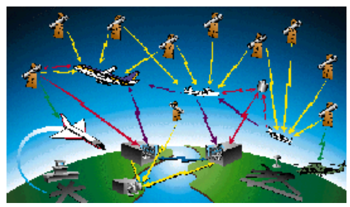
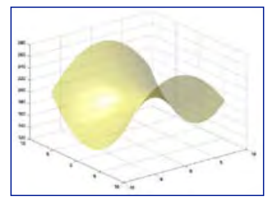
Task Planning and Multi-Agent Systems

Robert Stengel
Robotics and Intelligent Systems,
MAE 345, Princeton University, 2015

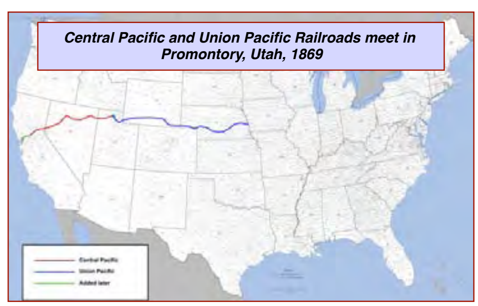
- Decision making
- Task decomposition, communities, and connectivity
- Cooperation, collaboration, competition, and conflict
- Path planning (see Lecture 5)
- Multi-agent architectures





Task Planning Goals

- Accomplish an objective
  - Make a decision
  - Gather information
  - Build something
  - Analyze something
  - Destroy something
- Determine and follow a path
  - Minimize time or cost
  - Take the shortest path
  - Avoid obstacles or hazards
- Work toward a common goal
  - Integrate behavior with higher objectives
  - Do not impede other agents



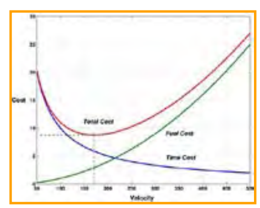
Central Pacific and Union Pacific Railroads meet in Promontory, Utah, 1869

Driving "The Golden Spike"

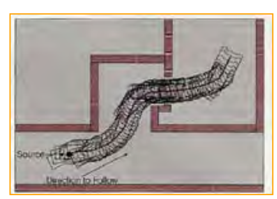
More Task Planning Goals

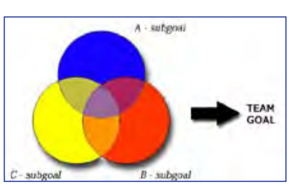
- **Provide leadership for other agents**
  - **Issue commands**
  - **Receive and decode information**
- **Provide assistance to other agents**
  - **Coordinate actions**
  - **Respond to requests**
- **Defeat opposing agents**
  - **Compete and win**
- **Path planning**
  - **See Lecture 5**

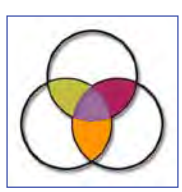
Common Threads in Task Accomplishment



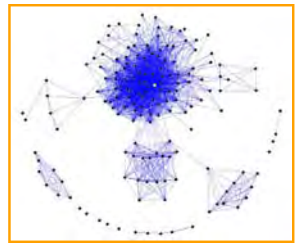
- **Optimize a cost function**
- **Satisfy or avoid constraints**
- **Exhibit desirable behavior**
- **Tradeoff individual and team goals**
- **Use resources effectively and efficiently**
- **Negotiate**
- **Cooperate with team members**
- **Overcome adversity and ambiguity**





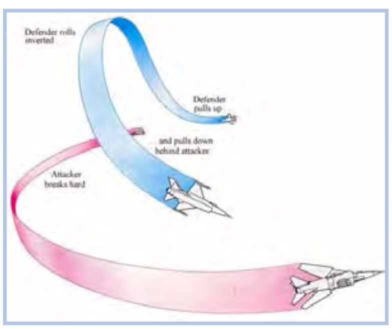


Task Planning



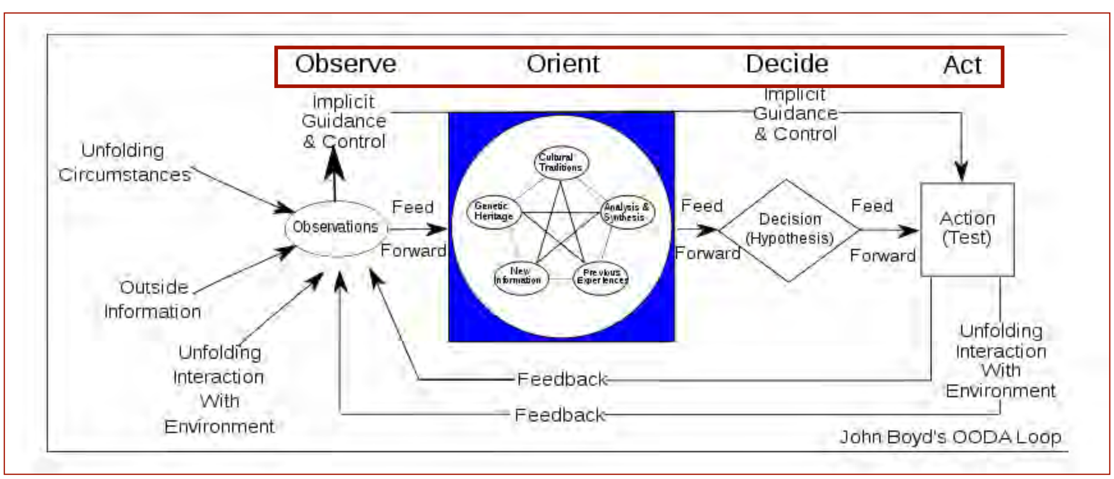
- Situation awareness
- Decomposition and identification of communities
- Development of strategy and tactics

		Phase	
		Process	Outcome
Objective	Tactical (short-term)	Situation Assessment	Situation Awareness
	Strategic (long-term)	Comprehension	Understanding

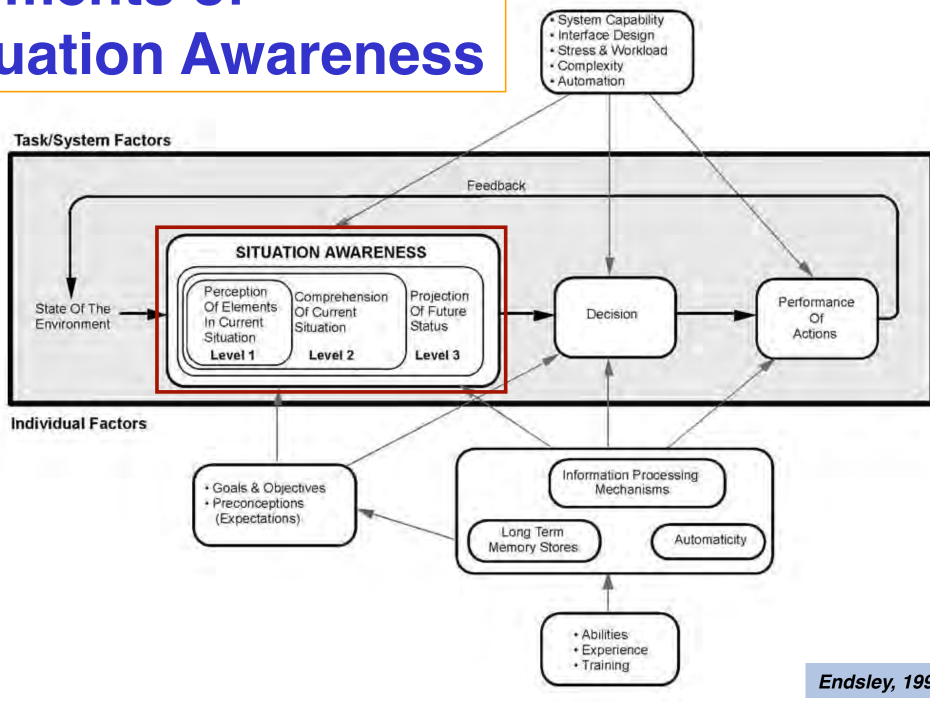


Boyd's "OODA Loop" for Combat Operations

- Derived from air-combat maneuvering strategy
- General application to learning processes other than military



