LECTURE 1: INTRODUCTION

Multiagent Systems
Based on "An Introduction to MultiAgent Systems" by Michael Wooldridge, John Wiley & Sons, 2002.
http://www.csc.liv.ac.uk/~mjw/pubs/imas/

Overview

- Five ongoing trends have marked the history of computing:
  - *ubiquity*;
  - *interconnection*;
  - *intelligence*;
  - *delegation*; and
  - *human-orientation*

Ubiquity

- The continual reduction in cost of computing capability has made it possible to introduce processing power into places and devices that would have once been uneconomic
- As processing capability spreads, sophistication (and intelligence of a sort) becomes ubiquitous
- What could benefit from having a processor embedded in it…?

Interconnection

- Computer systems today no longer stand alone, but are networked into large distributed systems
- The internet is an obvious example, but networking is spreading its ever-growing tentacles…
- Since distributed and concurrent systems have become the norm, some researchers are putting forward theoretical models that portray computing as primarily a process of interaction

Intelligence

- The complexity of tasks that we are capable of automating and delegating to computers has grown steadily
- If you don't feel comfortable with this definition of "intelligence", it's probably because you are a human

Delegation

- Computers are doing more for us – without our intervention
- We are *giving control* to computers, even in safety critical tasks
- One example: fly-by-wire aircraft, where the machine's judgment may be trusted more than an experienced pilot
- Next on the agenda: fly-by-wire cars, intelligent braking systems, cruise control that maintains distance from car in front…

Human Orientation

- The movement away from machine-oriented views of programming toward concepts and metaphors that more closely reflect the way we ourselves understand the world
- Programmers (and users!) relate to the machine differently
- Programmers conceptualize and implement software in terms of higher-level – more human-oriented – abstractions

Programming progression…

- Programming has progressed through:
  - machine code;
  - assembly language;
  - machine-independent programming languages;
  - sub-routines;
  - procedures & functions;
  - abstract data types;
  - objects;

to *agents*.

Global Computing

- What techniques might be needed to deal with systems composed of 10^{10} processors?
- Don't be deterred by its seeming to be "science fiction"
- Hundreds of millions of people connected by email once seemed to be "science fiction"…
- Let's assume that current software development models can't handle this…

Where does it bring us?

- Delegation and Intelligence imply the need to build computer systems that can act effectively on our behalf
- This implies:
  - The ability of computer systems to act *independently*
  - The ability of computer systems to act in a way that *represents our best interests* while interacting with other humans or systems

Interconnection and Distribution

- Interconnection and Distribution have become core motifs in Computer Science
- But Interconnection and Distribution, coupled with the need for systems to represent our best interests, implies systems that can *cooperate* and *reach agreements* (or even *compete*) with other systems that have different interests (much as we do with other people)

So Computer Science expands…

- These issues were not studied in Computer Science until recently
- All of these trends have led to the emergence of a new field in Computer Science: *multiagent systems*

Agents, a Definition

- An agent is a computer system that is capable of *independent* action on behalf of its user or owner (figuring out what needs to be done to satisfy design objectives, rather than constantly being told)

Multiagent Systems, a Definition

- A multiagent system is one that consists of a number of agents, which *interact* with one-another
- In the most general case, agents will be acting on behalf of users with different goals and motivations
- To successfully interact, they will require the ability to *cooperate*, *coordinate*, and *negotiate* with each other, much as people do

Agent Design, Society Design

- The course covers two key problems:
  - How do we build agents capable of independent, autonomous action, so that they can successfully carry out tasks we delegate to them?
  - How do we build agents that are capable of interacting (cooperating, coordinating, negotiating) with other agents in order to successfully carry out those delegated tasks, especially when the other agents cannot be assumed to share the same interests/goals?
- The first problem is *agent design*, the second is *society design* (micro/macro)

Multiagent Systems

- In Multiagent Systems, we address questions such as:
  - How can cooperation emerge in societies of self-interested agents?
  - What kinds of languages can agents use to communicate?
  - How can self-interested agents recognize conflict, and how can they (nevertheless) reach agreement?
  - How can autonomous agents coordinate their activities so as to cooperatively achieve goals?

Multiagent Systems

- While these questions are all addressed in part by other disciplines (notably economics and social sciences), what makes the multiagent systems field unique is that it emphasizes that the agents in question are *computational, information processing* entities.

