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
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

<table>
<thead>
<tr>
<th>Phase</th>
<th>Process</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective</td>
<td>Tactical (short-term)</td>
<td>Situation Assessment</td>
</tr>
<tr>
<td>Strategic (long-term)</td>
<td>Comprehension</td>
<td>Understanding</td>
</tr>
</tbody>
</table>

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

Important Dichotomies in Planning

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

“Knok-Knoks” and “Unk-Unks”
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”
**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

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

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

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

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
  \[
  \dot{x}(t) = Fx(t) + Gu(t) = \begin{bmatrix} F_A & F_B^A \\ F_A^B & F_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} + \begin{bmatrix} G_A & G_B^A \\ G_A^B & G_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix}
  \]

- Cost function minimizes performance-control tradeoff
  \[
  E(J) = \mathbb{E}\left\{ \frac{1}{2} \int_{t_0}^{T} (\dot{x}(t)^T Q x(t) + u(t)^T R u(t)) dt \right\}
  \]
  \[
  = \mathbb{E}\left\{ \frac{1}{2} \left[ x_0^T A x_0 + \int_{t_0}^{T} \left( x_0^T Q x_0 + u_0^T R u_0 \right) dt \right] \right\}
  \]

- Optimal feedback control laws are coupled to each other
  \[
  u(t) = -C\dot{x}(t) = \begin{bmatrix} u_A \\ u_B \end{bmatrix} = -\begin{bmatrix} C_A & C_B^A \\ C_A^B & C_B \end{bmatrix} \begin{bmatrix} \dot{x}_A \\ \dot{x}_B \end{bmatrix}
  \]

A Distributed Optimization Problem

- Coupling between actions of two agents, A and B, is negligible
  \[
  \dot{x}(t) = Fx(t) + Gu(t) = \begin{bmatrix} F_A & 0 \\ 0 & F_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} + \begin{bmatrix} G_A & 0 \\ 0 & G_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix}
  \]

- Each sub-system can be optimized separately
  \[
  E(J) = \mathbb{E}\left\{ \frac{1}{2} \int_{t_0}^{T} (\dot{x}(t)^T Q x(t) + u(t)^T R u(t)) dt \right\}
  \]
  \[
  = \mathbb{E}\left\{ \frac{1}{2} \left[ x_0^T A x_0 + \int_{t_0}^{T} \left( x_0^T Q x_0 + u_0^T R u_0 \right) dt \right] \right\}
  \]

- Each control depends only on separate sub-state
  \[
  u(t) = \begin{bmatrix} R_A & 0 \\ 0 & R_B \end{bmatrix}^{-1} G^T \dot{x}(t) = -Cx(t) = \begin{bmatrix} u_A \\ u_B \end{bmatrix} = -\begin{bmatrix} C_A & 0 \\ 0 & C_B \end{bmatrix} \begin{bmatrix} \dot{x}_A \\ \dot{x}_B \end{bmatrix}
  \]
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 $E$

\[
\dot{x}(t) = Fx(t) + Gu(t) = \begin{bmatrix} \dot{x}_P \\ \dot{x}_E \end{bmatrix} = \begin{bmatrix} F_P & 0 \\ 0 & F_E \end{bmatrix} \begin{bmatrix} x_P \\ x_E \end{bmatrix} + \begin{bmatrix} G_P & 0 \\ 0 & G_E \end{bmatrix} \begin{bmatrix} u_P \\ u_E \end{bmatrix}
\]

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

Pursuit-Evasion: A Competitive Optimization Problem

- Quadratic “minimax” (saddle-point) cost function

\[
E(J) = E\left\{\frac{1}{2} [x^T(t), S(t), x(t)] + \frac{1}{2} [\dot{x}^2(t)Qx(t) + u^T(t)Ru(t)] dt \right\}
\]

\[
= E\left\{\frac{1}{2} \begin{bmatrix} x_p^2(t) & x_e^2(t) \end{bmatrix} \begin{bmatrix} S_P & S_{PE} \\ S_{EP} & S_E \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \end{bmatrix} + \frac{1}{2} \begin{bmatrix} x_p^2(t) & x_e^2(t) \end{bmatrix} \begin{bmatrix} Q_P & Q_{PE} \\ Q_{EP} & Q_E \end{bmatrix} \begin{bmatrix} x_p(t) \\ x_e(t) \end{bmatrix} + \begin{bmatrix} u_p^2(t) \\ u_e^2(t) \end{bmatrix} \begin{bmatrix} R_p & 0 \\ 0 & R_e \end{bmatrix} \begin{bmatrix} u_p(t) \\ u_e(t) \end{bmatrix} \right\} dt
\]

- Optimal control laws for pursuer and evader

\[
u(t) = \begin{bmatrix} u_p(t) \\ u_E(t) \end{bmatrix} = - \begin{bmatrix} C_p(t) & C_{pE}(t) \\ C_{EP}(t) & C_E(t) \end{bmatrix} \begin{bmatrix} \hat{x}_p(t) \\ \hat{x}_E(t) \end{bmatrix}
\]
Decomposition into Fast and Slow Models

