero-Sum Games
Bipartite Synchronization on Antagonistic Communication Graphs
Multi-agent Pursuit-Evasion Games

H-Infinity Control Using Neural Networks

Disturbance Rejection

System



Two
Antagonistic Inputs

L_2 Gain Problem

Find control *u(t)* so that

$$\frac{\int_0^\infty \|z(t)\|^2 dt}{\int_0^\infty \|d(t)\|^2 dt} = \frac{\int_0^\infty (h^T h + \|u\|^2) dt}{\int_0^\infty \|d(t)\|^2 dt} \le \gamma^2$$

For all L_2 disturbances
And a prescribed gain γ^2

Zero-Sum differential game

Nature as the opposing player

Define 2-player zero-sum game as

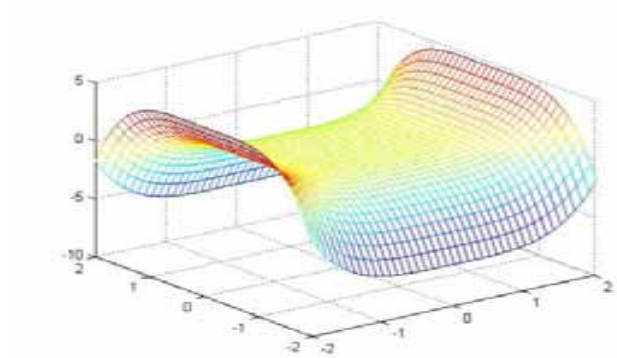
Two
Antagonistic Inputs

$$V^*(x(0)) = \min_u \max_d V(x(0),u,d) \quad = \min_u \max_d \int_0^\infty \left(h^T(x)h(x) + u^T R u - \gamma^2 \|d\|^2 \right) dt$$

The game has a unique value (saddle-point solution) iff the Nash condition holds

$$\min_u \max_d V(x(0),u,d) = \max_d \min_u V(x(0),u,d)$$

Optimal Game Design Yields
- Resilience
- Robustness
- Guaranteed Performance
- Disturbance Rejection

