Task Planning and Multi-Agent Systems
Robert Stengel
Robotics and Intelligent Systems, MAE 345, Princeton University, 2017

• Decision making
• Task decomposition, communities, and connectivity
• Cooperation, collaboration, competition, and conflict
• Single-agent path planning (see Lecture 6)
• Multi-agent architectures
• Swarm dynamics and control

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></td>
<td></td>
<td>Situation Awareness</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 $\leftarrow\rightarrow$ 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

- 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

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

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

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
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
Multi-Agent Scenarios Modeled as Optimal Control Problems

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
  \[
  E(J) = E \left\{ \phi[x(t_f)] + \int_{t_o}^{t_f} L[x(t), u(t)] \, dt \right\} \\
  = \frac{1}{2} \left\{ x^T(t_f)S_f x(t_f) + \int_{t_o}^{t_f} [x^T(t)Qx(t) + u^T(t)Ru(t)] \, dt \right\}
  \]
A Federated Optimization Problem

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

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

Cost function minimizes performance-control tradeoff

\[ E(J) = \mathbb{E} \left\{ \frac{1}{2} \int \begin{bmatrix} x_A^T & x_B^T \end{bmatrix} \begin{bmatrix} Q_A & Q_B^A \\ Q_A^B & Q_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} + \begin{bmatrix} u_A^T & u_B^T \end{bmatrix} \begin{bmatrix} R_A & R_B^A \\ R_A^B & R_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix} \right\} dt \]

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

\[ \dot{x}(t) = Fx(t) + Gu(t) = \begin{bmatrix} F_A & 0 \\ 0 & F_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} + \begin{bmatrix} G_A & 0 \\ 0 & G_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix} \]

\[ E(J) = \mathbb{E} \left\{ \frac{1}{2} \int \begin{bmatrix} x_A^T & x_B^T \end{bmatrix} \begin{bmatrix} Q_A & 0 \\ 0 & Q_B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} + \begin{bmatrix} u_A^T & u_B^T \end{bmatrix} \begin{bmatrix} R_A & 0 \\ 0 & R_B \end{bmatrix} \begin{bmatrix} u_A \\ u_B \end{bmatrix} \right\} dt \]

Each sub-system can be optimized separately

Each control depends only on separate sub-state

\[ u(t) = -\begin{bmatrix} R_A & 0 \\ 0 & R_B \end{bmatrix}^{-1} G^T S \dot{x}(t) = -C\dot{x}(t) = \begin{bmatrix} u_A \\ u_B \end{bmatrix} = -\begin{bmatrix} C_A & 0 \\ 0 & C_B \end{bmatrix} \begin{bmatrix} \dot{x}_A \\ \dot{x}_B \end{bmatrix} \]
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)

Quadratic “minimax” (saddle-point) cost function

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

\[ = E \left[ \frac{1}{2} \begin{bmatrix} x_P^T(t_f) & x_E^T(t_f) \end{bmatrix} \begin{bmatrix} S_P & S_{PE} \\ S_{EP} & S_E \end{bmatrix} \begin{bmatrix} x_P(t_f) \\ x_E(t_f) \end{bmatrix} \right] \]

\[ + E \left[ \frac{1}{2} \begin{bmatrix} x_P^T(t) & x_E^T(t) \end{bmatrix} \begin{bmatrix} Q_P & Q_{PE} \\ Q_{EP} & Q_E \end{bmatrix} \begin{bmatrix} x_P(t) \\ x_E(t) \end{bmatrix} + \begin{bmatrix} u_P^T(t) & u_E^T(t) \end{bmatrix} \begin{bmatrix} R_P & 0 \\ 0 & -R_E \end{bmatrix} \begin{bmatrix} u_P(t) \\ u_E(t) \end{bmatrix} \right] dt \]