Elements of Situation Awareness



Endsley, 1995

Important Dichotomies in Planning

Strength, Weakness, Opportunity, and Threat (SWOT) Analysis

"Knok-Knoks" and "Unk-Unks"

Known Knowns	Known Unknowns
Unknown Knowns	Unknown Unknowns

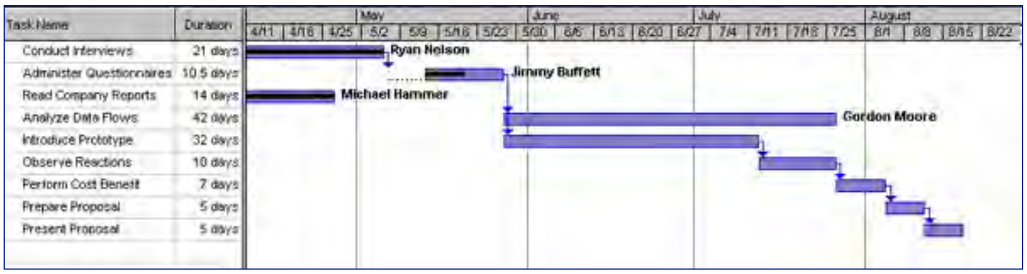
Strategy/Tactics Development and Deployment

- Development of long- and short-term actions/activities for implementation and operation
- Sequence of procedures to be executed
  - fixed or adaptive
- Exposition of approach
  - Rules of engagement
  - Concept of Operations (CONOPS)
- Spectrum of flexibility
  - Rigid sequence <---> Learning systems
- Think "Expert System"

Planning Tools

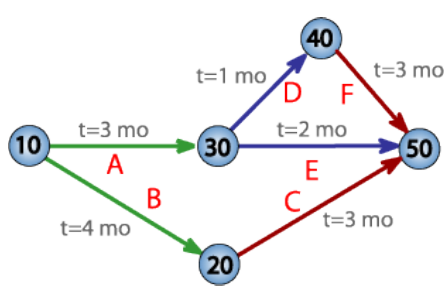
Program Management: Gantt Chart

- Project schedule
- Task breakdown and dependency
- Start, interim, and finish elements
- Time elapsed, time to go

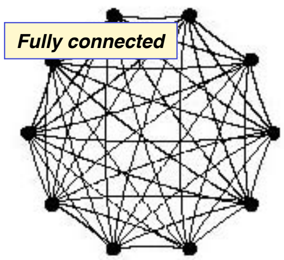


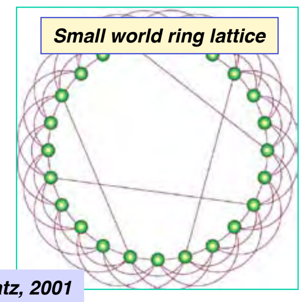
Program Evaluation and Review Technique (PERT) Chart

- Milestones
- Path descriptors
- Activities, precursors, and successors
- Timing and coordination
- Identification of critical path
- Optimization and constraint



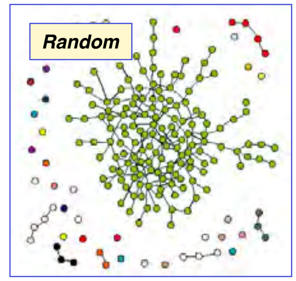
Task Decomposition: Community Identification

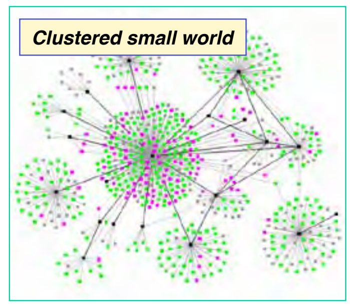




Strogatz, 2001

- Connectivity of individuals
- Individuals assemble in communities or clusters
- Complex networks
  - Random networks
  - Small-world networks
  - Scale-free networks
- Degrees of separation





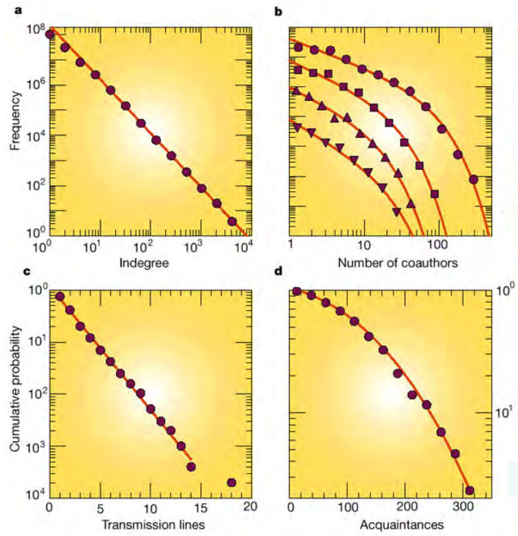
Community <-> Communication

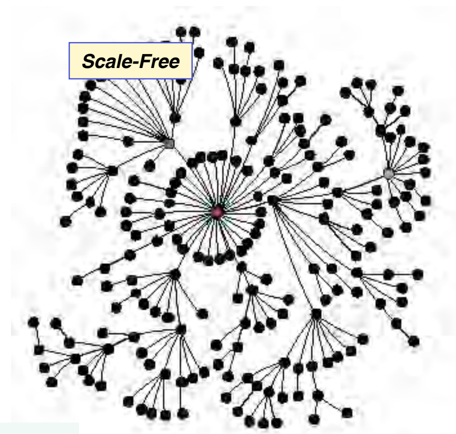
Communities and Networks

Scale-Free Networks

Frequency and cumulative distributions of cluster sizes, *k*, inversely proportional to k^x, $x \sim -2$ or -3

No "knee" that implies a scale in the distribution



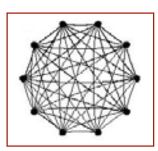


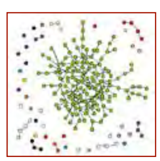
Strogatz, 2001

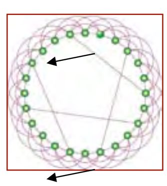
https://en.wikipedia.org/wiki/Scale-free_network

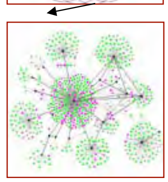
Community Examples

- **Families**
- **Classmates**
- **Neighbors**
- **Social Networks**
  - Facebook
  - LinkedIn
- **Media Networks**
- **Corporations**
- **Employees**
- **Customers**
- **Sports Leagues**
  - Teams
    - Managers
      - Players
      - Trainers
- **Airlines**
- **Cities**









- **Associations**
- **Governments**
  - Agencies
    - Laboratories
      - Managers
      - Scientists
- **Military organizations**
  - Army
    - Corps
      - Division
        - Brigade
  - Regiment
    - Battalion
      - Company
        - Platoon
  - Squad
    - Soldier
  - Special Operations
- **Terrorist organizations**

Multi-Agent Systems

- Specialized vs. general-purpose agents
- Organizational models
- Cooperators
  - Leader/follower (hierarchical)
  - Equal members
- Collaborators
  - Air, ground, and sea traffic
  - Customers
- Competitors
  - Individual game players
  - Sports teams
  - Political/military organizations
- Negotiators
  - Politicians
  - Employer/employee representatives

Multi-Agent Systems

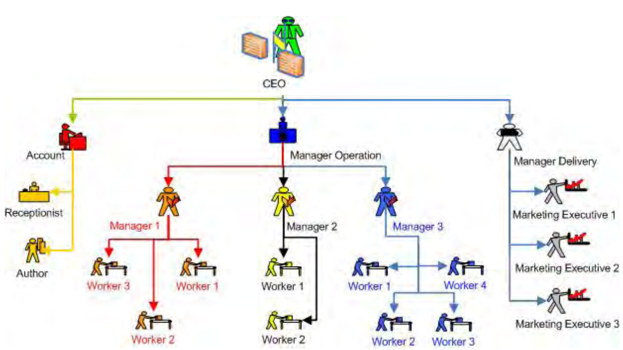
- Cooperation and collaboration should lead to “win-win” (non-zero-sum) solutions
- Competition should lead to “win-lose” (zero-sum) solutions
- Negotiation should lead to “win-win” but may lead to “win-lose” solutions

