

Distributed Problem Solving

- How do the agents work together so that they can attain the goals?
- Requires:
 - Coherence: The agents have to want to work together.
 - Competence: The agents have to know how to work well together.

Distributed Problem Solving

- Why use distributed problem solving?
 - Speed up the solution process.
 - Expertise and problem solving abilities may be inherently distributed.
 - The knowledge to solve the problem may be distributed.
 - The solution may need to be distributed in order for the solution to be executed.

Task Sharing

- An agent cannot complete all tasks, so other agents assist him.
 - 1. Task decomposition
 - 2. Task allocation
 - 3. Task accomplishment
 - 4. Result synthesis
- ToH example is provided in the text. Section 3.3.1.

Task Sharing

- What are the differences when applying task sharing to a system of homogeneous agents versus a system of heterogeneous agents?

Heterogeneous Task Sharing

- How does one handle the task distribution when all agents with the required capabilities are busy?
 - Rebroadcast announcement for bids.
 - Revise the announcement and rebroadcast.
 - Determine an alternative task decomposition.

Result Sharing

- It is possible that multiple agents may solve the same problem but attain different results.
 - Agents can learn by sharing their results and therefore improve the system's performance.

Result Sharing

- The system's performance can be improved by:
 - Confidence: Higher confidence in a result if the agents independently derive the same result.
 - Completeness: A broader range of solutions provides a more complete solution.
 - Precision: Individual agents are able to refine their result based upon the results of the other agents.

Results Sharing

- Timeliness: If the agents work together to formulate the solution in parallel, the result can be attained quicker.

Functionally Accurate Cooperation

- Agents share partial results potentially refining previous results.
 - Iterative process.
 - Eventually, an agent will have enough information to formulate a result.
 - High communication load and possibly computational expense.
 - Distraction occurs when too many results are shared and all agents migrate towards the same problem-solving techniques.

Shared Repositories

- Backboard architecture can be used.
- Permits searching of alternative solutions.
- Agents do not know whose solution constraints are affected by the agent's solution choices.
 - Hence the need for a shared repository

Shared Repositories

- Agents can relax the constraints if necessary.
 - Implies heuristics to control the distributed search exist.

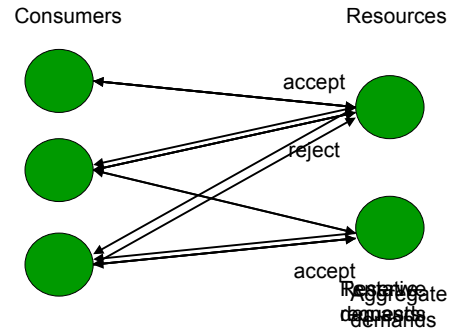
Negotiated Search

- Negotiated search is usually used with Shared repositories.
- The typical operators for solving problems are:
 - *initiate-solution*: propose a new start point
 - *extend-solution*: revise existing partial solution.
 - *critique-solution*: solution viability
 - *relax-solution-requirement*: modify the requirements for solution acceptance.

DCHS

- Distributed Constrained Heuristic Search (DCHS).
 - Assign an agent to each resource and permit the agent to process the requests for the resource.
 - Distributed resource allocation.
 - Market mechanisms: auctions
 - Aggregate the requests

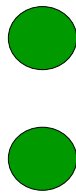
DCHS



DCHS

Consumers

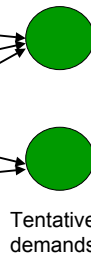
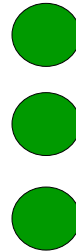
Resources

