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

Clusters of galaxies
Distributed Adaptive Control for Multi-Agent Systems
Flocking

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

Nature and Biological Systems are Resilient!
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

**Key Point**

Lyapunov Functions and Performance Indices Must depend on graph topology


Hongwei Zhang, F.L. Lewis, and Abhijit Das
Synchronized Motion of Biological Groups

Fish school

Birds flock

Locusts swarm

Fireflies synchronize

Nature and Biological Systems are Resilient
The Power of Synchronization

Coupled Oscillators
Diurnal Rhythm
Diameter = length of longest path between two nodes
Volume = sum of in-degrees \[ \text{Vol} = \sum_{i=1}^{N} d_i \]

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

Tree - every node has in-degree=1

Spanning tree
Root node

Leader or root node
Followers
**Algebraic Graph Theory**

**Communication Graph**

(V,E)

N nodes

Adjacency matrix

\[ A = \begin{bmatrix} a_{ij} \end{bmatrix} \]

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

if \( j \in N_i \)

\[ d_i = \sum_{j=1}^{N} a_{ij} \]

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

\[ D = \begin{bmatrix} d_1 & \cdots & \cdots & d_N \\ \cdots & \cdots & \cdots & \cdots \\ \end{bmatrix} \]

\[ L = D - A = \text{graph Laplacian matrix} \]
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
\[
    u_i = \sum_{j \in N_i} a_{ij} (x_j - x_i)
\]

\[
    u_i = -x_i \sum_{j \in N_i} a_{ij} + \sum_{j \in N_i} a_{ij} x_j = -d_i x_i + \left[ a_{i1} \ldots a_{iN} \right] \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}
\]

\[
    A = [a_{ij}]\quad L = D - A = \text{graph Laplacian matrix}
\]

\[
    u = -Dx + Ax = -(D - A)x = -Lx
\]

\[
    \dot{x} = -Lx \quad \text{Closed-loop dynamics}
\]

If \( x \) is an \( n \)-vector then
\[
    \dot{x} = -(L \otimes I_n)x
\]
Communication Graph

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

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

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 \( v \)

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

Local Neighborhood Tracking Error

\( b_i \neq 0 \) If control \( v \) is in the neighborhood of node \( i \)

Global state

\[
x = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}
\]

\[
\dot{x} = -(L + B)x + B_1 v \quad B = \text{diag}\{b_i\}
\]

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

\[
\dot{x}_i = Ax_i + Bu_i
\]

\[
u_i = -Kx_i
\]

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

Couple 6 agents with communication graph

Local neighborhood tracking error

\[
\epsilon_i = \sum_{j \in N_i} e_{ij}(x_j - x_i) + g_i(x_0 - x_i)
\]

\[
u_i = -K\epsilon_i
\]

Nodes synchronize to consensus heading
Example - Non Resilience in Communication Networks

Agent Dynamics and local Feedback design

\[ \dot{x}_i = Ax_i + Bu_i \]

\[ u_i = -Kx_i \]

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

ADD another comm. Link - more information flow

Local neighborhood tracking error

\[ \epsilon_i = \sum_{j \in N_i} e_{ij}(x_j - x_i) + g_i(x_0 - x_i) \]

\[ u_i = K \epsilon_i \]

Causes Unstable Formation! WHY?
Trust-based Control
Define $\xi_{ij}$ 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

$$\xi_i = \begin{bmatrix} \xi_{i1} \\ \xi_{i2} \\ \vdots \\ \xi_{iN} \end{bmatrix} \in \mathbb{R}^N$$

Trust vector for node $i$ has for node $3$

N vector

Standard local voting protocol

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

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**
\[ \dot{\theta}_i = \sum_{j \in N^c_i} \xi_{ij} a_{ij} (\theta_j - \theta_i) \] 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

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

Node 5 injects negative trust values

Internal attack
Malicious node puts out bad trust values
i.e. false information
c.f. virus propagation

Causes Unstable Formation
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

Node 5 injects negative trust values

If node 3 distrusts node 5, Cut out node 5

Other nodes agree that node 5 has negative trust

Convergence of trust

Work by Sajal Das
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

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

Permeability control of the cell membrane

Cellular Metabolism

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

Actor-Critic Learning
We want OPTIMAL performance
ADP- Approximate Dynamic Programming