The Vision Thing

- It's easiest to understand the field of multiagent systems if you understand researchers' vision of the future
- Fortunately, different researchers have different visions
- The amalgamation of these visions (and research directions, and methodologies, and interests, and…) define the field
- But the field's researchers clearly have enough in common to consider each other's work relevant to their own

Spacecraft Control

- When a space probe makes its long flight from Earth to the outer planets, a ground crew is usually required to continually track its progress, and decide how to deal with unexpected eventualities. This is costly and, if decisions are required *quickly*, it is simply not practicable. For these reasons, organizations like NASA are seriously investigating the possibility of making probes more autonomous — giving them richer decision making capabilities and responsibilities.
- *This is not fiction:* NASA*'s DS1 has done it!*

Deep Space 1

- http://nmp.jpl.nasa.gov/ds1/
- "Deep Space 1 launched from Cape Canaveral on October 24, 1998. During a highly successful primary mission, it tested 12 advanced, high-risk technologies in space. In an extremely successful extended mission, it encountered comet Borrelly and returned the best images and other science data ever from a comet. During its fully successful hyperextended mission, it conducted further technology tests. The spacecraft was retired on December 18, 2001." – NASA Web site

Autonomous Agents for specialized tasks

- The DS1 example is one of a generic class
- Agents (and their physical instantiation in robots) have a role to play in high-risk situations, unsuitable or impossible for humans
- The degree of autonomy will differ depending on the situation (remote human control may be an alternative, but not always)

Air Traffic Control

- "A key air-traffic control system…suddenly fails, leaving flights in the vicinity of the airport with no air-traffic control support. Fortunately, autonomous air-traffic control systems in nearby airports recognize the failure of their peer, and cooperate to track and deal with all affected flights."
- Systems taking the initiative when necessary
- Agents cooperating to solve problems beyond the capabilities of any individual agent

Internet Agents

- Searching the Internet for the answer to a specific query can be a long and tedious process. So, why not allow a computer program — an agent — do searches for us? The agent would typically be given a query that would require synthesizing pieces of information from various different Internet information sources. Failure would occur when a particular resource was unavailable, (perhaps due to network failure), or where results could not be obtained.

What if the agents become better?

- Internet agents need not simply search
- They can plan, arrange, buy, negotiate – carry out arrangements of all sorts that would normally be done by their human user
- As more can be done electronically, software agents theoretically have more access to systems that affect the real-world
- But new research problems arise just as quickly…

Research Issues

- How do you state your preferences to your agent?
- How can your agent compare different deals from different vendors? What if there are many different parameters?
- What algorithms can your agent use to negotiate with other agents (to make sure you get a good deal)?
- These issues aren't frivolous – automated procurement could be used massively by (for example) government agencies
- The Trading Agents Competition…

Multiagent Systems is Interdisciplinary

- The field of Multiagent Systems is influenced and inspired by many other fields:
  - Economics
  - Philosophy
  - Game Theory
  - Logic
  - Ecology
  - Social Sciences
- This can be both a strength (infusing well-founded methodologies into the field) and a weakness (there are many different views as to what the field is about)
- This has analogies with artificial intelligence itself

Some Views of the Field

- *Agents as a paradigm for software engineering*: Software engineers have derived a progressively better understanding of the characteristics of complexity in software. It is now widely recognized that *interaction* is probably the most important single characteristic of complex software
- Over the last two decades, a major Computer Science research topic has been the development of tools and techniques to model, understand, and implement systems in which interaction is the norm

Some Views of the Field

- *Agents as a tool for understanding human societies*:
  Multiagent systems provide a novel new tool for simulating societies, which may help shed some light on various kinds of social processes.
- This has analogies with the interest in "theories of the mind" explored by some artificial intelligence researchers

Some Views of the Field

- *Multiagent Systems is primarily a search for appropriate theoretical foundations:*
  We want to build systems of interacting, autonomous agents, but we don't yet know what these systems should look like
- You can take a "neat" or "scruffy" approach to the problem, seeing it as a problem of *theory* or a problem of *engineering*
- This, too, has analogies with artificial intelligence research

Objections to MAS

- Isn't it all just Distributed/Concurrent Systems? There is much to learn from this community, but:
- Agents are assumed to be autonomous, capable of making independent decision – so they need mechanisms to synchronize and coordinate their activities at run time
- Agents are (can be) self-interested, so their interactions are "economic" encounters

Objections to MAS

- Isn't it all just AI?
- We don't need to solve all the problems of artificial intelligence (i.e., all the components of intelligence) in order to build really useful agents
- Classical AI ignored *social* aspects of agency. These are important parts of intelligent activity in real-world settings

Objections to MAS

- Isn't it all just Economics/Game Theory? These fields also have a lot to teach us in multiagent systems, but:
- Insofar as game theory provides *descriptive* concepts, it doesn't always tell us *how* to compute solutions; we're concerned with computational, resource-bounded agents
- Some assumptions in economics/game theory (such as a rational agent) may not be valid or useful in building artificial agents

Objections to MAS

- Isn't it all just Social Science?
- We can draw insights from the study of human societies, but there is no particular reason to believe that artificial societies will be constructed in the same way
- Again, we have inspiration and cross-fertilization, but hardly subsumption