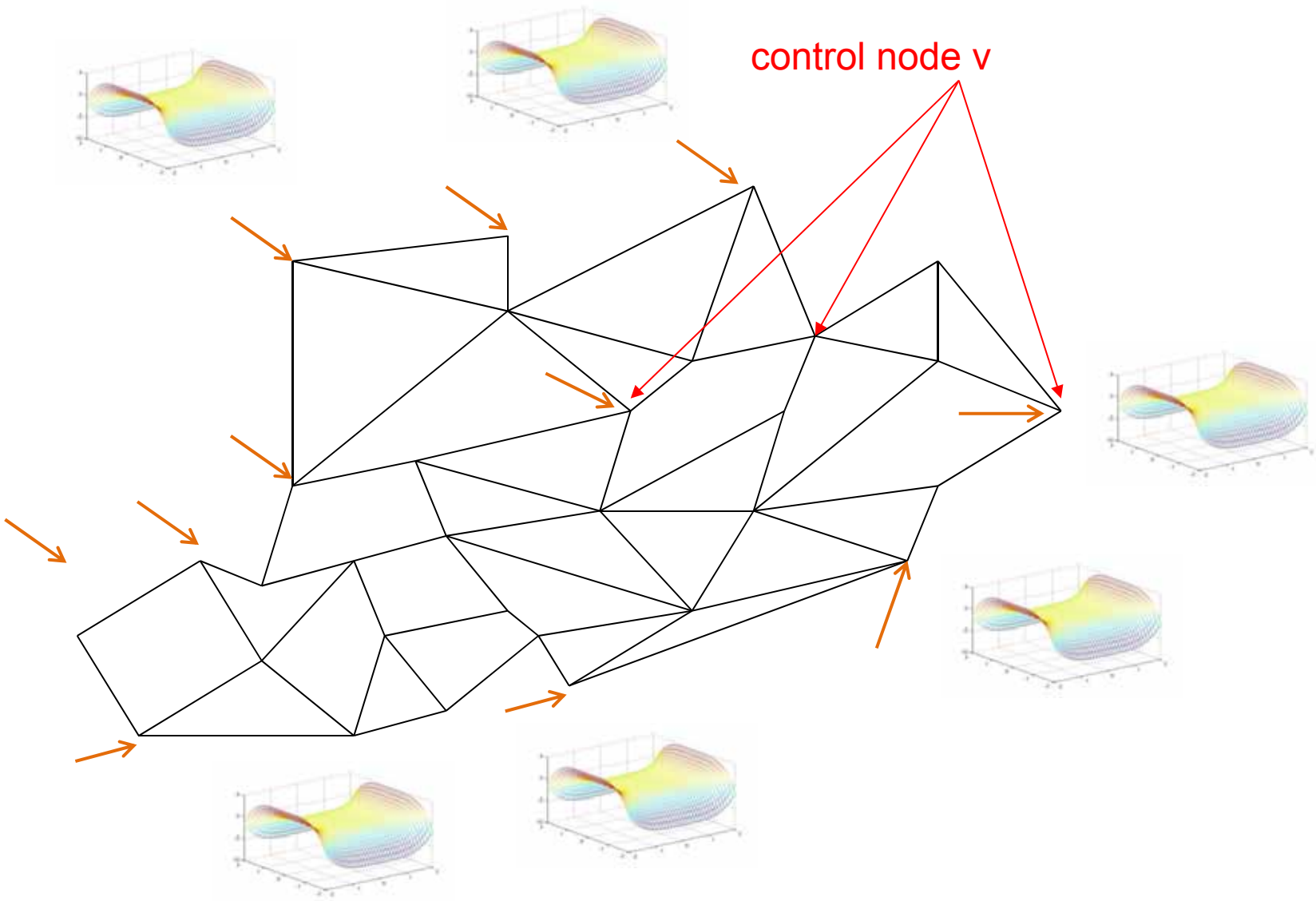


Zero-sum Graphical Games

Cooperative and Adversarial Multi-agent Network

Adversarial disturbance at each agent

Agents want to cooperate to synchronize to the control leader



Cooperative H_∞ Synchronization

System

$$\dot{x}_i = Ax_i + Bu_i + Dd_i$$
$$y_i = Cx_i$$

leader

$$\dot{x}_0 = Ax_0$$
$$y_0 = Cx_0$$

$$\dot{\delta} = (I \otimes A)\delta + (I \otimes B)u + (I \otimes D)\mathrm{d}$$
$$\mathrm{y} = (\mathrm{I} \otimes C)\delta$$

$$\delta = x - \underline{I}x_0$$

Performance output $\quad \|z(t)\|^2 = \delta^T Q\delta + u^T Ru$

Bounded L_2 synchronization error

$$\frac{\int_0^\infty \|z(t)\|^2\, dt}{\int_0^\infty \|d(t)\|_T^2\, dt} = \frac{\int_0^\infty (\delta^T Q\delta + u^T Ru)dt}{\int_0^\infty d(t)^T Td(t)dt} \le \gamma^2$$

H-inf performance index

$$J(\delta, u, d) = \int_0^\infty (\delta^T Q\delta + u^T Ru - \gamma^2 d^T Td)dt$$

Resilience to Network Disturbances

Cooperative H_∞ Synchronization

Kristian Hengster-Movric

Theorem 2. Let the synchronization error dynamics with disturbances acting upon the agents be given in global form as (29). Let the graph have a spanning tree with at least one non-zero pinning gain connecting to a root node. Suppose there exist matrices P_1, P_2, symmetric and positive definite, satisfying

$$P_1 = cR_1\,(L+G) \tag{36}$$

$$A^T P_2 + P_2 A + Q_2 - P_2 B R_2^{-1} B^T P_2 + \gamma^{-2} P_2 D D^T P_2 + M_2^T R_2^{-1} M_2 = 0 \tag{37}$$

for some $Q_2 = Q_2^T > 0$, $R_1 = R_1^T > 0$, $R_2 = R_2^T > 0$, and the coupling gain, $c > 0$. Define the output feedback gain matrix, $K_2 \equiv K_2^*$ as the one satisfying

$$K_2 C = R_2^{-1}(B^T P_2 + M_2)\,, \tag{38}$$

then the control

$$u = -c(L+G) \otimes K_2 C \delta \tag{39}$$

is the solution of the H_∞ static output distributed feedback problem, guaranteeing boundedness of the L_2 gain (30) by γ, defined by the matrices Q, R, T given as