Desired performance

Sutton & Barto book
Books


New Chapters on:

- Reinforcement Learning
- Differential Games

F. L. Lewis and D. Vrabie,
"Reinforcement learning and adaptive dynamic programming for feedback control,"

IEEE Control Systems Magazine,
F. Lewis, D. Vrabie, and K. Vamvoudakis,
Optimal Control- The Linear Quadratic Regulator (LQR)

User prescribed optimization criterion

\[ V(x(t)) = \int_{t}^{\infty} (x^T Q x + u^T R u) \, d\tau \]

\[ (Q, R) \]

\[ 0 = PA + A^T P + Q - PBR^{-1} B^T P \]

\[ K = R^{-1} B^T P \]

Control \( K \)

\[ \dot{x} = Ax + Bu \]

System

Optimal Control is resilient, robust, has guaranteed stability margins

Formal Design Procedure
Off-line Design Loop Using ARE

On-line real-time Control Loop

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
\[ V(x(t)) = \int_{t}^{\infty} r(x,u) \, dt = \int_{t}^{\infty} (Q(x) + u^T R u) \, dt \]

Bellman Equation, in terms of the Hamiltonian function
\[ H(x, \frac{\partial V}{\partial x}, u) = \dot{V} + r(x,u) = \left( \frac{\partial V}{\partial x} \right)^T \dot{x} + r(x,u) = \left( \frac{\partial V}{\partial x} \right)^T (f(x) + g(x)u) + r(x,u) = 0 \]

Stationarity condition
\[ \frac{\partial H}{\partial u} = 0 \]

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

Hamilton-Jacobi-Bellmen Equation
\[ 0 = \left( \frac{dV^*}{dx} \right)^T f + Q(x) - \frac{1}{4} \left( \frac{dV^*}{dx} \right)^T g R^{-1} g^T \frac{dV^*}{dx} \]

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

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
\[ V(x(t)) = \int_{t}^{t+T} r(x,u) \, d\tau + V(x(t+T)), \quad V(0) = 0 \]

Stationarity condition
\[ \frac{\partial H}{\partial u} = 0 \]

Stationary Control Policy
\[ u = h(x) = -\frac{1}{2} R^{-1} g^T(x) \frac{\partial V}{\partial x} \]
Integral Reinforcement Learning (IRL)- Draguna Vrabie

IRL Policy iteration

Policy evaluation- IRL Bellman Equation

Cost update

\[ V_k(x(t)) = \int_{t}^{t+T} r(x, u_k) \, dt + V_k(x(t + T)) \]

\[ \text{CT Bellman eq.} \]

\[ f(x) \text{ and } g(x) \text{ do not appear} \]

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

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

\[ g(x) \text{ needed for control update} \]

Initial stabilizing control is needed

Converges to solution to HJB eq.

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

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  
\( V = W^T \phi(x) \)
\[
W_k^T \phi(x(t)) = \int_{t}^{t+T} \left( Q(x) + u_k^T R u_k \right) dt + W_k^T \phi(x(t+T))
\]
\[
W_k^T \left[ \phi(x(t)) - \phi(x(t+T)) \right] = \int_{t}^{t+T} \left( Q(x) + u_k^T R u_k \right) dt
\]

Scalar equation with vector unknowns
Regression vector

Reinforcement on time interval \([t, t+T]\)

Same form as standard System ID problems

Now use RLS along the trajectory to get new weights  \( W_k \)

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

Direct Optimal Adaptive Control for Partially Unknown CT Systems
Integral Reinforcement Learning (IRL)

Solve Bellman Equation - Solves Lyapunov eq. without knowing dynamics

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

Data set at time \([t, t+T)\]

\((x(t), \rho(t, t+T), x(t+T))\)

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

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

\[ V_k(x(t)) = \int_{t}^{t+T} r(x,u_k) \, dt + V_k(x(t+T)) \]

Theta waves 4-8 Hz

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

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.