Optimal control laws for pursuer and evader

\[ u(t) = \begin{bmatrix} u_P(t) \\ u_E(t) \end{bmatrix} = \begin{bmatrix} C_P(t) & C_{PE}(t) \\ C_{EP}(t) & C_E(t) \end{bmatrix} \begin{bmatrix} \dot{x}_P(t) \\ \dot{x}_E(t) \end{bmatrix} \]
Requirements for Guaranteeing Stability of the LQ Regulator

\[ \Delta \dot{x}(t) = F \Delta x(t) + G \Delta u(t) = [F - GC] \Delta x(t) \]

Closed-loop system is stable whether or not open-loop system is stable if ...

\[
\begin{align*}
Q &> 0 \\
R &> 0
\end{align*}
\]

... and \((F,G)\) is a controllable pair

\[
\text{Rank} \left[ \begin{array}{cccc}
G & FG & \cdots & F^{n-1}G \\
\end{array} \right] = n
\]

Coordination
Collaboration

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**
Supplementary Material

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.
Intelligent Aircraft/Airspace System

Flow Control

Accept or Reject Proposed Trajectory (Decision Made by Contact TRMA)

Request & Negotiate for Arrival & Departure Slots

Project Acceptance Rate and Accept or Reject Slot Requests

Propose Trajectories

Regional & Continental TRMA

Area TRMA

Sector or TRACON TRMA

Consider Wide-Scale, Long-Term Implications

Consider Regional, Medium-Term Implications

Consider Local, Short-Term Implications

TrMA Hierarchy

Intelligent Aircraft/Airspace System

Departure Control

Weather data

Negotiation over departure time slot and runway allocation

Planned flight data from central TRMA database

OUTBOUND AIRCRAFT

INBOUND AIRCRAFT

Operations timeline for a mixed-use runway.

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

Taxi plans developed and downloaded to inbound and outbound flights.

A8065 - D

CG037 - D

XA 967 - A

CS345 - D

DR835 - D

TA335 - A

DL047 - A

TD323 - A

TS723 - D

A8065 - D

A - arrival

D - departure
A Cooperative Multi-Agent System

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

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

\[
= \begin{bmatrix}
F_f & 0 \\
0 & F_s
\end{bmatrix}
\begin{bmatrix}
\Delta x_f \\
\Delta x_s
\end{bmatrix}
+ \begin{bmatrix}
G_f & 0 \\
0 & G_s
\end{bmatrix}
\begin{bmatrix}
\Delta u_f \\
\Delta u_s
\end{bmatrix}
\]

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

\[
\Delta \dot{x}_f = F_f \Delta x_f + G_f \Delta u_f
\]

\[
\Delta \dot{x}_s = F_s' \Delta x_s + G_s \Delta u_s
\]
Residualization of a Dynamic Model

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

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

- 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

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

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

\[
\Delta x_f = F_f \Delta x_f + G_f \left( \Delta u_s - C_x \Delta x_s \right)
+ \left( F_f' \Delta x_f + G_f' \Delta u_f \right)_{\text{bias}}
= \left( F_f - G_f C_x \right) \Delta x_f + G_f \Delta u_f
+ \left( F_f' \Delta x_s + G_f' \Delta u_s \right)_{\text{bias}}
\]
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

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

Algebraic solution for fast variable

\[ 0 \approx F_f \Delta x_f + F_s' \Delta x_s + G_f \Delta u_f + G_s' \Delta u_s \]
\[ F_f \Delta x_f = -F_s' \Delta x_s - G_f \Delta u_f - G_s' \Delta u_s \]
\[ \Delta x_f = -F_f^{-1} \left( F_s' \Delta x_s + G_f \Delta u_f + G_s' \Delta u_s \right) \]

Residualized Slow Mode

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

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

Residualized equation for slow variable

\[ \Delta x_s = F_{sNEW} \Delta x_f \quad + \begin{bmatrix} \Delta u_f \\ \Delta u_s \end{bmatrix} \quad G_{sNEW} \]

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