CogSysI Lecture 12: Multi-Agent Planning

Intelligent Agents

Ute Schmid (lecture)

Emanuel Kitzelmann (practice)

Applied Computer Science, Bamberg University

last change: July 7, 2008

Working Together

- Sharing of tasks, of information, dynamic coordination of activities, distributed problem solving and planning
- Differences to distributed systems
  - Agents in MAS might not share common goals: encounters resemble games where agents must act strategically to achieve a preferred outcome
  - Autonomous agents make decisions at run-time; in classical distributed systems, coordination and cooperation are hard-wired during design-time

CDPS

Cooperative, distributed problem solving

- Historically:
  - System with nodes, working together to solve problems beyond their individual capacities
  - Each node is capable of problem solving and can work independently, but the problems cannot be solved without cooperation (expertise, resources, information)
  - Benevolence assumption: agents implicitly share a common goal, there is no potential conflict
  - Simplifies design: agents need only to worry about the overall utility

CDPS in MAS

- Societies of self-interested agents
  - Conflicts of interest, as in human societies
  - Need of cooperation to achieve (individual) goal
  - Problems: Reasons for and kinds of cooperation, techniques for recognitizing and resolving conflicts, negotiation and making compromises
- Distinguish from parallel problem solving: problem decomposition into independently solvable subproblems

Coherence and Coordination

- Two criteria for evaluating MAS
- Coherence:
  - How well does the MAS behave as a unit
  - Measured in terms of solution quality, efficiency of resource usage, degrading in the presence of uncertainty or failure
- Coordination
  - Degree to which agents can avoid extraneous activity, such as synchronizing activities
  - Poor coordination leeds to conflicts between agents resulting in destructive interactions

Main focus of research in MAS

Main Issues in CDPS

- Dividing problems into smaller tasks to distribute between agents
- Synthesis of a problem solution from sub-problem results
- Optimization of problem-solving activities, maximization of coherence
- Techniques to coordinate agents activities

Durfee and Lesser, 1987; Weiß, 1993

Stages in CDPS

- Problem decomposition
  - hierarchically, until granularity is appropriate such that sub-problem can be solved by individual agent
    extreme case: decompose until only atomic actions are left
    high overhead of managing interactions
  - performed often by a single agent with knowledge about the task structure
  - Knowledge about the capabilities of agents necessary for “suitable” decomposition
- Subproblem solution
  typically involves information sharing between agents
- Solution synthesis
  might be hierarchical

Task Sharing and Result Sharing

- Task Sharing
  - Allocation of sub-problems to different agents
  - Homogenous agents: any task to any agent
  - Otherwise: Reaching agreements by negotiation
- Result Sharing
  - Sharing information relevant to sub-problems
  - proactively or reactively (by communication)

Task Sharing in CNETS

- Contrat Net (CNET) protocol: Smith and Davis, 1980
- Process of how companies organize putting contracts out to tender as metaphor
- Task announcement
  - general broadcast, limited broadcast, or point-to-point
  - announcer becomes task manager
- Listeners evaluate announcements and make bids for suitable tasks
- Manager selects a node/agent and communicates an award message to the succesful bidders (which become contractors)

Further CNET Processes

- Special case: direct contract without announcement (option of refusal)
- Request and information processing: Request causes an inform message to be sent to the requestor

CNET is the most implemented and best-studied framework for CDP

Result Sharing

Improving group performance:

- Confidence: cross-checking independently derived solutions
- Completeness: Share local views to achieve a better global view
- Precision: share results to ensure precision of overall solution
- Timeliness: sharing can result in quicker derivations

Combining Task and Result Sharing

- in the cooperating expert system FELINE (Wooldridge et al. 1991)
- Agents with expertise in distinct, but related areas
- Each agent: independent rule-based system
  - database containing information about the current state
  - collection of rules encoding domain knowledge
  - representation of beliefs about itself and its environment (all "acquaintances")
  - Attributes: skills and interests (represented by identifiers)

FELINE cont.