$$R = R_1 \otimes R_2\,,$$

and either

i. $T = P_1 \otimes I = cR_1(L+G) \otimes I$ with

$$Q = c^2 (L+G)^T R_1 (L+G) \otimes (Q_2 + A^T P_2 + P_2 A + \gamma^2 P_2 D D^T P_2) - cR_1 (L+G) \otimes (A^T P_2 + P_2 A + \gamma^2 P_2 D D^T P_2) > 0 \tag{40}$$

or

ii. $T = R_1 \otimes I$ with

$$Q = c^2 (L+G)^T R_1 (L+G) \otimes (Q_2 + A^T P_2 + P_2 A) - cR_1 (L+G) \otimes (A^T P_2 + P_2 A) > 0 \tag{41}$$

Condition on graph topology

$$R_1\,(L+G) = \mathrm{P}_1$$

Condition for existence of local OPFB

$$R_2 K_2 C = B^T P_2 + M_2$$

J. Gadewadikar, Frank L. Lewis, L. Xie, V. Kucera and M. Abu-Khalaf, "Parameterization of all stabilizing H_∞ static state-feedback gains: Application to output-feedback design," *Automatica*, vol. 43, no. 9, pp. 1597-1604, September 2007

Communication Network Limits Available Information Exchange

Communication Network Tries to Destroy Optimality
Proper Controls Design Restores Optimality

Symmetry between OPFB and Graph Topology Information Restrictions

$$u_i = cKC\varepsilon_i = cKe_{yi} = cK\left(\sum_{j\in N_i} e_{ij}(y_j - y_i) \quad + g_i(y_0 - y_i)\right)$$

Network info flow restrictions

$$y_i = Cx_i$$

Local system measurement restrictions

$$\boxed{e_y = -\left(\left(L+G\right)\otimes C\right)\delta}$$

Coarse global OPFB structure

Fine Local OPFB structure

Condition on graph topology

$$R_1\,\underline{(L+G)} = P_1$$

Same Same

Condition for existence of local OPFB

$$R_2 K_2 \underline{C} = B^T P_2 + M_2$$

2-player Zero-sum Nash Equilibrium

Command generator

$$\dot{x}_0 = Ax_0,$$
$$y_0 = Cx_0,$$

Agent Dynamics

$$\dot{x}_i = Ax_i + Bu_i + Dd_i,$$
$$y_i = Cx_i,$$

H-inf performance index

$$J(\delta, u, d) = \int_0^\infty (\delta^T Q \delta + u^T R u - \gamma^2 d^T T d) dt$$

Nash Equilibrium

Definition 3. Given a performance criterion, $J(x_0, u, d)$, the policies u^*, d^* are in the Nash equilibrium if