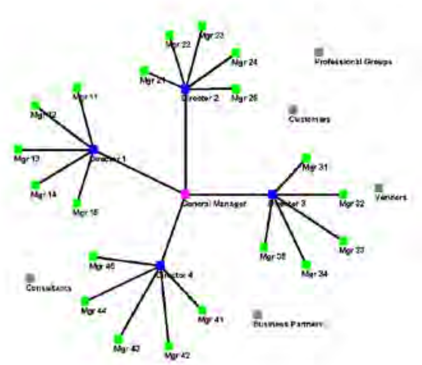
Typical Characteristics of Multi-Agent Architectures

- **Federated (centralized) problem solving**
  - Doctrinaire
  - Coupled
  - Synchronous
  - Fragile
  - Complex
  - Strategic
  - Information-rich
  - Unified
  - Integrated
  - Top-down
  - Globally optimal

- **Distributed problem solving**
  - Autonomous
  - Independent
  - Asynchronous
  - Robust
  - Simple
  - Tactical
  - Parsimonious
  - Idiosyncratic
  - Modular
  - Bottom-up
  - Locally optimal

Hierarchical Tree or Hub-and-Spoke Network?



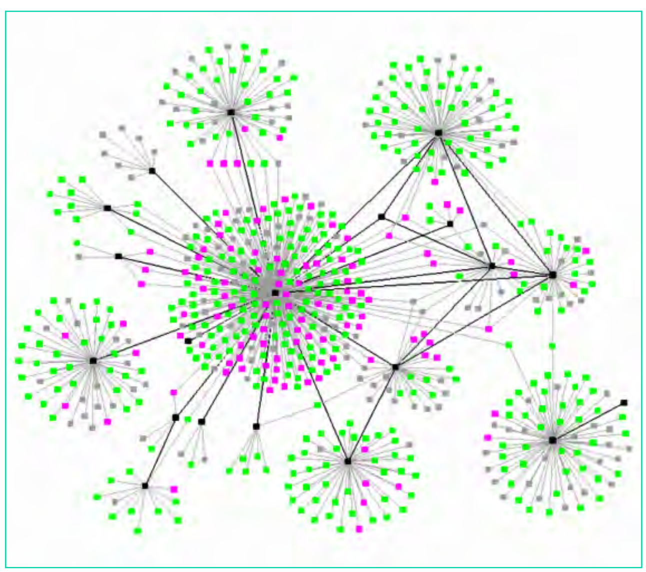


What is the Nature, Quality, and Significance of Connections?

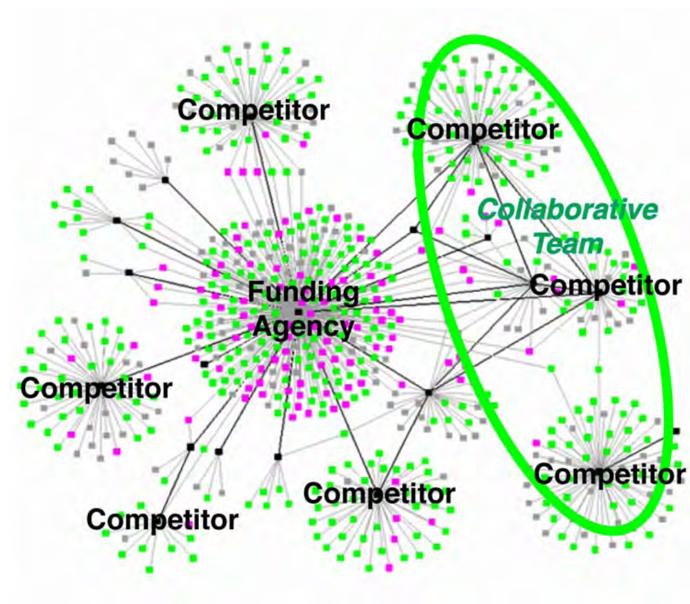
- Communication
- Collaboration
- Coordination
- Negotiation
- Competition
- Conflict

Connections May Connote Different Relationships

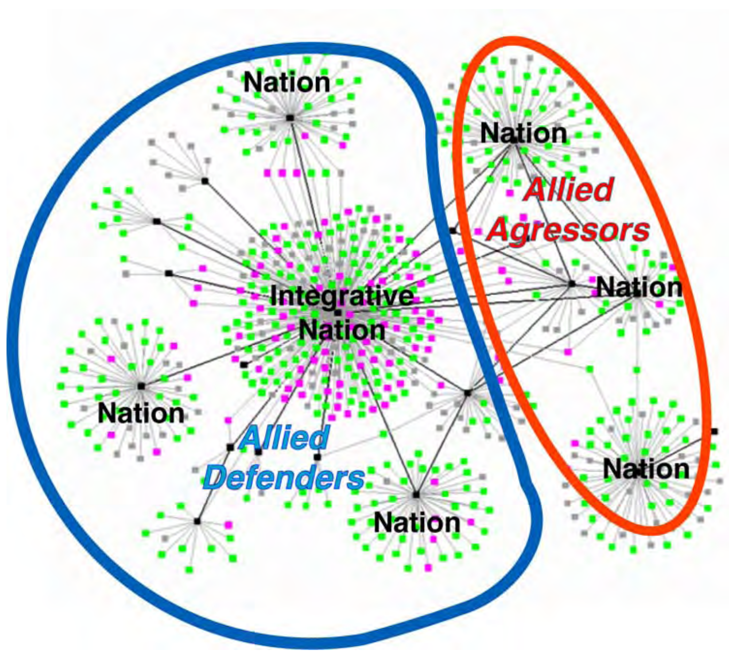
- Communication
- Collaboration
- Coordination
- Negotiation
- Competition
- Conflict