- Communication
  - Triples: sender, receiver, content (as message type, attribute, value)
  - Message types: request, response, inform
- Check whether a current node is a skill of another agent, if yes, request and wait for response
- If a new fact is generated, check whether any agent has this as interest, if yes, inform

Inconsistency

- Because of autonomy, inconsistencies can arise
  - in beliefs (represented information about the “world”)
  - in goals/intentions (things wanted to be achieved)
- Sources of inconsistency
  - limited access to environment
  - faulty sensors
  - faulty information sources (other agents)

Handling Inconsistency

- Not allow or ignore: CNET approach, only view of “manager” matters
- Resolve through negotiation (too much communicational and computational overhead)
- Graceful degradation

Functionally accurate/cooperative (FA/C) Systems

- Problem solving is opportunistical and incremental (no strict order, take advantage and piece together what you get at a time)
- Exchange high-level results not raw-data
- Resolve uncertainty and inconsistency during the problem solving process by exchanging and comparing partial solutions
- Solutions should not be constrained to a single route

Coordination

- Managing interdependencies between agents' activities
- Main approaches:
  - Partial global planning
  - Joint intentions
  - Mutual modeling
  - Norms and social laws
- at run-time

Examples

- You and I want to leave the room through a single door
  one resource which can only used by one agent at the same time (door)
- I intend to submit a grant proposal and need your signature
  dependence of activities
- I obtain an interesting document and pro-actively make a copy for you
  Increase utility of another agent (non-requested)

Positive, Non-Requested Relationships

- Action equality: two agents want to perform an action, it is enough, if one performs the action and saves the other the effort (doing the dishes)
- Consequence: Action has a side-effect contributing to goals of another agent
- Favour: Action facilitates acievement of goals for other agent (e.g. achieving some necessary preconditions)

Partial Global Planning

- Lesser et al.
- Testbed for MAS: distributed vehicle monitoring testbed (DVMT)
- Track a number of vehicles passing within a range of distributed sensors
- Process information as rapidly as possible
- PGP principle: information exchange to reach common conclusion
- partial: system cannot generate a plan for the entire problem
- global: exchanging local plans, achieve non-local view
- Three iterating stages
  - Agents goals and short-term plans to achieve
  - Information exchange to determine where plans and goals interact
  - Alter local plans for better coordination

Partial Global Plans

- Meta-level structure, to prevent incoherence
- Which agent should exchance information with which under what conditions (*)
- Cooperatively generated partial global plan
  - Objective: larger goal of the system
  - Activity maps: what agents are doing to what results
  - Solution construction graph (*)
- Extensions to Generalized partial global plans in a new testbed (Decker, 1996)

Joint Intentions

- Use of human team-work models
- Distinguish coordinated action from cooperative coordinated action
  e.g. people run to a tree because it is raining vs. as part of a choreography
- Cooperative coordination defined in the notion of a joint persistent goal (Levesque et al. 1990)
- “mental state” of agents: goal and motivation
- System: ARCHON (Jennings, 1993)

Joint Persistent Goal

- Initially, every agent does not believe that goal φ is satisfied, but believes that φ is possible
- Every agent i has a goal of φ until the termination condition is satisfied
- Termination condition:
  - goal φ is satisfied, or
  - goal φ is impossible to achieve, or
  - motivation/justification for Ψ for goal φ is no longer present

Joint Persistent Goal cont.

- Until the termination condition is satisfied, do
  - If any agent i believes that goal φ is achieved, then it will have as goal that this becomes a mutual belief, and will retain this goal until the termination condition is satisfied
  - if any agent i believes that goal φ is impossible, then it will have as goal that this becomes a mutual belief, and will retain this goal until the termination condition is satisfied
  - if any agent i believes that the motivation Ψ for the goal is no longer present, then it will have as goal that this becomes a mutual belief, and will retain this goal until the termination condition is satisfied

Teamwork Model of CDPS

- Recognition: of potential for cooperation (can not/want not achieve goal alone) and believe that there exist some group of agents which can achive the goal
- Team Formation: soliciting assistance; agreement to the ends to be achieved (not the means); agents are assumed to be rational (only commit if they belief that the goal is reachable)
- Plan Formation: agreement about course of action (by negotiation or communication)
- Team Action: execusion of joint action (convention which each agent follows, e.g. by JPG)