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.
Simulation 1- F-16 aircraft pitch rate controller

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

\[
Q = I, \quad R = I
\]

ARE

\[
0 = PA + A^T P + Q - PBR^{-1}B^T P
\]

Select quadratic NN basis set for VFA

Exact solution

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

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

Stevens and Lewis 2003

\[
x = [\alpha \quad q \quad \delta_e]
\]
Simulations on: F-16 autopilot

A matrix not needed

Converge to SS Riccati equation soln

Solves ARE online without knowing A

\[ 0 = PA + A^T P + Q - PBR^{-1}B^T P \]
Simulation 2: Load Frequency Control of Electric Power system

\[ \dot{x} = Ax + Bu \]

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

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

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 \( A \)

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

IRL period of \( T = 0.1 \text{s} \).

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

Hence, the value estimate was updated every 1.5s.

P matrix parameters \( P(1,1), P(1,3), P(2,4), P(4,4) \)

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

Identify the performance value-Optimal Adaptive

Identify the system model-Indirect Adaptive

Identify the Controller-Direct Adaptive

\[ V(x) = W^T \phi(x) \]
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

Sun Tz bin fa
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

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

Mohammed I. Abouheaf, Frank L. Lewis, Kyriakos G. Vamvoudakis, Sofie Haesaert, Robert Babuska
Manufacturing as the Interactions of Multiple Agents

Each machine has its 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
\[ \dot{\delta}_i = A\delta_i + (d_i + g_i)B_iu_i - \sum_{j \in N_i} e_{ij}B_ju_j \]

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

Key Point

Lyapunov Functions and Performance Indices Must depend on graph topology


Hongwei Zhang, F.L. Lewis, and Abhijit Das
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)

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

Local nbhd. tracking error dynamics

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

Impose Optimality to get Resilience and Robustness

Define Local nbhd. performance index

\[ J_i(\delta_i(0), u_i, u_{-i}) = \frac{1}{2} \int_0^\infty (\delta_i^T Q_i \delta_i + u_i^T R_i u_i + \sum_{j \in N_i} u_j^T R_j u_j) \, dt \equiv \frac{1}{2} \int_0^\infty L_i(\delta_i(t), u_i(t), u_{-i}(t)) \, dt \]

Values driven by neighbors’ controls

New Differential Graphical Game

Local Dynamics
Local Value Function
Only depends on graph neighbors

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 $i$

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

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

Value function of player $i$

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

Central Dynamics

$$\dot{z} = Az + \sum_{i=1}^N B_i u_i$$

Control action of player $i$
Team Interest vs. Self Interest

Cooperation vs. Collaboration

The objective functions of each player can be written as a \textit{team average} term plus a \textit{conflict of interest} term:

\begin{align*}
J_1 &= \frac{1}{3} (J_1 + J_2 + J_3) + \frac{1}{3} (J_1 - J_2) + \frac{1}{3} (J_1 - J_3) \equiv J_{\text{team}} + J_{1}^{\text{coi}} \\
J_2 &= \frac{1}{3} (J_1 + J_2 + J_3) + \frac{1}{3} (J_2 - J_1) + \frac{1}{3} (J_2 - J_3) \equiv J_{\text{team}} + J_{2}^{\text{coi}} \\
J_3 &= \frac{1}{3} (J_1 + J_2 + J_3) + \frac{1}{3} (J_3 - J_1) + \frac{1}{3} (J_3 - J_2) \equiv J_{\text{team}} + J_{3}^{\text{coi}}
\end{align*}

For \textit{N-players}

\[ J_i = \frac{1}{N} \sum_{j=1}^{N} J_{j} + \frac{1}{N} \sum_{j=1}^{N} (J_i - J_j) \equiv J_{\text{team}} + J_i^{\text{coi}}, \quad i = 1, N \]

For \textit{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

\( \{u_1^*, u_2^*, ..., u_N^*\} \) are in Interactive Nash equilibrium if

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

2. There exists a policy \( u_j \) such that

\[ J_i(u_j, u_{G-j}^*) \neq J_i(u_j^*, u_{G-j}^*), \ \forall i, j \in N \]

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

Theorem 3. Let \( (A, B_j) \) be reachable for all \( i \).
Let agent \( i \) be in local best response

\[ J_i(u_i^*, u_{-i}) \leq J_i(u_i, u_{-i}), \ \forall i \]