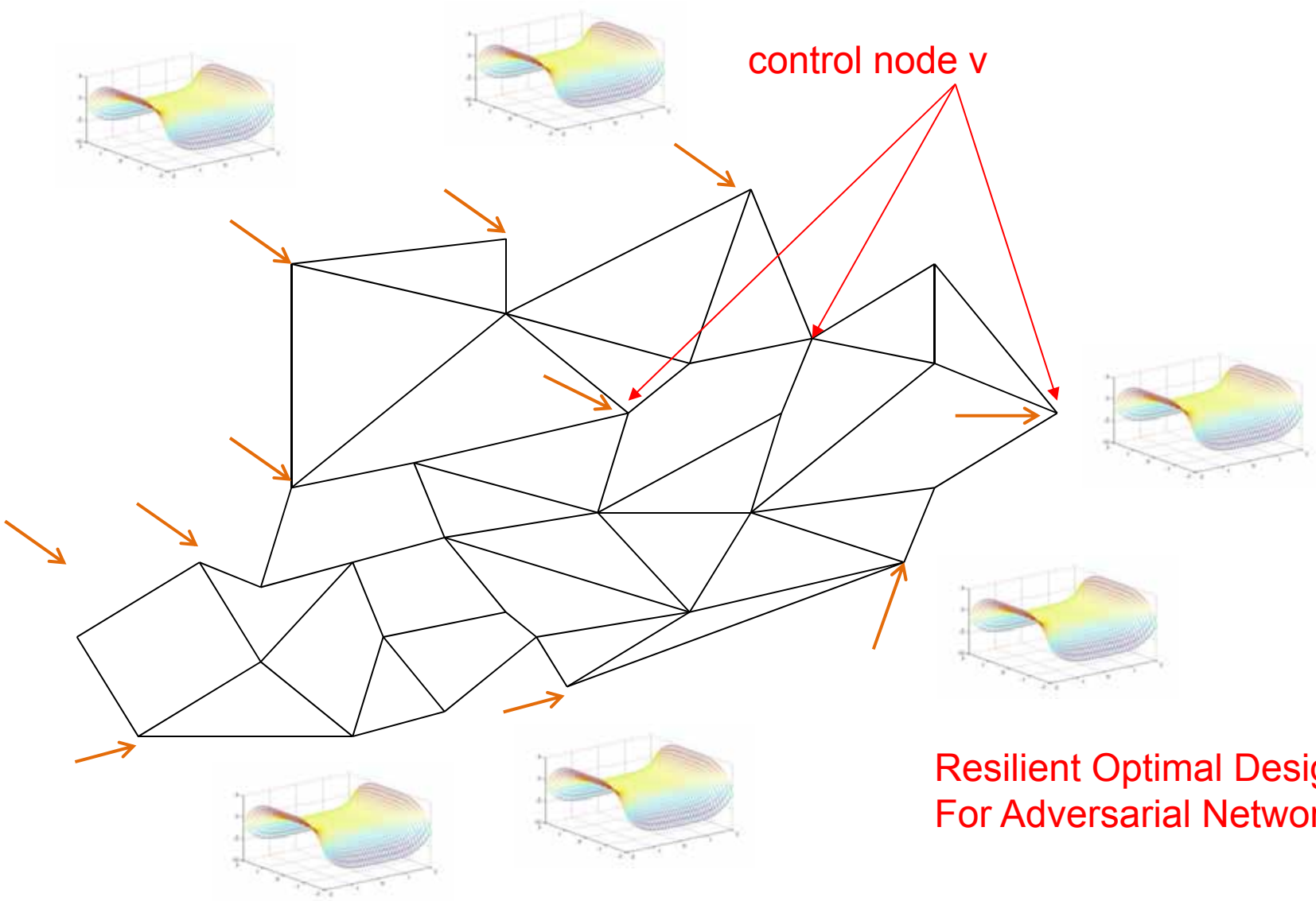
$$J(x(0), u^*, d) \le J(x(0), u^*, d^*) \le J(x(0), u, d^*).$$

Resilience to Network Disturbances

Cooperative and Adversarial Multi-agent Network

Adversarial disturbance at each agent

Agents want to cooperate to synchronize to the control leader



Resilient Optimal Design
For Adversarial Networks

Modeling and Control of Adversarial Networked Teams
Zero-Sum Games
Bipartite Synchronization on Antagonistic Communication Graphs
Multi-agent Pursuit-Evasion Games

Structural Balance for Networked Multi-agent Systems

Continuous-time model of structural balance

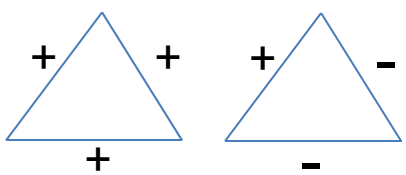
Seth A. Marvel[a], Jon Kleinberg[b,1], Robert D. Kleinberg[b], and Steven H. Strogatz[a]

In almost all human endeavors There are TWO Opposing Teams
- WHY?

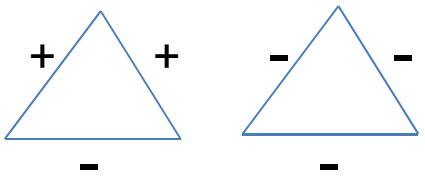
Sports
War
International conglomerates in macroeconomics

Social Psychology Dynamics of Friendship Groups
Can be understood in terms of TRIANGLES of relationships

Stable Groups



Unstable Groups



Stable groups have an odd number of friendship links
Structurally Balanced = Stability