Mutual Modeling

- Genesereth et al., 1986
- Game-theoretic models: if both you and the other agents share a common view of the scenario (payoff matrix), then you can do a game-theoretical analysis to determine which is the rational thing for each agent to do
  $\hookrightarrow$ Cooperation without communication

MACE

MACE (Grasser et al., 1987): agents have acquaintance knowledge about other agents

- Class (agents are organized in structured groups, identified by a class name)
- Name (unique agent name within a class)
- Roles (description of the part the agent plays in the class)
- Skills (what agent knows about capabilities of other modelled agents)
- Goals (what agent knows about goals of other modelled agents)
- Plans (what agent knows about how other modelled agents achieve goals)

Example Addition Agent

- Models itself $\hookrightarrow$ this is how other agents will perceive its skills, goals, etc.)
- Models an decomposition agent which knows how an expression is decomposed into simpler components
- Models an simple-plus agent which knows how to add two numbers

Norms and Social Laws

- Norm: established pattern of behavior
  e.g., forming a queue at a bus stop, allowing the persons coming first, to enter the bus first
- Not enforced: if violated, no consequence: but: convention to regulate behavior
- Social law, similar, but associated with some authority
- Templates to structure action repertoire
- Balance between individual freedom and goal of the agent society
- Simplify decision processes, by dictating a course of action
- Most of our social nature is dependent on convention (starting with natural language)

Two Approaches

- Offline desing: hardwired into agents, simpler, but not possible in complex systems where not all characteristics are known at design time
- Emergence from within the system

Emergent Norms

- How to come to a global agreement on the use of social conventions by using only locally available information?
- First investigated by Shoham and Tennenholtz, 1992
- Coming to an agreement about t-shirt color (red or blue)
- Each agent wears a random color initially
- How can it be reached that all wear the same color in the end?
- Agents meet as pairs, strategy: keep own color or change

Strategy Update Functions

- Simple majority: Change, if other strategy was observed more often than own strategy
- Simple majority with agent types: Include “confidence” (shared memory) of agents of one type
- Simple majority with communication on success: boradcast only memory related to the succesfull strategy
- Highest cummulation reward: presupposed ability so “see” that a given strategy gives a particular payoff

Efficiency of convergence: how many rounds need to be played

- Strategy change might be expensive (e.g., change of computer system)
- Strategy change might danger system stability

Evaluation

- Highest cummulative reward
- For any value ϵ with $0 \leq \epsilon \leq 1$, there exists some bounded value n such that an agreement is reached in n rounds with probability $1 - \epsilon$
- Strategy update is stable: once reached, agents will not diverge from norm
- Strategy update is efficient: guarantees each agent a payoff no worse than with the initial strategy

Planning and Synchronization

(Durfee, 1999)

- Centralized planning for distributed plans: Master plans, slaves execute their part
- Distributed planning: Specialist agents contribute parts to a global plan, other agents will execute
- Distributed planning for distributed plans: cooperation to form individual plans, dynamic coordination of activities (coordination problems, resolvable by negotiation)

Plan Merging

- Georgeff (1983): Algorithm for merging a set of plans into a conflict free Not necessarily optimal) multiagent plan
- STRIPS with an additional DURING list, set of conditions which need to hold while an action is performed
- Plan as set of states, action as function from states to states
- Three synchronization steps:
  - Interaction analysis: where do single agent plans interact?
  - Safety analysis: identify harmful actions
  - Interaction resolution

Interaction Analysis

- Satisfiability of two actions: there exists some sequence which may be executed without invalidating preconditions of one or both
- Commutativity: if two actions might be performed in parallel
- Precedence: one action generates postconditions needed as preconditions of the other

Safty Analysis

- First remove all harmless actions from the plan (no interaction, or actions commute)
- Secondly, generate the set of all harmful interactions
- Identifying “critical sections” where mutual exlusiveness must be guaranteed