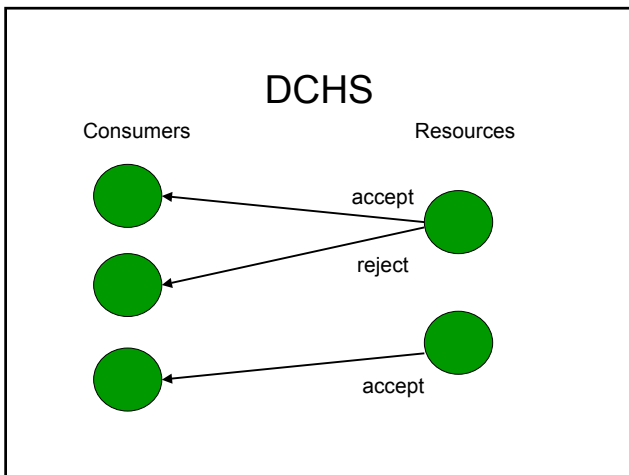
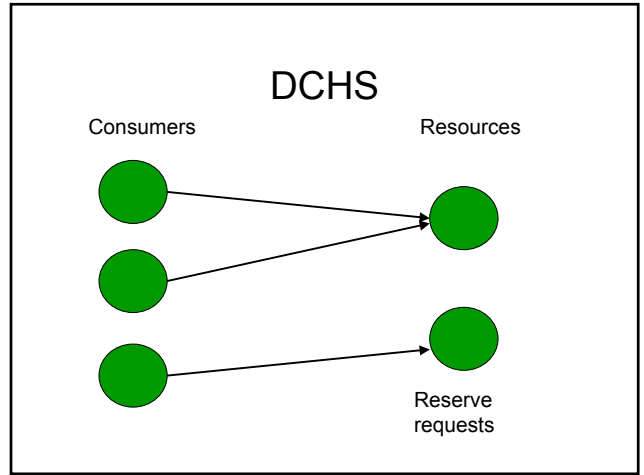
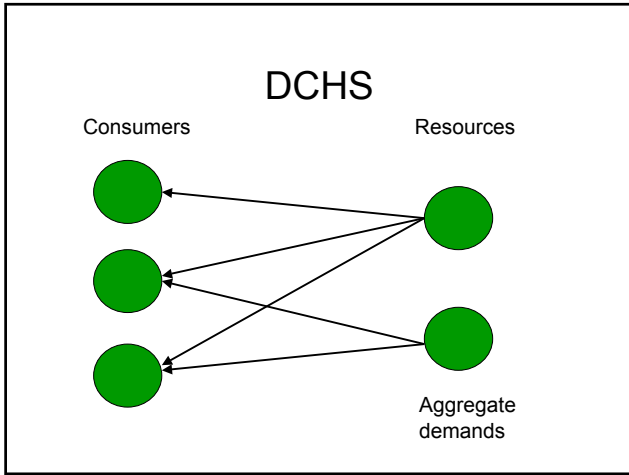


DCHS

Consumers

Resources





Organizational Structuring

- An alternative strategy is to apply an organizational structure when solving a problem.
 - Physical proximity, related capabilities, required but unrelated capabilities, etc.
 - Defines roles, responsibilities, and preferences that determine the control and communication patterns.

Organizational Structuring

- The structure indicates each agent's capabilities and prioritization of such capabilities.
 - Permits overlapping responsibilities that can increase probability of success.
- Agents only learn about partial results that affect their assigned task.
 - Structure based communication.
- Agents only need knowledge of their local substructure.
 - What does the agent do, who does it communicate with, who does it report too, etc.

Organizational Structuring

- Organizational structuring is usually implemented using stored pattern-response rules.
 - A partial result matches a pattern, then the rule is executed.
 - May be multiple rules executed from a single partial result.

Communication Strategies

- The organizational structure concept does not handle timing issues.
 - Can lead to messages that agents will not use for a long period of time.
 - Clutters memory and potentially distracts agent.
 - Ensures that the information is available to all in a timely fashion.

Communication Strategies

- If cluttering memory is not a bad thing and distractions are short, then send all partial results.
 - Otherwise, only send partial results when the generating agent has finished using the information.

Communication Strategies

- An alternative:
 - Communicate results only when requested.
 - Two communication delays, request and reply
- What about communication reliability?
 - Acknowledgement
 - Predict recipient reaction and watch for the reaction.