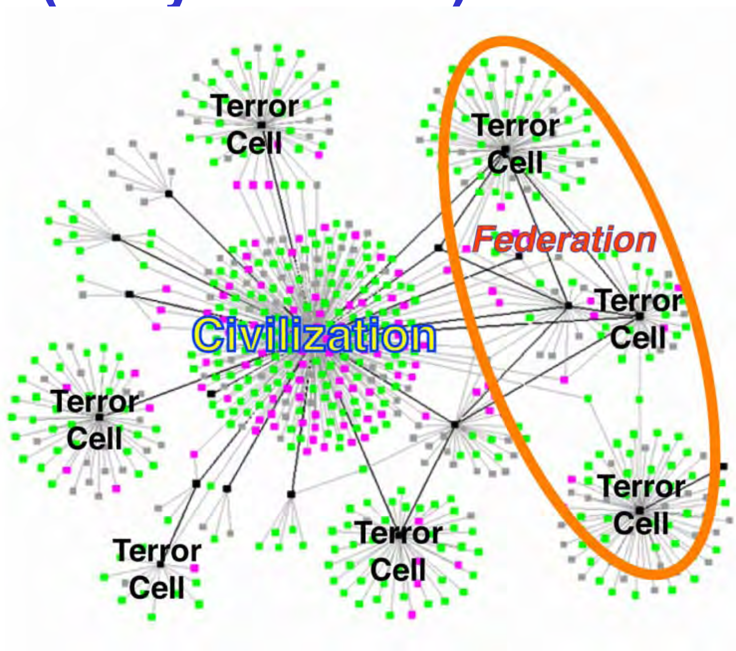
Competition



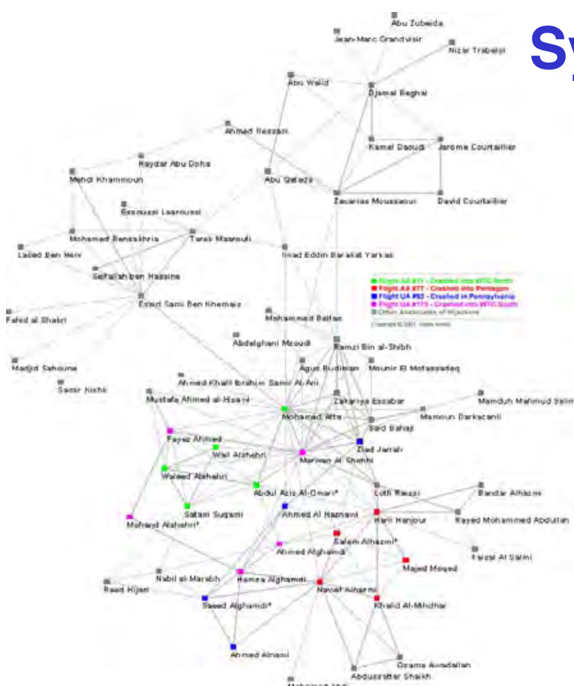
Conventional Conflict



Unconventional ("Asymmetric") Conflict

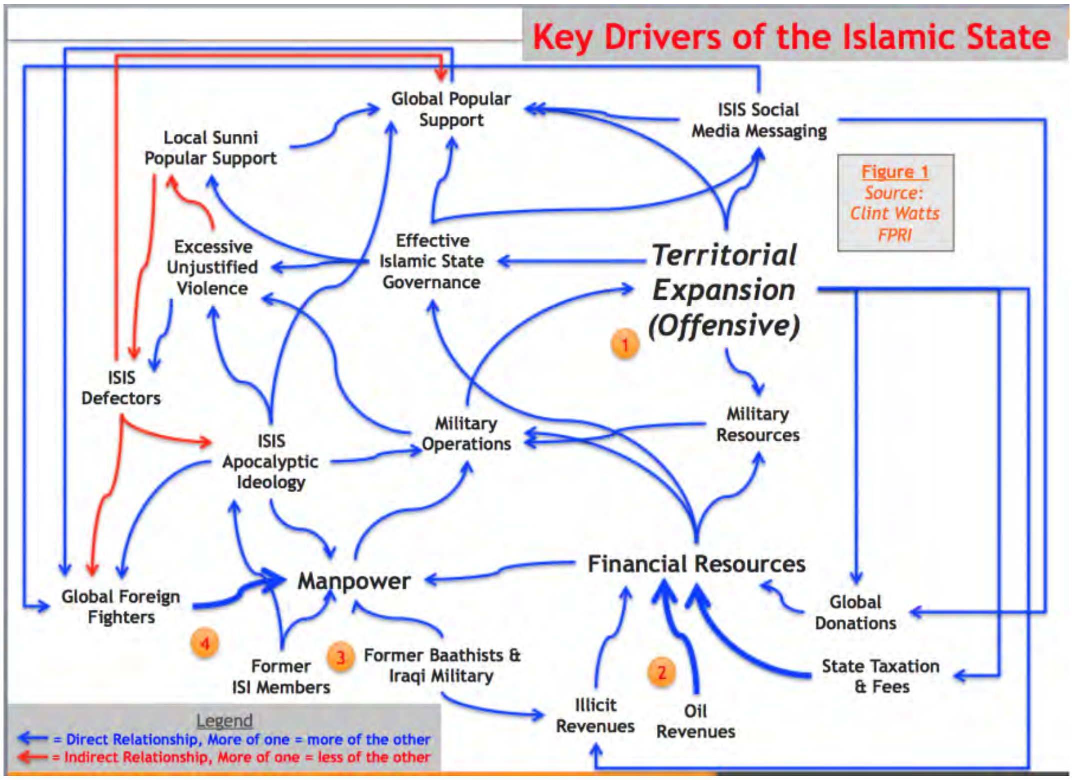


System Analysis of the 9/11 Terrorist Network



- Hijackers
  - –AA11
  - –AA77
  - –UA93
  - –UA175
- Accomplices

http://pear.accc.uic.edu/ojs/index.php/fm/article/view/941/863



Multi-Agent Scenarios Modeled as Optimal Control Problems

A Federated Optimization Problem

- Dynamic models for two agents, A and B, are coupled to each other and expressed as a single system

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\mathbf{u}(t) = \begin{bmatrix} \mathbf{F}_A & \mathbf{F}_B^A \\ \mathbf{F}_A^B & \mathbf{F}_B \end{bmatrix} \begin{bmatrix} \mathbf{x}_A \\ \mathbf{x}_B \end{bmatrix} + \begin{bmatrix} \mathbf{G}_A & \mathbf{G}_B^A \\ \mathbf{G}_A^B & \mathbf{G}_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix}$$

- Cost function minimizes performance-control tradeoff

$$E(J) = E\left\{ \frac{1}{2} \int_{t_o}^{t_f} \left[\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t) \right] dt \right\}$$

$$= E\left\{ \frac{1}{2} \int_{t_o}^{t_f} \left[\begin{bmatrix} \mathbf{x}_A^T & \mathbf{x}_B^T \end{bmatrix} \begin{bmatrix} \mathbf{Q}_A & \mathbf{Q}_B^A \\ \mathbf{Q}_A^B & \mathbf{Q}_B \end{bmatrix} \begin{bmatrix} \mathbf{x}_A \\ \mathbf{x}_B \end{bmatrix} + \begin{bmatrix} u_A^T & u_B^T \end{bmatrix} \begin{bmatrix} \mathbf{R}_A & \mathbf{R}_B^A \\ \mathbf{R}_A^B & \mathbf{R}_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix} \right] dt \right\}$$

- Optimal feedback control laws are coupled to each other

$$\mathbf{u}(t) = -\mathbf{C}\hat{\mathbf{x}}(t) = \begin{bmatrix} u_A \\ u_B \end{bmatrix} = -\begin{bmatrix} \mathbf{C}_A & \mathbf{C}_B^A \\ \mathbf{C}_A^B & \mathbf{C}_B \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_A \\ \hat{\mathbf{x}}_B \end{bmatrix}$$

A Distributed Optimization Problem

- Coupling between actions of two agents, A and B, is negligible

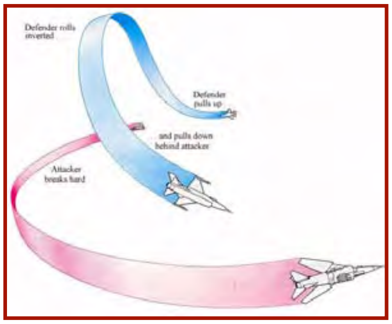
$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\mathbf{u}(t) = \begin{bmatrix} \mathbf{F}_A & 0 \\ 0 & \mathbf{F}_B \end{bmatrix} \begin{bmatrix} \mathbf{x}_A \\ \mathbf{x}_B \end{bmatrix} + \begin{bmatrix} \mathbf{G}_A & 0 \\ 0 & \mathbf{G}_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix}$$

$$E(J) = E\left\{ \frac{1}{2} \int_{t_o}^{t_f} \left[\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t) \right] dt \right\}$$

$$= E\left\{ \frac{1}{2} \int_{t_o}^{t_f} \left[\begin{bmatrix} \mathbf{x}_A^T & \mathbf{x}_B^T \end{bmatrix} \begin{bmatrix} \mathbf{Q}_A & 0 \\ 0 & \mathbf{Q}_B \end{bmatrix} \begin{bmatrix} \mathbf{x}_A \\ \mathbf{x}_B \end{bmatrix} + \begin{bmatrix} u_A^T & u_B^T \end{bmatrix} \begin{bmatrix} \mathbf{R}_A & 0 \\ 0 & \mathbf{R}_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix} \right] dt \right\}$$

- Each sub-system can be optimized separately
- Each control depends only on separate sub-state

$$\mathbf{u}(t) = -\begin{bmatrix} \mathbf{R}_A & 0 \\ 0 & \mathbf{R}_B \end{bmatrix}^{-1} \mathbf{G}^T\mathbf{S}\hat{\mathbf{x}}(t) = -\mathbf{C}\mathbf{x}(t) = \begin{bmatrix} u_A \\ u_B \end{bmatrix} = -\begin{bmatrix} \mathbf{C}_A & 0 \\ 0 & \mathbf{C}_B \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_A \\ \hat{\mathbf{x}}_B \end{bmatrix}$$