Then \( \{u_1^*, u_2^*, ..., u_N^*\} \) are in global Interactive Nash iff the graph is strongly connected.
Graphical Game Solution Using Integral Reinforcement Learning

Value function

\[ V_i(\delta_i(t)) = \frac{1}{2} \int_t^\infty (\delta_i^T Q_i \delta_i + u_i^T R_i u_i + \sum_{j \in N_i} u_j^T R_{ij} u_j) \, dt \]

Differential equivalent (Leibniz formula) is Bellman’s Equation

\[ H_i(\delta_i, \frac{\partial V_i}{\partial \delta_i} , u_i, u_{-i} ) \equiv \frac{\partial V_i}{\partial \delta_i} ^T \left( A_i \delta_i + (d_i + g_i) B_i u_i - \sum_{j \in N_i} e_{ij} B_j u_j \right) + \frac{1}{2} \delta_i^T Q_i \delta_i + \frac{1}{2} u_i^T R_i u_i + \frac{1}{2} \sum_{j \in N_i} u_j^T R_{ij} u_j = 0 \]

Stationarity Condition

\[ 0 = \frac{\partial H_i}{\partial u_i} \Rightarrow u_i = -(d_i + g_i) R_i^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i} \]

1. Coupled HJ equations

\[ \frac{\partial V_i}{\partial \delta_i} ^T A_i^c + \frac{1}{2} \delta_i^T Q_i \delta_i + \frac{1}{2} (d_i + g_i)^2 \frac{\partial V_i}{\partial \delta_i} ^T B_i R_i^{-1} B_i^T \frac{\partial V_i}{\partial \delta_i} + \frac{1}{2} \sum_{j \in N_i} (d_j + g_j)^2 \frac{\partial V_j}{\partial \delta_j} ^T B_j R_j^{-1} B_j^T \frac{\partial V_j}{\partial \delta_j} = 0, \, i \in N \]

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

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

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

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

To find online methods for GG

Iterate on these two equations
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)

\[
H_i(\delta, \frac{\partial V_i^k}{\partial \delta}, u_i^k, u_{-i}^k) = 0, \forall i = 1, \ldots, N \tag{38}
\]

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

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

which explicitly is

\[
u_i^{k+1} = -(d_i + e_i) R_{ii}^{-1} B_i^T \frac{\partial V_i^k}{\partial \delta}, \forall i = 1, \ldots, N \tag{39}
\]

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 \( \rho_{ij} = \bar{\sigma}(R_{ij}^{-1} R_{ji}) \), where \( \bar{\sigma}(R_{ij}^{-1} R_{ji}) \) is the maximum singular value of \( R_{ij}^{-1} R_{ji} \).

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 \( \rho_{ij}, \mu_i \) 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 from 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
Micro-grid Hierarchical Control Structure

- **Tertiary Control**
  - Optimal operation in both operating modes
  - Power flow control in grid-tied mode

- **Secondary Control**
  - Voltage deviation mitigation
  - Frequency deviation alleviation
  - Do coop. ctrl. here to
  - Synchronize frequency and voltage

- **Primary Control**
  - Voltage stability provision
  - Frequency stability
  - Plug and play capability for DGs
  - 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.
Active Load Power Buffer
\[
\begin{align*}
\dot{e}_i &= \frac{v_i^2}{r_i} - p_i, \\
\dot{r}_i &= u_i
\end{align*}
\]

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

\[
\begin{bmatrix}
    \dot{e}_i \\
    \dot{r}_i \\
    \dot{p}_i
\end{bmatrix} = \begin{bmatrix}
    0 & 2_i^q \gamma_{ii} - (i_i^q)^2 & -1 \\
    0 & 0 & 0 \\
    0 & 0 & 0
\end{bmatrix} \begin{bmatrix}
    e_i \\
    r_i \\
    p_i
\end{bmatrix} + \begin{bmatrix}
    0 & 1 & 0 \\
    0 & 0 & 1
\end{bmatrix} w_i +
\]

Define coupled performance indices

\[
J_i = \int_0^\infty \left( \sum_{j \in N_i} x_j^T Q_{ij} x_j + \rho_i u_i^2 \right) dt, \quad i = M + 1, \ldots, M + N,
\]

Define Communication Graph
Sparse efficient topology
Optimal design provides Resilience and disturbance rejection
Micro-grid secondary control: Distributed CPS structure

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
Humans can learn optimal policies while actually applying suboptimal policies.