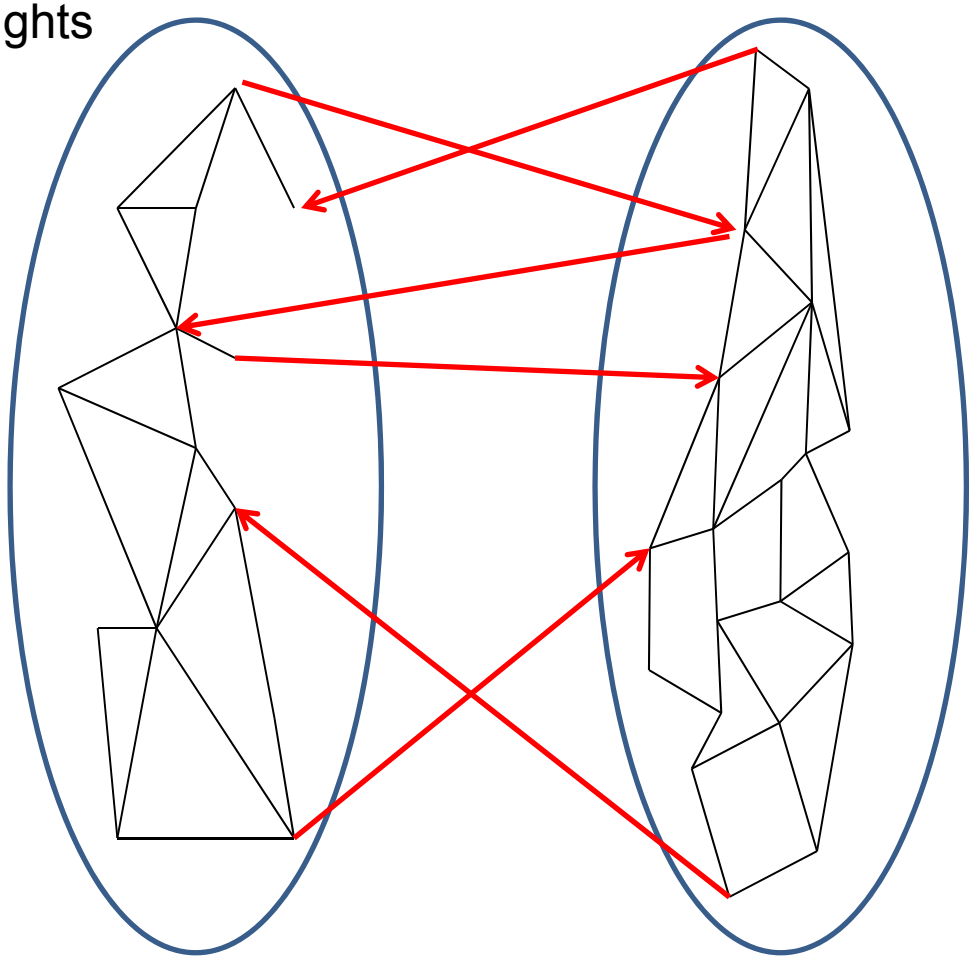
Bipartite Synchronization on Antagonistic Communication Graphs