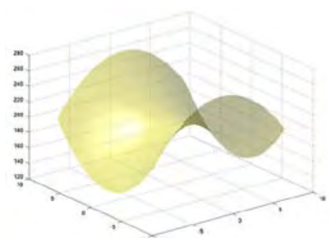
Pursuit-Evasion: A Competitive Optimization Problem

- **Pursuer's goal: minimize final miss distance**
- **Evader's goal: maximize final miss distance**

- **Linear model with two competitors, *P* and *E***

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\mathbf{u}(t) = \begin{bmatrix} \dot{\mathbf{x}}_P \\ \dot{\mathbf{x}}_E \end{bmatrix} = \begin{bmatrix} \mathbf{F}_P & 0 \\ 0 & \mathbf{F}_E \end{bmatrix} \begin{bmatrix} \mathbf{x}_P \\ \mathbf{x}_E \end{bmatrix} + \begin{bmatrix} \mathbf{G}_P & 0 \\ 0 & \mathbf{G}_E \end{bmatrix} \begin{bmatrix} u_P \\ u_E \end{bmatrix}$$

- **Example of a *differential game*, Isaacs (1965), Bryson & Ho (1969)**



Pursuit-Evasion: A Competitive Optimization Problem

- **Quadratic "minimax" (saddle-point) cost function**

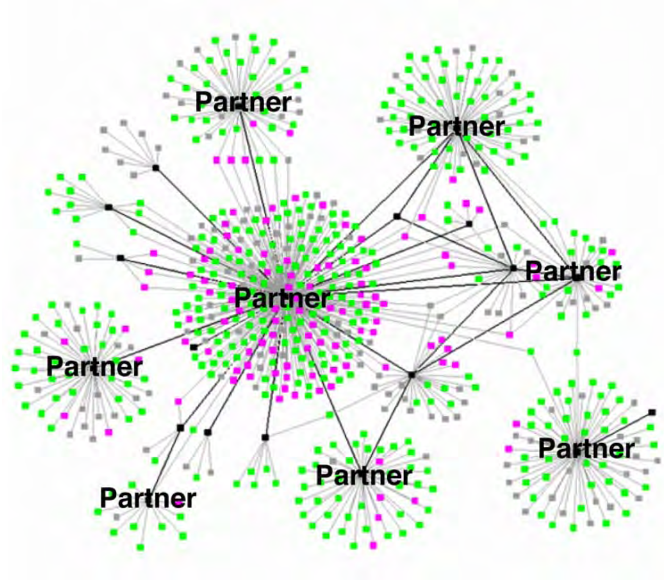
$$E(J) = E\left\{\frac{1}{2}\left[\mathbf{x}^T(t_f)\mathbf{S}(t_f)\mathbf{x}(t_f)\right] + \frac{1}{2}\int_{t_o}^{t_f}\left[\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)\right]dt\right\}$$

$$= E\left\{\begin{array}{l} \frac{1}{2}\begin{bmatrix} \mathbf{x}_P^T(t_f) & \mathbf{x}_E^T(t_f) \end{bmatrix} \begin{bmatrix} \mathbf{S}_P & \mathbf{S}_{PE} \\ \mathbf{S}_{EP} & \mathbf{S}_E \end{bmatrix}_f \begin{bmatrix} \mathbf{x}_P(t_f) \\ \mathbf{x}_E(t_f) \end{bmatrix} \\ + \frac{1}{2}\int_{t_o}^{t_f}\left[\begin{bmatrix} \mathbf{x}_P^T(t) & \mathbf{x}_E^T(t) \end{bmatrix} \begin{bmatrix} \mathbf{Q}_P & \mathbf{Q}_{PE} \\ \mathbf{Q}_{EP} & \mathbf{Q}_E \end{bmatrix} \begin{bmatrix} \mathbf{x}_P(t) \\ \mathbf{x}_E(t) \end{bmatrix} + \begin{bmatrix} u_P^T(t) & u_E^T(t) \end{bmatrix} \boxed{\begin{bmatrix} \mathbf{R}_P & 0 \\ 0 & -\mathbf{R}_E \end{bmatrix}} \begin{bmatrix} u_P(t) \\ u_E(t) \end{bmatrix}\right]dt \end{array}\right\}$$

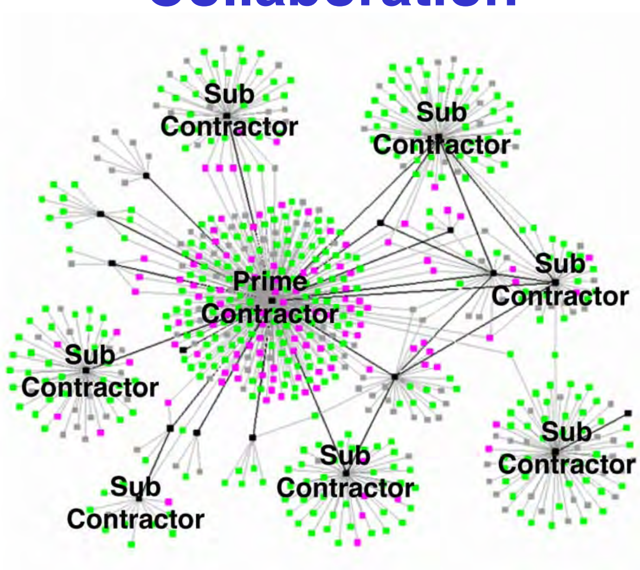
- **Optimal control laws for pursuer and evader**

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{u}_P(t) \\ \mathbf{u}_E(t) \end{bmatrix} = -\begin{bmatrix} \mathbf{C}_P(t) & \mathbf{C}_{PE}(t) \\ \mathbf{C}_{EP}(t) & \mathbf{C}_E(t) \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_P(t) \\ \hat{\mathbf{x}}_E(t) \end{bmatrix}$$

Coordination



Collaboration



Decomposition into Fast and Slow Models

Reduction of Dynamic Model Order

- Separation of high-order models into loosely coupled or decoupled lower order approximations

$$\begin{bmatrix} \Delta\dot{\mathbf{x}}_{fast} \\ \Delta\dot{\mathbf{x}}_{slow} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{fast} & \mathbf{F}_{slow}^{fast} \\ \mathbf{F}_{fast}^{slow} & \mathbf{F}_{slow} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_{fast} \\ \Delta\mathbf{x}_{slow} \end{bmatrix} + \begin{bmatrix} \mathbf{G}_{fast} & \mathbf{G}_{slow}^{\ \ fast} \\ \mathbf{G}_{fast}^{\ \ slow} & \mathbf{G}_{slow} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_{fast} \\ \Delta\mathbf{u}_{slow} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{F}_f & small \\ small & \mathbf{F}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_f \\ \Delta\mathbf{x}_s \end{bmatrix} + \begin{bmatrix} \mathbf{G}_f & small \\ small & \mathbf{G}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s \end{bmatrix}$$

Truncation of a Dynamic Model

- Dynamic model order reduction when
  - Two modes are only slightly coupled
  - Time scales of motions are far apart
  - Forcing terms are largely independent

$$\begin{bmatrix} \Delta\dot{\mathbf{x}}_f \\ \Delta\dot{\mathbf{x}}_s \end{bmatrix} = \begin{bmatrix} \mathbf{F}_f & \mathbf{F}_s^f \\ \mathbf{F}_f^s & \mathbf{F}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_f \\ \Delta\mathbf{x}_s \end{bmatrix} + \begin{bmatrix} \mathbf{G}_f & \mathbf{G}_s^f \\ \mathbf{G}_f^s & \mathbf{G}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{F}_f & small \\ small & \mathbf{F}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_f \\ \Delta\mathbf{x}_s \end{bmatrix} + \begin{bmatrix} \mathbf{G}_f & small \\ small & \mathbf{G}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s \end{bmatrix}$$
$$\approx \boxed{\begin{bmatrix} \mathbf{F}_f & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_s \end{bmatrix}} \begin{bmatrix} \Delta\mathbf{x}_f \\ \Delta\mathbf{x}_s \end{bmatrix} + \boxed{\begin{bmatrix} \mathbf{G}_f & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_s \end{bmatrix}} \begin{bmatrix} \Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s \end{bmatrix}$$