Target policy and behavior policy are different.
Off-policy IRL

Humans can learn optimal policies while actually applying suboptimal policies

system \[ \dot{x} = f(x) + g(x)u \]
value \[ J(x) = \int_{t}^{\infty} r(x(\tau), u(\tau)) d\tau \]

On-policy IRL

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

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

Must know \( g(x) \)

Off-policy IRL

\[ \dot{x} = f + g u^{[i]} + g(u - u^{[i]}) \]
\[ J^{[i]}(x(t)) - J^{[i]}(x(t-T)) = -\int_{t-T}^{t} Q(x) d\tau - \int_{t-T}^{t} u^{[i]T} Ru^{[i]} d\tau + 2\int_{t-T}^{t} u^{[i+1]T} R(u^{[i]} - u) d\tau \]

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 –
LQ Tracker Problem for Continuous-time Systems

System Dynamics
\[
\dot{x} = Ax + Bu \\
y = Cx
\]

Assume the reference trajectory is generated by
\[
\dot{y}_d = F y_d
\]

Augmented system state:
\[
X(t) = \begin{bmatrix} x(t)^T & y_d(t)^T \end{bmatrix}^T
\]

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

\[
Q_T = C_1^T QC_1, \quad C_1 = [C -I]
\]

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

\[
\begin{align*}
X &= TX + B_1u = T_1X + B_1(K^iX + u), \\
T_1 &= T - B_1K^i \\
\end{align*}
\]

\[
e^{-\gamma T}X(t+T)^T P^i X(t+T) - X(t)^T P^i X(t) = -\int_t^{t+T} \frac{d}{d\tau} (e^{-\gamma(\tau-t)}X^T P^i X) d\tau \\
= -\int_t^{t+T} e^{-\gamma(\tau-t)}X^T (Q^T + K^{iT} R K^i) X d\tau + 2\int_t^{t+T} e^{-\gamma(\tau-t)}(u + K^iX)^T \underbrace{B_1^T P^i X d\tau}_{R K^{i+1}}
\]

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

\[
e^{-\gamma T}X(t+T)^T P^i X(t+T) - X(t)^T P^i X(t) = -\int_t^{t+T} e^{-\gamma(\tau-t)}X^T Q_i X d\tau + 2\int_t^{t+T} e^{-\gamma(\tau-t)}(u + K^iX)^T R K^{i+1} X d\tau
\]

No knowledge of the dynamics is required
Optimal Leader-Follower Trackers
CPS #2 – Resilient Human-robot Interaction Systems
PR2 meets Isura
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
Human-Robot Interaction Using Reinforcement Learning

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
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

Performance output

\[ \dot{x} = f(x) + g(x)u + k(x)d \]
\[ y = h(x) \]
\[ z = \begin{bmatrix} y^T & u^T \end{bmatrix} \]
\[ u = l(x) \]

Two Antagonistic Inputs

L₂ Gain Problem

Find control \( u(t) \) so that

\[ \int_0^\infty \|z(t)\|^2 \, dt = \int_0^\infty (h^T h + \|u\|^2) \, dt \leq \gamma^2 \]

For all L₂ disturbances
And a prescribed gain \( \gamma^2 \)

Zero-Sum differential game

Nature as the opposing player
Define 2-player zero-sum game as

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

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

color node $v$
Cooperative $H_\infty$ Synchronization

System

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

leader

\[ \dot{x}_0 = Ax_0 \]
\[ y_0 = Cx_0 \]

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

Performance output

\[ \|z(t)\|^2 = \delta^TQ\delta + u^TRu \]

Bounded $L_2$ synchronization error

\[ \int_0^\infty \|z(t)\|^2 dt \leq \int_0^\infty (\delta^TQ\delta + u^TRu)dt \]

\[ \int_0^\infty \|d(t)\|^2_t dt \leq \int_0^\infty d(t)^T Td(t)dt \leq \gamma^2 \]