Planning

- Find all possible actions that will lead to the desired result and determine in what order to execute the actions.
 - Essentially a search problem of the possible world states.
 - “Find some or all possible orderings of the actions that would result in achieving the specified goal, given the constraints of the world.” - Gerogeff

Planning

- Typical search issues from heuristic search, including A* algorithms, apply to planning.
 - Sub-optimal node expansion (not closest to goal)
 - Random search (slow development)

Traditional AI Planning

- Very knowledge-intensive
 - Organize pieces of knowledge or partial plans to create a plan.
- A plan is not complete until all of the goals can be represented as a set of primitive actions.
 - Total-order
 - STRIPS, INTERPLAN
 - or partial-order
 - NOAH, NONLIN, MOLGEN, SIPE

Traditional AI Planning

- Issues related to generation of complete plans before plan execution.
- Issues related to partial plan generation during plan execution.
 - Reactive planning
 - Procedural Reasoning System (PRS)
 - Hierarchical planning
 - ABSTRIPS, NOAH, NONLIN, MOLGEN

Frame Problem

- The Frame Problem
 - “The specification of what is true in one state of the world and exactly what is changed by performing some action in the world.” – Luger
 - What happens as the “world” becomes larger?

Frame Axiom

- A **Frame Axiom** is a rule that attempts to predict which predicates are not changed by an action and therefore remain the same in the new description of the world.

STRIPS

- **STRIPS** – **ST**anford **R**esearch **I**nstitute **P**lanning **S**ystem.
 - Employed to drive Shakey.
 - Early 1970’s
 - Provided an efficient representation and implementation of operations in a planner.

STRIPS

- Requires representation of the initial world, the incremental changes to the world, and the goal state of the world.

STRIPS

- Define actions (operators) in terms of preconditions, delete list and add list.
 - Preconditions represent the initial world at the time the action begins.
 - Delete list represents the facts about the world that are no longer true.
 - Add list represents the facts about the world that are now true.

STRIPS

- Pickup(x)
 - Preconditions: $\text{OnTable}(x) \wedge \text{HandEmpty} \wedge \text{Clear}(x)$
 - Delete list: $\text{OnTable}(x), \text{HandEmpty}, \text{Clear}(x)$
 - Add list: $\text{Holding}(x)$

STRIPS

- Issues:
 - Add and delete lists must specify everything necessary to satisfy the frame axiom.
 - Attempting to solve sub-problems independently when in fact they are dependent.
 - Can lead to difficult or impossible searches.
- Available planners:<http://www.csc.ncsu.edu/faculty/stamant/planning-resources.html>

STRIPS

- PlanIt

Distributed Planning

- A combination of traditional AI planning and distributed problem solving.
 - Either creation of the plan is centralized but the result is a plan that can be distributed,
 - or the creation of the plan is distributed but the completion of the plan is not.
 - or the create of the plan is distributed and the plan completion is also distributed.

Traditional AI Issues

- Traditional AI makes many assumptions that permit the techniques to work:
 - Planner is omniscient
 - The actions are deterministic and instantaneous.
 - The goals are fixed and categorical
 - The environment is static.
 - “The agent is the only source of change in the environment.”

Traditional AI Issues

- “The goals presented to the agent remain unchanged throughout the process of planning and execution.”
- “The actions that the agent can perform can be modeled as instantaneous state transducers.”
 - “They have neither temporal extent nor fixed times of occurrence.”
- Pollack and Horty. *There's More to Life Than Making Plans: Plan Management in Dynamic, Multiagent Environments.*

Addressing the Issues

- Addressing the issues
 - Omniscience:
 - conditional plan generation.
 - Deterministic Actions:
 - Including representations of actions with probabilistic outcomes.
 - Categorical Goals:
 - Utility-based and decision-theoretic planning systems.
 - Creates plans with “rich, time-dependent utility functions.”

Addressing the Issues

- Omniscience, Deterministic Actions, and Categorical Goals:
 - “Modeling planning as a (fully or partially observable) Markov decision process (MDP).”
- Static Environment:
 - Planning under uncertainty, incorporates changes into the predicted outcomes.
- Dynamic Environments:
 - Reactive planners

Addressing the Issues

- Which goals when:
 - Deliberation scheduling
- Pollack and Horty. *There's More to Life Than Making Plans: Plan Management in Dynamic, Multiagent Environments.*

What is needed

“Autonomous agents in dynamic, multiagent environments also need to be able to manage the plans they generate. They need to determine which planning problems and opportunities to consider in the first place. They need to be able to weigh alternative incomplete plans and decide among competing alternatives. They need to be able to form incomplete plans now, adding detail later, and thus, they need to be able to decide how much detail to include now and when to add more detail. They need to be able to integrate plans with one another and to decide when to treat an existing plan as an inflexible commitment and when, instead, to consider modifications of it. They also need to be able to do this in a way that comports with the inherent bounds on their computational resources.”