- Approximation: Modes can be analyzed and control systems can be designed separately

$$\boxed{\begin{aligned} \Delta\dot{\mathbf{x}}_f &= \mathbf{F}_f\Delta\mathbf{x}_f + \mathbf{G}_f\Delta\mathbf{u}_f \\ \Delta\dot{\mathbf{x}}_s &= \mathbf{F}_s\Delta\mathbf{x}_s + \mathbf{G}_s\Delta\mathbf{u}_s \end{aligned}}$$

Residualization of a Dynamic Model

- Dynamic model order reduction when
  - Two modes are coupled
  - Time scales of motions are separated
  - Fast mode is stable

$$\begin{bmatrix} \Delta\dot{\mathbf{x}}_f \\ \Delta\dot{\mathbf{x}}_s \end{bmatrix} = \begin{bmatrix} \mathbf{F}_f & \mathbf{F}_s^f \\ \mathbf{F}_f^s & \mathbf{F}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_f \\ \Delta\mathbf{x}_s \end{bmatrix} + \begin{bmatrix} \mathbf{G}_f & \mathbf{G}_s^f \\ \mathbf{G}_f^s & \mathbf{G}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{F}_f & small \\ small & \mathbf{F}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_f \\ \Delta\mathbf{x}_s \end{bmatrix} + \begin{bmatrix} \mathbf{G}_f & small \\ small & \mathbf{G}_s \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s \end{bmatrix}$$

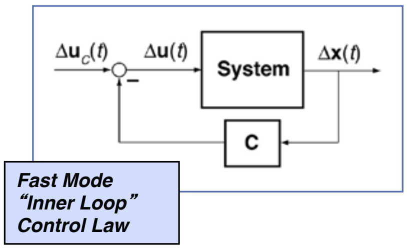
- Approximation: Motions can be analyzed separately using different "clocks"
  - Fast mode reaches steady state instantaneously on slow-mode time scale
  - Slow mode produces slowly changing bias disturbances on fast-mode time scale

Residualized Fast Mode

Fast mode dynamics

$$
\begin{aligned}
\Delta\dot{\mathbf{x}}_f &= \mathbf{F}_f\Delta\mathbf{x}_f + \mathbf{G}_f\Delta\mathbf{u}_f \\
&+ \left(\mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}^f{}_s\Delta\mathbf{u}_s\right)_{\sim Bias}
\end{aligned}
$$

If fast mode is not stable, it could be stabilized by "inner loop" control



$$
\begin{aligned}
\Delta\dot{\mathbf{x}}_f &= \mathbf{F}_f\Delta\mathbf{x}_f + \mathbf{G}_f\left(\Delta\mathbf{u}_c - \mathbf{C}_f\Delta\mathbf{x}_f\right) \\
&\quad + \left(\mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}_s^f\Delta\mathbf{u}_s\right)_{\sim Bias} \\
&= \left(\mathbf{F}_f - \mathbf{G}_f\mathbf{C}_f\right)\Delta\mathbf{x}_f + \mathbf{G}_f\Delta\mathbf{u}_{f_c} \\
&\quad + \left(\mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}_s^f\Delta\mathbf{u}_s\right)_{\sim Bias}
\end{aligned}
$$

Fast Mode in Quasi-Steady State

Assume that fast mode reaches steady state on a time scale that is short compared to the slow mode

$$
\begin{aligned}
0 &\approx \mathbf{F}_f\Delta\mathbf{x}_f + \mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}_f\Delta\mathbf{u}_f + \mathbf{G}_s^f\Delta\mathbf{u}_s \\
\Delta\dot{\mathbf{x}}_s &= \mathbf{F}_f^s\Delta\mathbf{x}_f + \mathbf{F}_s\Delta\mathbf{x}_s + \mathbf{G}_s\Delta\mathbf{u}_s + \mathbf{G}_f^s\Delta\mathbf{u}_f
\end{aligned}
$$

Algebraic solution for fast variable

$$
\begin{aligned}
0 &\approx \mathbf{F}_f\Delta\mathbf{x}_f + \mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}_f\Delta\mathbf{u}_f + \mathbf{G}_s^f\Delta\mathbf{u}_s \\
\mathbf{F}_f\Delta\mathbf{x}_f &= -\mathbf{F}_s^f\Delta\mathbf{x}_s - \mathbf{G}_f\Delta\mathbf{u}_f - \mathbf{G}_s^f\Delta\mathbf{u}_s \\
\Delta\mathbf{x}_f &= -\mathbf{F}_f^{-1}\left(\mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}_f\Delta\mathbf{u}_f + \mathbf{G}_s^f\Delta\mathbf{u}_s\right)
\end{aligned}
$$

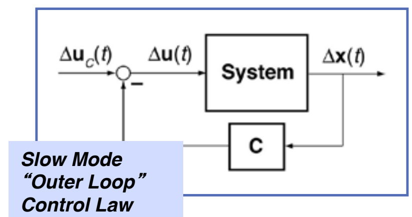
Residualized Slow Mode

Substitute quasi-steady fast variable in differential equation for slow variable

$$
\begin{aligned}
\Delta\dot{\mathbf{x}}_s &= -\mathbf{F}_f^s\left[\mathbf{F}_f^{-1}\left(\mathbf{F}_s^f\Delta\mathbf{x}_s + \mathbf{G}_f\Delta\mathbf{u}_f + \mathbf{G}^f{}_s\Delta\mathbf{u}_s\right)\right] + \mathbf{F}_s\Delta\mathbf{x}_s + \mathbf{G}_s\Delta\mathbf{u}_s + \mathbf{G}^s{}_f\Delta\mathbf{u}_f \\
&= \left[\mathbf{F}_s - \mathbf{F}_f^s\mathbf{F}_f^{-1}\mathbf{F}_s^f\right]\Delta\mathbf{x}_s + \left[\mathbf{G}_s - \mathbf{F}_f^s\mathbf{F}_f^{-1}\mathbf{G}_s^f\right]\Delta\mathbf{u}_s + \left[\mathbf{G}_f^s - \mathbf{F}_f^s\mathbf{F}_f^{-1}\mathbf{G}_f\right]\Delta\mathbf{u}_f
\end{aligned}
$$

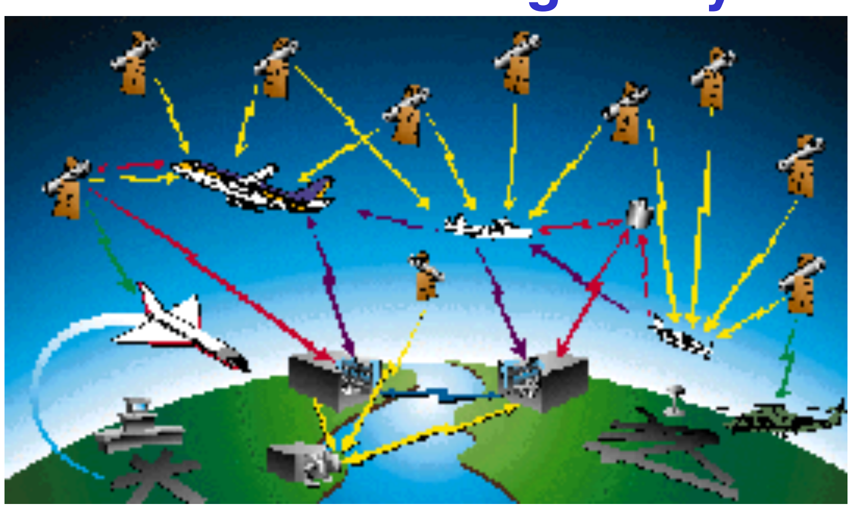
Residualized equation for slow variable

$$
\boxed{\Delta\dot{\mathbf{x}}_s = \mathbf{F}_{s_{NEW}}\Delta\mathbf{x}_s + \mathbf{G}_{s_{NEW}}\begin{bmatrix}\Delta\mathbf{u}_f \\ \Delta\mathbf{u}_s\end{bmatrix}}
$$