$H$-inf performance index

\[ J(\delta, u, d) = \int_0^\infty (\delta^TQ\delta + u^TRu - \gamma^2 d^T Td)dt \]

Resilience to Network Disturbances
Cooperative $H_\infty$ Synchronization

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

$$P_1 = cR_1 (L + G)$$

(36)

$$A^TP_2 + P_2A + Q_2 - P_2BR_2C^TB^TP_2 + \gamma^2 P_2DD^TP_2 + M_2^TR_2^{-1}M_2 = 0$$

(37)

for some $Q_2 > 0, R_1 > 0, R_2 > 0$, and the coupling gain, $c > 0$. Define the output feedback gain matrix, $K_2 = K^T_2$ as the one satisfying

$$K_2C = R_2^{-1}(B^TP_2 + M_2),$$

(38)

then the control

$$u = -c(L + G) \otimes K_2C\delta$$

(39),

is the solution of the $H_\infty$ static output distributed feedback problem, guaranteeing boundedness of the $L_2$ gain (30) by $\gamma$, 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

$$Q = c^2(L + G)^TP_2R_2(L + G) \otimes (Q_2 + A^TP_2 + P_2A + \gamma^2 P_2DD^TP_2) - cR_2(L + G) \otimes (A^TP_2 + P_2A) \otimes (A^TP_2 + P_2A) > 0$$

(40)

or

ii. $T = R_1 \otimes I$ with

$$Q = c^2(L + G)^TP_2R_2(L + G) \otimes (Q_2 + A^TP_2 + P_2A) - cR_1(L + G) \otimes (A^TP_2 + P_2A) > 0$$

(41)

J. Gadewadikar, Frank L. Lewis, L. Xie, V. Kucera and M. Abu-Khalaf, “Parameterization of all stabilizing $H_\infty$ 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

\[ u_i = cKC \epsilon_i = cKe_y = cK \left( \sum_{j \in N_i} e_{ij} (y_j - y_i) + g_i (y_0 - y_i) \right) \]  

Network info flow restrictions

\[ y_i = Cx_i \]  

Local system measurement restrictions

\[ e_y = - \left( (L + G) \otimes C \right) \delta \]

Coarse global OPFB structure

Fine Local OPFB structure

Condition on graph topology

\[ R_1 (L + G) = P_1 \]

Same-Same

Condition for existence of local OPFB

\[ R_2 K_2 C = B^T P_2 + M_2 \]
2-player Zero-sum Nash Equilibrium

Command generator
\[ \dot{x}_0 = Ax_0, \]
\[ y_0 = Cx_0, \]

Agent Dynamics
\[ \dot{x}_i = Ax_i + Bu_i + Dd_i, \]
\[ y_i = Cx_i, \]

\[ H_{-inf} \text{ performance index} \]
\[ J(\delta, u, d) = \int_0^\infty (\delta^T Q \delta + u^T Ru - \gamma^2 d^T T d) dt \]

Nash Equilibrium

**Definition 3.** Given a performance criterion, \( J(x_0, u, d) \), the policies \( u^*, d^* \) are in the Nash equilibrium if
\[ J(x(0), u^*, d) \leq J(x(0), u^*, d^*) \leq J(x(0), u, d^*). \]

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\textsuperscript{a}, Jon Kleinberg\textsuperscript{b,}\textsuperscript{1}, Robert D. Kleinberg\textsuperscript{b}, and Steven H. Strogatz\textsuperscript{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

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

Absolute Row sum

\[ d_i^s = \sum_{j=1}^{N} |a_{ij}| \]

In-neighbors of node i

\[ N_i \]

Absolute In-Degree Matrix

\[ D^s = \begin{bmatrix} d_1^s & \cdots & \cdots & \cdots & d_N^s \end{bmatrix} \]

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
\[
u_i = cK \left( \sum_{j \in N_i} (a_{ij}x_j - |a_{ij}|x_i) + (g_ix_0 - |g_i|x_i) \right)
\]
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

Emergent Behaviors
Emergence of Two Antagonistic Teams

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