1

Intelligent Autonomous Agents
ICS606 / EE606
Spring 2010

Nancy E. Reed

nreed@hawaii.edu

2

Lecture #9 An Overview of Learning in Multi-Agent Systems

- References
  - Weiss – Ch. 6, sec 1-3
  - Russell & Norvig – Ch. 18-21
  - Wooldridge – Ch. 8
- Some figures © MIT Press/Prentice Hall
- Some figures © Morgan Kaufman

3

Outline

- General Introduction and Characterization
  - Differencing Features
  - The credit-assignment problem
- Learning and activity coordination
  - Reinforcement learning
    - Q-Learning
    - Learning classifier systems
    - Isolated, concurrent reinforcement learning
  - Interactive Reinforcement learning of Coordination
    - Action Estimation Algorithm (ACE)
    - Action Group Estimation Algorithm (AGE)
- Layered Learning in RoboCup Soccer (separate slides)
- http://www.cs.utexas.edu/~pstone/thesis/

4

Introduction

- What is Learning?
  - Acquisition of **new knowledge, motor** and **cognitive** skills
  - **Incorporation** of the acquired knowledge and skills **in future system activities**
  - **Conducted by the system** itself
  - Leads to an **improvement in** the system's **performance**
  - Centralized or decentralized (next)
- Note: many techniques are used for learning in systems without agents, and aren't covered here

5

Learning in Multiagent Systems

ML(machine learning)
Learning in stand-alone systems

(distributed AI) DAI
Distributed multiagent systems whose structural organization and functional behavior is fixed

- Perspectives
  - Learning and activity coordination
  - Learning about and from other agents
  - Learning and communication

6

Characterization of Learning in Multiagent Systems

- Broad learning categories:
  - Centralized (or isolated) learning – single agent
  - Decentralized (or interactive) learning – multiple agents
- Details:
  1. The degree of decentralization
  2. Interaction-specific features
  3. Involvement-specific features
  4. Goal-specific features
  5. The learning method
  6. The learning feedback

1. The Degree of Decentralization

- How distributed – how many processes?
- How parallel – how many simultaneous processes?

2. Interaction-Specific Features

- The level of interaction
  - Separate --- negotiation
- The persistence of interaction
  - Short --- long
- The frequency of interaction
  - Low --- high
- The pattern of interaction
  - Unstructured --- hierarchical
- The variability of interaction
  - Fixed --- changing

3. Involvement-Specific Features

- The relevance of involvement
- Role played during involvement (generalist / specialist)

4. Goal-Specific Features

- The type of improvement the learning is to achieve
- The compatibility of the learning goals pursued by the agents

5. The Learning Method

- **Rote** learning (repetition)
- Learning from **instruction** and by advice taking
- Learning from **examples** and by practice
- Learning by **analogy**
- Learning by **discovery**

6. Learning Feedback

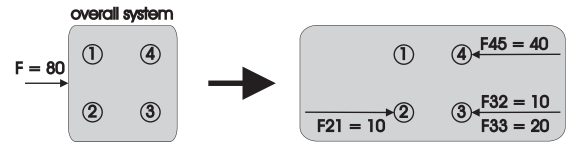
- Unsupervised learning – finding clusters/groupings, data mining. Labels and/or categories may not be known
- Reinforcement learning – from a series of reinforcements +/-. utility of position measures its “goodness”, better positions are encouraged, worse ones are penalized
- Supervised learning – correct result is known during training (agent gets input-output exemplars)
- Semi-supervised learning – give some labels and try to label a much larger number of examples. There may be error in labels and/or categories

The Credit-Assignment Problem (CAP)

CAP - properly assign feedback (credit or blame) for an overall performance change to each activity that contributed to that change

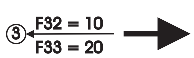
- **Inter-agent** CAP (what action carried out by what agent contributed to what extent to the performance change)
- **Intra-agen**t CAP (what knowledge, inferences and decisions within an agent led to an action)

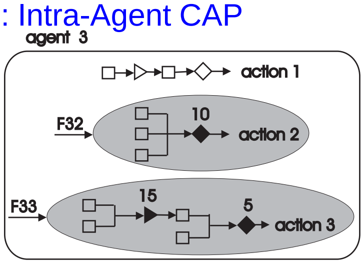
Example Inter-Agent Credit Assignment Problem



- 4 agent system
- 5 actions with total F = 80
- Decomposition based on actions of each agent:
- $\mathbf{F_{i,j}}$ - **i^th agent** and **j^th external action**

Example: Intra-Agent CAP





- Agent **3's role**
- **3** actions
  - □ internal knowledge
  - ◇ decisions
  - ▷ inferences

Which knowledge is used for each part of each action?