Control law can be designed for reduced-order slow model, assuming inner loop has been stabilized separately

Air Traffic Management: A Collaborative Multi-Agent System



https://www.flightradar24.com

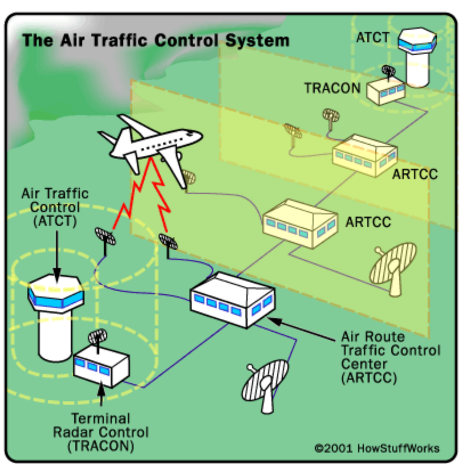
Elements of Principled Negotiation

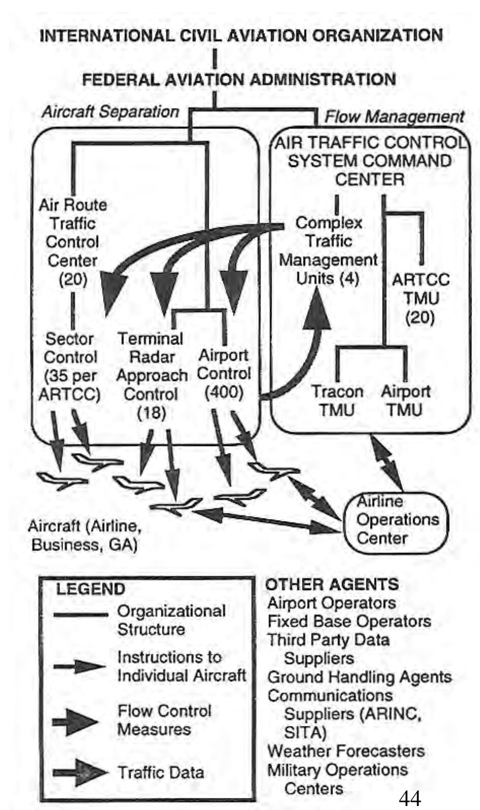
[Fisher, Ury (1981) Fry (1991)]

- Example of decision-making
- Separate agents* from the problem
- Focus on interests, not positions
- Invent options for mutual gain
- Insist on using objective criteria

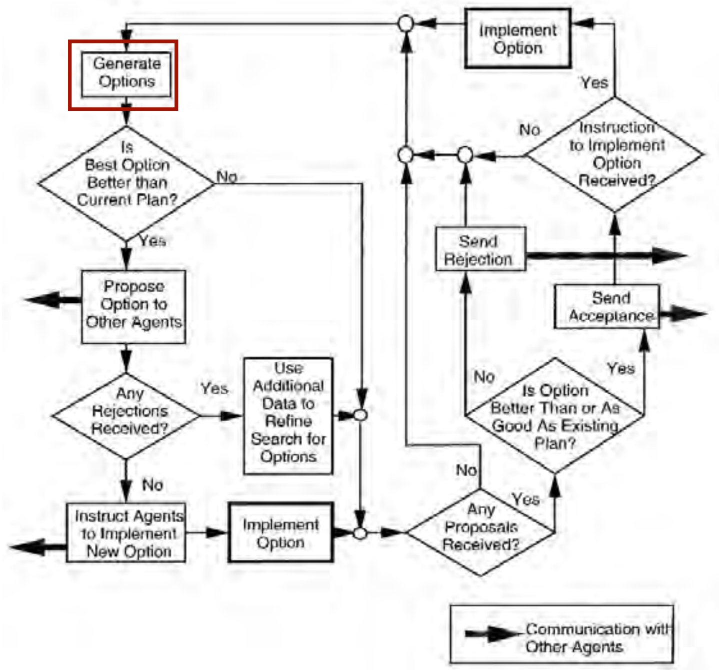
*** people, organizations, entities, …***

Intelligent Agents in Air Traffic Management



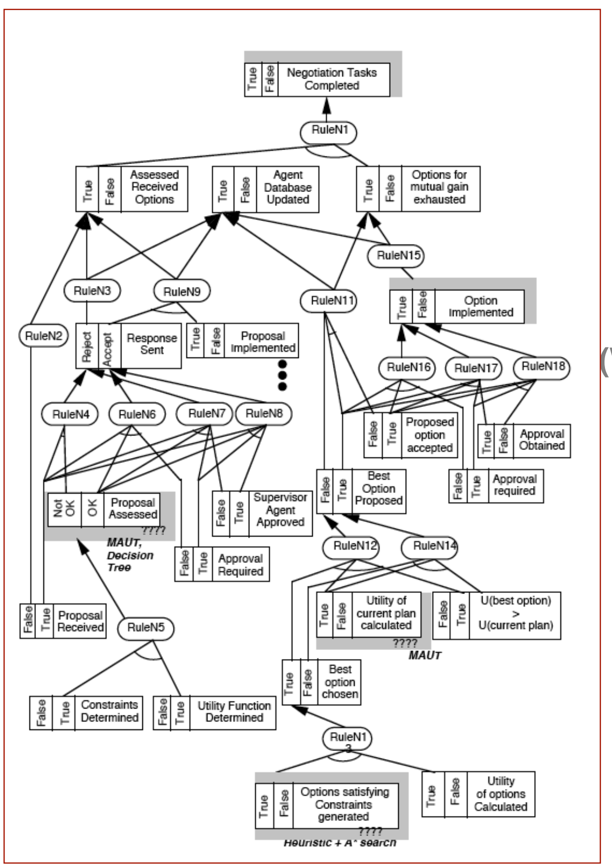


Principled Negotiation Flow Chart



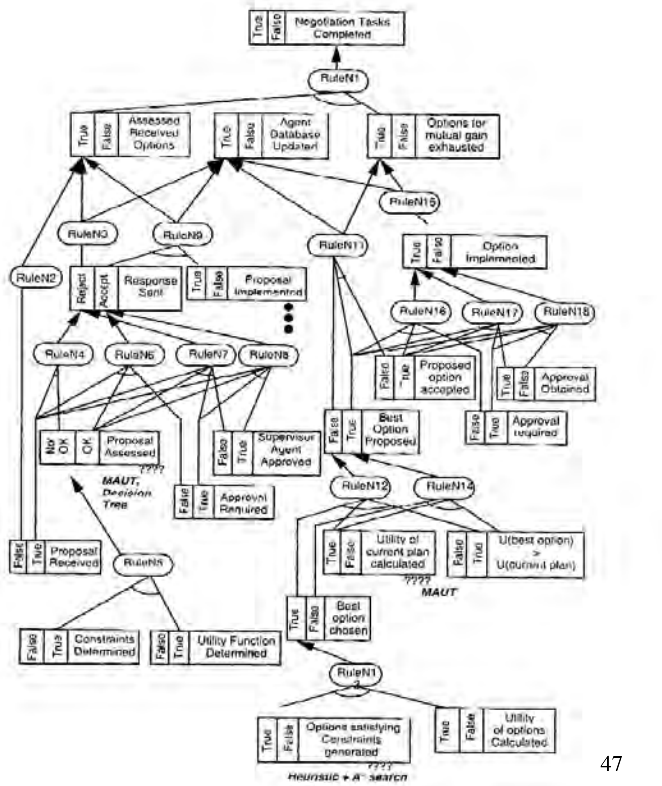
Expert System Diagram for Principled Negotiation

(Wangermann and Stengel)



- Separate agents* from the problem
- Focus on interests, not positions
- Invent options for mutual gain
- Insist on using objective criteria