Reduction of Dynamic Model Order

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

\[
\begin{bmatrix}
\Delta x_{\text{fast}} \\
\Delta x_{\text{slow}}
\end{bmatrix} =
\begin{bmatrix}
F_{\text{fast}} & F_{\text{fast}} \\
F_{\text{slow}} & F_{\text{slow}}
\end{bmatrix}
\begin{bmatrix}
\Delta x_{\text{fast}} \\
\Delta x_{\text{slow}}
\end{bmatrix} +
\begin{bmatrix}
G_{\text{fast}} & G_{\text{fast}} \\
G_{\text{slow}} & G_{\text{slow}}
\end{bmatrix}
\begin{bmatrix}
\Delta u_{\text{fast}} \\
\Delta u_{\text{slow}}
\end{bmatrix}
+ \begin{bmatrix}
F_f & small \\
small & F_s
\end{bmatrix}
\begin{bmatrix}
\Delta x_{f} \\
\Delta x_{s}
\end{bmatrix} +
\begin{bmatrix}
G_f & small \\
small & G_s
\end{bmatrix}
\begin{bmatrix}
\Delta u_{f} \\
\Delta u_{s}
\end{bmatrix}
\]
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

\[
\begin{bmatrix}
\Delta x_f \\
\Delta x_s
\end{bmatrix} =
\begin{bmatrix}
F_f & F_s' \\
F_f' & F_s
\end{bmatrix}
\begin{bmatrix}
\Delta x_f \\
\Delta x_s
\end{bmatrix} +
\begin{bmatrix}
G_f & G_s' \\
G_f' & G_s
\end{bmatrix}
\begin{bmatrix}
\Delta u_f \\
\Delta u_s
\end{bmatrix}
\]

- \( \Delta x_f = F_f \Delta x_f + G_f \Delta u_f \)
- \( \Delta x_s = F_s \Delta x_s + G_s \Delta u_s \)

Residualization of a Dynamic Model

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

\[
\begin{bmatrix}
\Delta x_f \\
\Delta x_s
\end{bmatrix} =
\begin{bmatrix}
F_f & F_s' \\
F_f' & F_s
\end{bmatrix}
\begin{bmatrix}
\Delta x_f \\
\Delta x_s
\end{bmatrix} +
\begin{bmatrix}
G_f & G_s' \\
G_f' & G_s
\end{bmatrix}
\begin{bmatrix}
\Delta u_f \\
\Delta u_s
\end{bmatrix}
\]

- \( \Delta x_f = F_f \Delta x_f + G_f \Delta u_f \)
- \( \Delta x_s = F_s \Delta x_s + G_s \Delta u_s \)

- 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**

\[
\Delta \dot{x}_f = F_f \Delta x_f + G_f \Delta u_f \\
+ \left( F_s' \Delta x_s + G_s' \Delta u_s \right)_{Bias}
\]

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

\[
\Delta x_f = F_f \Delta x_f + G_f \left( \Delta u_c - C_f \Delta x_f \right) \\
+ \left( F_s' \Delta x_s + G_s' \Delta u_s \right)_{Bias}
\]

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

\[
0 = F_f \Delta x_f + F_s' \Delta x_s + G_f \Delta u_f + G_s' \Delta u_s
\]

\[
\Delta \dot{x}_s = F_s' \Delta x_s + F_f \Delta x_f + G_s \Delta u_s + G_f' \Delta u_f
\]

**Algebraic solution for fast variable**

\[
0 = F_f \Delta x_f + F_s' \Delta x_s + G_f \Delta u_f + G_s' \Delta u_s
\]

\[
F_f \Delta x_f = -F_s' \Delta x_s - G_f \Delta u_f - G_s' \Delta u_s
\]

\[
\Delta x_f = -F_f^{-1} \left( F_s' \Delta x_s + G_f \Delta u_f + G_s' \Delta u_s \right)
\]
Residualized Slow Mode

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

\[
\Delta x_s = -F_f^{\prime}\left(F_f^{\prime \prime}\Delta x_s + G_f \Delta u_f + G_f^{\prime} \Delta u_f \right) + F_s \Delta x_s + G_s \Delta u_s + G_f \Delta u_f
\]

Residualized equation for slow variable

\[
\Delta \dot{x}_s = F_{\text{new}} \Delta x_s + G_{\text{new}} \begin{bmatrix} \Delta u_f \\ \Delta u_s \end{bmatrix}
\]

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

- 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

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

Expert System Diagram for Principled Negotiation
(Wangermann and Stengel)
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**

Weather data

Planned flight data from central TIMA database

Negotiation over departure time slot and runway allocation

OUTBOUND AIRCRAFT

INBOUND AIRCRAFT

Taxi plans developed and downloaded to inbound and outbound flights

Operations timeline for a mixed-use runway

Different time spacings caused by the varying vortex-shedding and vortex resistance characteristics of the individual aircraft.

A - arrival
D - departure

A Cooperative Multi-Agent System

Co-Pilot
Crew Network
Executive
Observer
Communication
System
System
System
System
System
System
Multi-Agent Control Example Based on Linear-Quadratic-Gaussian (LQG) Optimal Control

- **Linear dynamic model**

\[
\dot{x}(t) = Fx(t) + Gu(t) + Lw(t)
\]

- **Quadratic cost function**

\[
E(J) = E\left\{ \phi[x(t_f)] + \int_{t_0}^{t_f} L[x(t), u(t)] dt \right\} = \frac{1}{2} \left[ x^T(t_f)Sx(t_f) + \int_{t_0}^{t_f} x^T(t)Qx(t) + u^T(t)Ru(t) dt \right]
\]

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.