- Pollack and Horty. *There's More to Life Than Making Plans: Plan Management in Dynamic, Multiagent Environments.*

Centralized Planning & Distributed Plans

- A centralized planner is employed to develop plans that contain distributable actions.
 - Partial order planning, may require synchronization.
 - Really a variation of the decompose-allocate-execute-synthesize algorithm discussed for task sharing.
 - Want the plan that can be most efficiently and effectively decomposed and distributed.

Centralized Planning & Distributed Plans

- Algorithm:
 1. Generate a partial order plan based upon goal, operator set, and initial world state.
 2. Decompose the plan
 - Maintain ordering relationship between steps within subplans and minimize the relationships across subplans.
 3. Insert required synchronization into subplans.

Centralized Planning & Distributed Plans

4. Allocate subplans to appropriate agents.
 - If fail, go back to step 2.
5. Initiate and monitor plan execution.

Centralized Planning & Distributed Plans

- Issue:
 - If the environment has unreliable or slow communications, then may not want to distribute subplans to multiple agents.
 - Consider plans requiring fewer agents.
 - Cost of plan distribution and synchronization for large plans.

Distributed Planning & Centralized Plans

- Distribute planning process but execute in a centralized fashion.
 - Planning requires cooperative planning specialists.
 - Similar to task-sharing and results-sharing.
- Agents may share partial plans that can be merged into a complete plan using negotiation.
 - Results-sharing.

Distributed Planning & Distributed Plans

- Distribute planning process and plan execution.
 - Most challenging.
 - The plan may be distributed, never existing as a complete plan residing at one location.
 - The partial plans must be compatible so that agents do not have conflicts while executing plans and are also able to help each other achieve their plans.

Plan Merging

- Plan merging refers to the case in which agents create individual plans and they must be executed to complete the overall plan.
 - Conflicts
 - Identify and resolve

Plan Merging

- Centralized plan coordination:
 - Single agent collects and analyzes all individual plans.
 - Find conflicts and modify to remove.
 - Reachability analysis
 - Potentially intractable problem

Plan Merging

- Centralized plan coordination:
 - Constrain the search space based upon STRIPS concept of preconditions.
 1. Determine the interactions between pairs of actions to be executed by different agents.
 - May lead to a stricter ordering
 2. Identify the set of unsafe situations.
 3. Insert the required synchronization.

Plan Merging

- Centralized plan coordination:
 - Plan Synchronization techniques
 - Communication
 - Temporal constraints: scheduling
 - Precedence ordering
 - Issues:
 - Individual plans may not consider coordination

Iterative Plan Formation

- Control local plans by requiring agents to consider larger spaces of plans based upon global constraints.
 - Agents construct all feasible plans.
 - All feasible plans are searched to construct overall plan.

Plan Combination Search

- Begin with the set of all possible plans and then refine the possible plans to converge on a nearly optimal subset.
 - Agents determine which plan to use and determine synchronization constraints on actions.

Distributed Hierarchical Planning

- Concentrates on the hierarchical nature of the plan space.
 - Each agent starts with an abstract plan.
 - Agents refine their plans and communicate the changes with the other agents and all agents look for conflicts.
 - Continues until a synchronized set of detailed plans is created.

Hierarchical Behavior-Space Search

- Each agent represents the local plans at multiple abstraction levels, any of which can be used to resolve all conflicts.
 - The protocol chooses an abstraction level to work with and then determines if conflicts are resolved at this level or passed to more detailed levels.
 - Conflicts are resolved using a distributed constraint satisfaction search.

Hierarchical Behavior-Space Search

- Method terminates iff there are finite abstraction levels, and the agents make changes to their plans such that they cannot get into cyclic plan generation patterns.