Graphical Representation of Knowledge: **Principled Negotiation in Air Traffic Management**

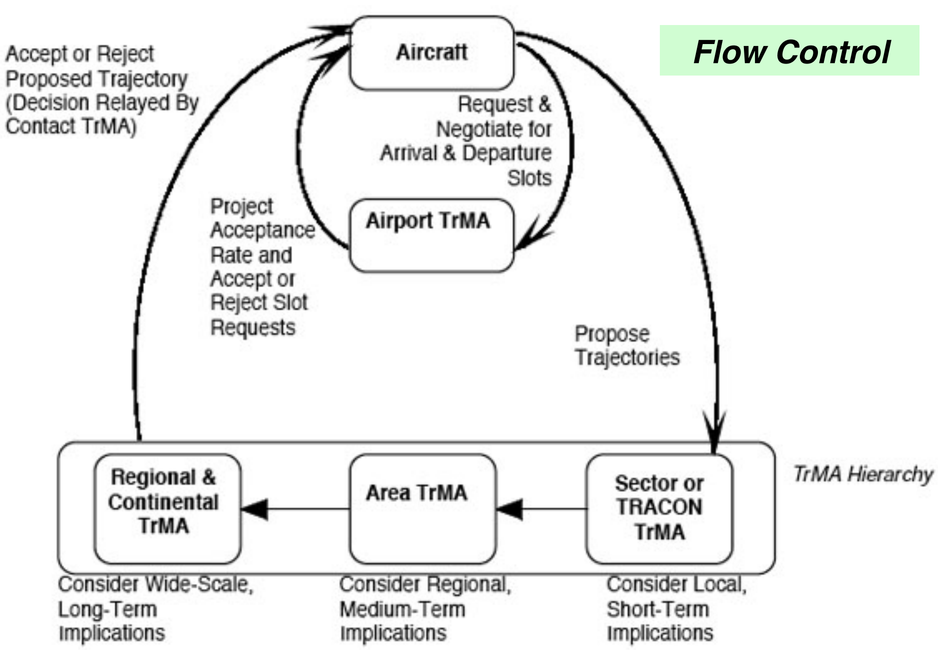


Principled Negotiation: Getting Past No (Ury, 1991)

- Prepare by identifying barriers to cooperation, options, standards, and your *Best Alternative to a Negotiated Agreement* (BATNA)
- Understand your goals, limits, and acceptable outcomes
- Buy time to think
- Know your “hot buttons”, deflect attacks
- Acknowledge opposing arguments
- Agree when you can without conceding
- Express your views without provoking
- “I” statements, not “you” statements
- Negotiate the rules of the game
- Reframe the negotiation
- Build a “golden bridge” that allows opponent to retreat gracefully
- Engage third-party mediation or arbitration
- Aim for mutual satisfaction, not victory
- Forge a lasting agreement

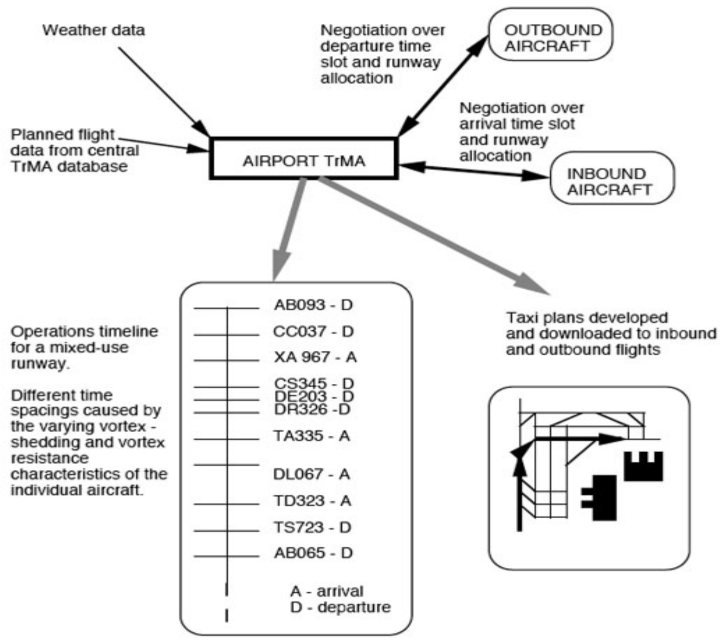
Supplementary Material

Intelligent Aircraft/Airspace System

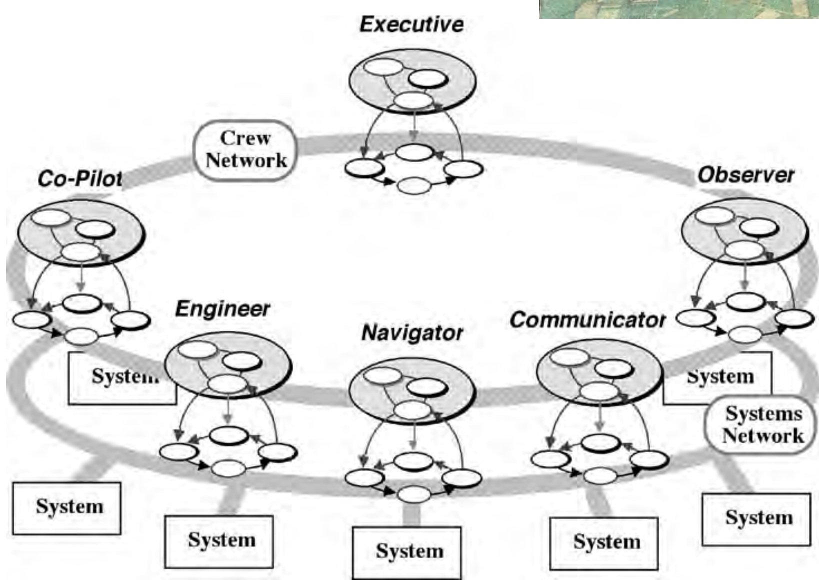


Intelligent Aircraft/Airspace System

Departure Control



A Cooperative Multi-Agent System



Multi-Agent Control Example Based on Linear-Quadratic-Gaussian (LQG) Optimal Control

- **Linear dynamic model**

$$\boxed{\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\mathbf{u}(t) + \mathbf{L}\mathbf{w}(t)}$$

- **Quadratic cost function**

$$E(J) = E\left\{ \begin{aligned} &\phi\left[\mathbf{x}(t_f)\right] + \int_{t_o}^{t_f} L\left[\mathbf{x}(t),\mathbf{u}(t)\right]dt \\ &\quad = \frac{1}{2}\left\{ \mathbf{x}^T(t_f)\mathbf{S}_f\mathbf{x}(t_f) + \int_{t_o}^{t_f}\left[\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)\right]dt \right\} \end{aligned} \right\}$$

Conclusion

- **Robots and Robotics**
  - ‘Mechanical’ devices
  - Design of ‘mechanical’ devices
  - Use of ‘mechanical’ devices
  - Control processes, sensors, and algorithms used in humans, animals, and machines
- **Intelligent Systems**
  - Systems to perform useful functions driven by goals and current knowledge
  - Systems that emulate biological and cognitive processes
  - Systems that process information to achieve objectives
  - Systems that learn by example
  - Systems that adapt to a changing environment
  - Optimization
- **Robots + Intelligent Systems = Intelligent Robotics**

MAE 345 Course Learning Objectives

- *Dynamics and control of robotic devices.*
- *Cognitive and biological paradigms for system design.*
- *Estimate the behavior of dynamic systems.*
- *Apply of decision-making concepts, including neural networks, expert systems, and genetic algorithms.*
- *Components of systems for decision-making and control, such as sensors, actuators, and computers.*
- *Systems-engineering approach to the analysis, design, and testing of robotic devices.*
- *Computational problem-solving, through thorough knowledge, application, and development of analytical software.*
- *Historical context within which robotics and intelligent systems have evolved.*
- *Global and ethical impact of robotics and intelligent systems in the context of contemporary society.*
- *Oral and written presentation.*