Structural Balance

Node Bipartition

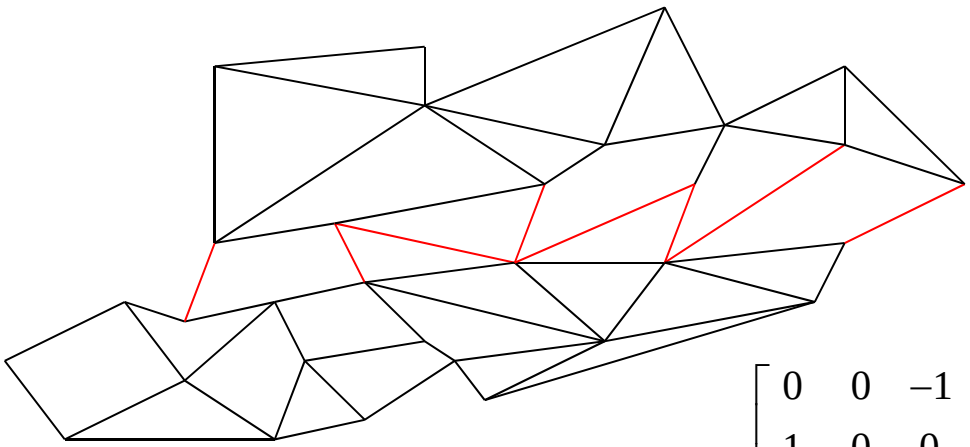
Red edge weights are negative

Black edge weights are positive



Two opposing antagonistic groups

All triangles have an odd number of positive edges



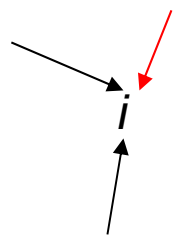
Adjacency matrix

$$A = [a_{ij}]$$

$$A = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \end{bmatrix}$$

$$d_i^s = \sum_{j=1}^{N} \left| a_{ij} \right|$$ Absolute Row sum

N_i In-neighbors of node i



Absolute In-Degree Matrix

$$D^s = \begin{bmatrix} d_1^s & & \\ & \ddots & \\ & & d_N^s \end{bmatrix}$$

Signed graph Laplacian matrix

$$L^s = D^s - A$$

Bipartite Consensus

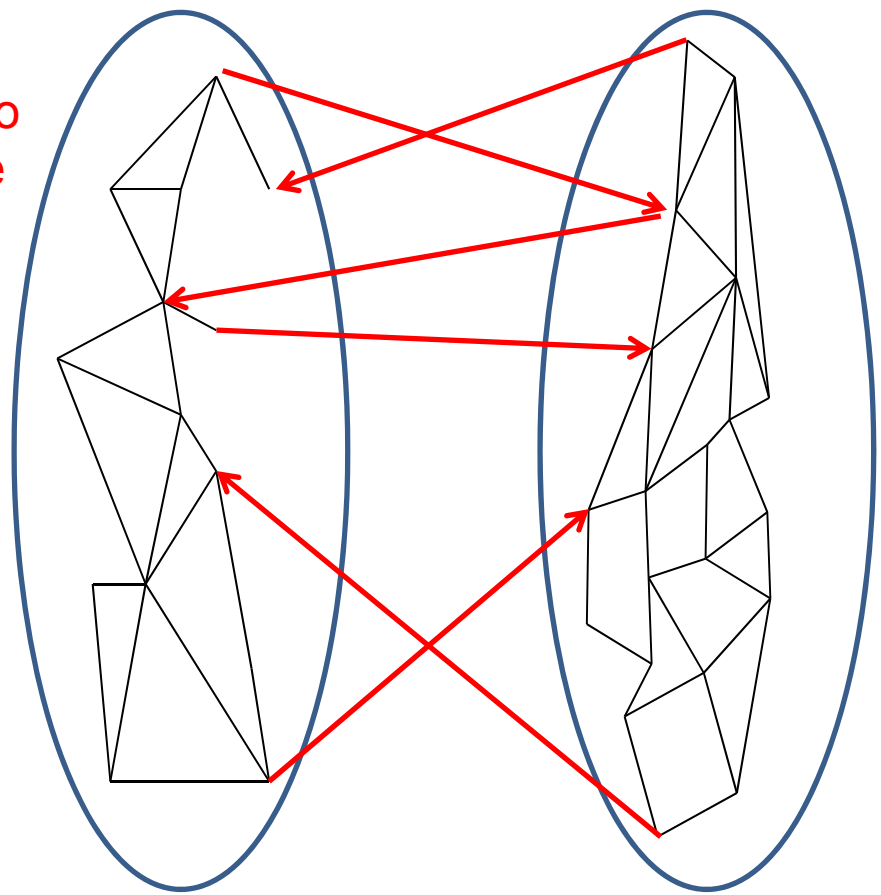
Agent Dynamics $\dot{x}_i = Ax_i + Bu_i, \quad \forall i \in N$

Leader Dynamics $\dot{x}_0 = Ax_0$

New Bipartite Distributed Control Law $u_i = cK\left(\sum_{j \in N_i} (a_{ij}x_j - |a_{ij}|x_i) + (g_i x_0 - |g_i|x_i)\right)$

One group goes to
The leader's state

x_0



Opposing group goes to
Negative of the
Leader's state

$-x_0$

Emergent Behaviors
Emergence of Two Antagonistic Teams

Time-Varying Edge Weights

Adjacency matrix

$$A = [a_{ij}]$$

Undirected graph

$$A = A^T$$

Adaptive edge weights

$$\frac{dA}{dt} = A^2 \quad \text{or} \quad \frac{da_{ij}}{dt} = \sum_k a_{ik} a_{kj}$$