Hierarchical Behavior-Space Search

- Advantage
 - Reduces the amount of search, therefore faster and fewer messages.
- Disadvantage
 - The coordination constraints at an abstract level may impose unnecessary limits on more detailed actions.
 - More inefficient coordination solutions at the abstract level.

Distributed Planning and Negotiation

- There will be situations in which multiple agents have conflicts and there is no clear resolution mechanism.
 - Negotiation may be used.

Distributed Planning

- Revisit PlanIt.

Distributed Plan Representation

- Typically assume that all agents in a system use the same representation and interpretation mechanisms.
 - There is no standard representation (like KQML).
 - Closest thing to a standard is STRIPS operators.

Distributed Plan Representation

- Cypress
 - Attempt to define a standard
 - SRI

Distributed Plan Representation

- ACT
 - Name: unique label
 - Cue: goals
 - Preconditions: required world states
 - Setting: World-state features bound to ACT variables.
 - Resources: required resources for execution
 - Properties: properties associated with the ACT
 - Comment: documentation
 - Plot: partially ordered sequence of goals/actions to be completed.

Post Planning Coordination

- What happens if at least on agent's plan fails?
 - Contingency plan: alternative branches
 - More complicated to merge into an overall plan.
 - Monitoring and replanning
 - Stops all agents if there is a problem
 - Requires a plan-coordinate-execute cycle.
 - Plan repair
 - Local vs system

Pre-planning Coordination

- Is it possible to coordinate before the planning?
 - Social laws:
 - “A prohibition against particular choices of action in particular contexts.” – Weiss chap 3.
 - Want to determine the constraints that prevent undesirable states while permitting the agents to enter acceptable or desirable states.

Interleaving Plan, Coordination and Execution

- Partial Global Planning (PGP) provides a dynamic capability to revise plans in a cost-effective manner.
 - Best used in applications that require some uncoordinated activity.
 - Does assume communication over time.

Interleaving Plan, Coordination and Execution

- PGP steps:
 - Task Decomposition:
 - Assumes tasks are inherently decomposed.
 - Agents may not know when beginning to develop a plan what tasks other agents are doing or plan to do and the relationship to the agent's plan.
 - No individual agent may be aware of the global task or state. Coordination allows them to develop an awareness to complete their task.

Interleaving Plan, Coordination and Execution

- Local Plan Formation:
 - Agent must first understand what goal it is to accomplish and what actions it may take to reach the goal.
 - Purely reactive agents cannot use PGP.
- Local Plan Abstraction:
 - Agents commit to an activity on one level of detail without committing to the activity at a more detailed level.

Interleaving Plan, Coordination and Execution

- Communication:
 - Meta-Level Organization (MLO) indicates the information and control flows between the agents.
 - Who needs to know my plan?
 - Who has authority to override my plan?
- Partial Global Goal Identification:
 - Agents only know parts of the global goal.
 - Is an interpretation problem.

Interleaving Plan, Coordination and Execution

- Partial Global Plan Construction and Modification:
 - A goal of PGP is to identify related tasks and avoid redundant task achievement.
 - Integrating local plans that satisfy a partial global goal into a partial global plan.
 - Looks for the best concurrent actions for the partial global plan. – See algorithm Pg. 155 Algorithm 3.2

Interleaving Plan, Coordination and Execution

- Communication Planning:
 - Communicating the result of an action.
 - Plan for this communication.
 - May be synthesized results that are transformed into a complete result.
- Acting on Partial Global Plans:
 - Agents may first modify the abstract representation of the local plans if the partial global plan requires it.
 - The agent then executes the next local action based upon the modified local plan.

Interleaving Plan, Coordination and Execution

- Ongoing Modification
 - Changes in actions or events may cause the agent's abstract plan to be modified which indicates replanning is required.
 - When do you replan? What sensitivity level?
- Task Reallocation:
 - PGP can identify agents that are over and under committed through the exchange of abstract action models.
 - Negotiation.

Runtime Plan Coordination without Communication

- How can coordination be accomplished without communications?
 - Observation-based plan coordination.
 - Focal points.

References

- Pollack and Horty. *There's More to Life Than Making Plans: Plan Management in Dynamic, Multiagent Environments*. AI Magazine. Winter 1999 Vol. 20, No. 4