In act1, equal weight for all (serial)

In act2, equal weight for all (parallel)

In act3, inference 15 and decision 5

Learning and Activity Coordination

- Agents can learn to share resources and maximize profits
- Initial agent coordination methods began breaking down in dynamic environments
- To effectively coordinate activities in dynamic environments, **agents must learn** about other agents and about the environment itself

Reinforcement Learning

- Successor states are chosen based on current state information and the desire to maximize the reinforcement/feedback received after each action.

- Q-Learning
- Learning Classifier Systems

Q-Learning

- Learn an action-utility function
- Maximizes the reinforcement *over time* of an agent's actions
  - Avoids local maxima
- Finds a decision policy that maximizes the expected value of the reward over all possible states
- Decision policy -- a function that estimates the long-term rewards for each state-action pair

Learning Classifier Systems

- Rule-based system [if X then Y]
- All rules have an associated strength, or weight
- Learning is by adjusting rule strengths based on environmental feedback
- Classifiers are described by a (condition, action) pair
- $S_t(c_i,a_i)$ = strength of classifier *i* at time t.

Classifier Systems (cont.)

- Initialize weights to a default value
- Environment feedback is matched up with the classifier's rules
  - A classifier is chosen and that classifier's action is executed
  - Feedback from the environment based on this action dictates the change in weight
  - Strengths are adjusted using the Bucket Brigade Algorithm (BBA)

Bucket Brigade Algorithm

- Strengthens the classifier leading to the previous action, and also the one before that ("parent")
- penultimate action gets a fraction of the strength of that given to the ultimate one
  - considered to be taken from the ultimate strength allocation
- By back-propagating strength, reinforcement over time is optimized (like Q-Learning)

CIRL: Concurrent, Isolated Reinforcement Learning

- Agents choose actions based on environmental feedback
- Decisions optimize rewards based on the agent's current perception/internal state
  - Other agents in the environment not modeled
  - No domain-specific information is necessary

CIRL: Concurrent, Isolated Reinforcement Learning (cont.)

- Advantages to CIRL:
  - No domain knowledge necessary
  - No communication overhead required
  - Possible with both cooperative and non-cooperative situations
- CIRL may be ineffective when:
  - Agent actions are strongly coupled/dependent
  - Feedback concerning agent actions is delayed
  - There is only one or a few combinations of agent actions that will optimally solve the task

Interactive Reinforcement Learning

- Using communication to decide individual and group actions
- Agents "see" actions being considered by other agents
- Eliminate from consideration actions incompatible with others
- Assumes all agents are working toward the same goal

Interactive Reinforcement Learning (2)

- Two variants of the BBA are used to coordinate group activities:
  - Action Estimation Algorithm (ACE)
  - Action Group Estimation Algorithm (AGE)
- Both ACE and AGE use communication to inform agent actions

ACE

- Each agent generates a set of all possible actions that it can execute in its current state
- The goal relevance for all actions is calculated
- For all actions with relevance above a certain threshold, the agent announces a bid for that action
- The action with the highest bid is executed and incompatible ones are no longer considered
- Repeat until all actions are selected or eliminated
- Actions selected form the activity context
- If the sequence of actions in the activity context receives a payoff at the end, it is distributed equally among all the member actions

AGE

- All possible actions of all the agents are generated
- All possible sets of mutually compatible actions (activity contexts) are saved
- For each activity context, bids are collected from each agent on its actions in that particular activity context
- Activity context with highest sum of bids is selected and executed
- Total payoff is distributed among all actions in the activity context

Interactive Reinforcement Learning (cont.)

- Both ACE and AGE shown to "enable agents to learn coordinated behavior in the sense that the agents were able to much more effectively solve problems compared to random action selection" (p. 272).
- AGE produced slightly better results, but with higher computational cost

Summary

- Learning in multi-agent systems
  - Differencing Features
  - The credit-assignment problem
- Learning and activity coordination
  - Reinforcement learning
    - Q-Learning
    - Learning classifier systems
    - Isolated, concurrent reinforcement learning
  - Interactive Reinforcement Learning
    - Action Estimation Algorithm (ACE)
    - Action Group Estimation Algorithm (AGE)

Questions