Theorem

For almost any initial weights $A(0)$

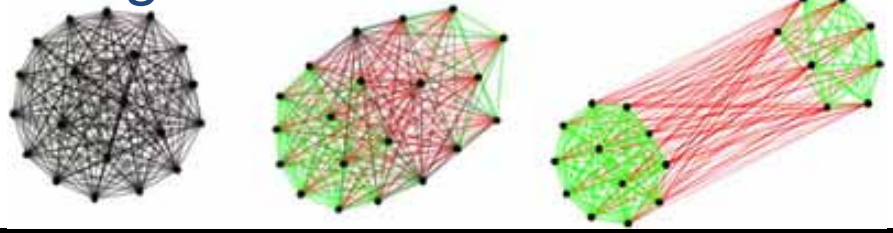
The network becomes structurally balanced
And splits into two antagonistic teams of almost the same size

S. A. Marvel, J. Kleinberg, R. D. Kleinberg, and S. H. Strogatz, “Continuous-time model of structural balance,” Proceedings of the National Academy of Sciences, vol. 108, no. 5, pp. 1771–1776, 2011.

http://vodpod.com/watch/5255321-mathematical-model-shows-how-groups-split-into-factions-physorg-com
January 4, 2011 By Bill Steele

Emergent Behaviors
Emergence of Two Antagonistic Teams

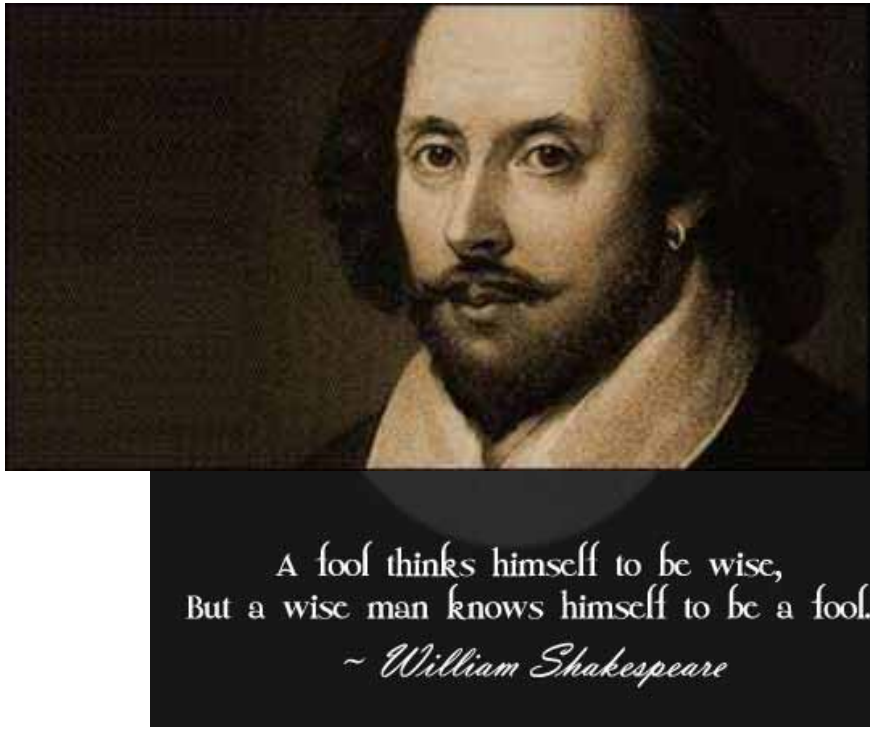
$$\frac{dA}{dt} = A^2$$



[redacted]

Our revels now are ended. These our actors,
As I foretold you, were all spirits, and
Are melted into air, into thin air.

The cloud-capped towers, the gorgeous palaces,
The solemn temples, the great globe itself,
Yea, all which it inherit, shall dissolve,
And, like this insubstantial pageant faded,
Leave not a rack behind.

We are such stuff as dreams are made on,
and our little life is rounded with a sleep.

Prospero, in The Tempest,
act 4, sc. 1, l. 152-6, Shakespeare