Multi-Agent Planning

Brad Clement
Artificial Intelligence Group
Jet Propulsion Laboratory
California Institute of Technology
brad.clement@jpl.nasa.gov
http://ai.jpl.nasa.gov/

Thanks to Edmund Durfee, U. Michigan for contributions

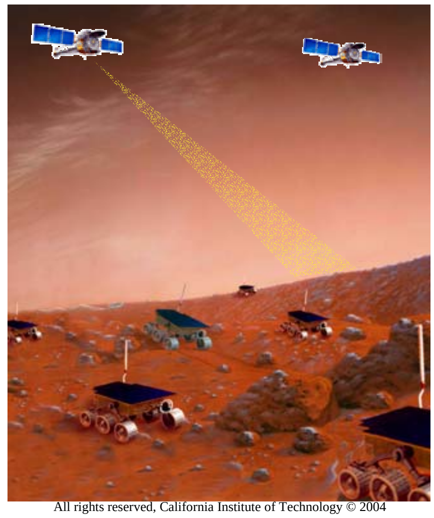
Outline

- What is multi-agent planning?
- Design Issues
- Applications
- Multi-agent planning problems and techniques

Why multiple agents?

(Dias & Stentz, 2003)

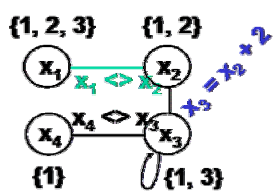
- A single agent cannot perform some tasks alone
- A *robot* team can accomplish a given task more quickly
- A robot team can make effective use of specialists
- A robot team can localize themselves more efficiently
- A team generally provides a more robust solution
- A team can produce a wider variety of solutions
- Decision-making too costly or sensitive to centralize
- Multi-agent system already exists

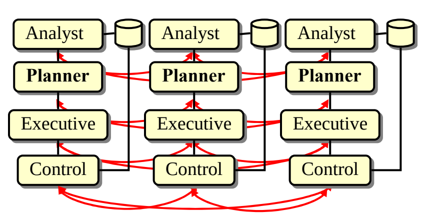


3

Role of Multi-Agent Planning

- Multi-agent problem solving
  - Contract nets
  - Auctions
  - Coalition formation
  - Distributed Constraint Satisfaction Problems (DCSP)
  - Distributed Constrained Heuristic Search (DCHS)
  - Multi-agent learning
  - **Multi-agent planning**
- Multi-agent system
  - Analysis
  - **Planning**
  - Execution
  - Control



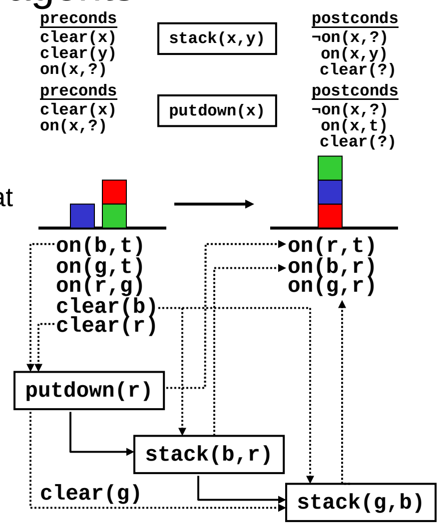


4

What is multi-agent planning?

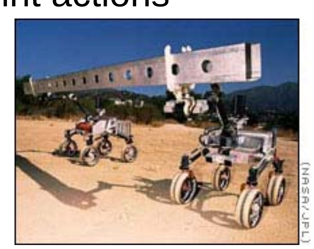
planning + agents

- Planning
  - near-term actions can effect subsequent ones in achieving longer-term goals
  - choose and order actions such that they lead from initial state to goals
- Multiple agents
  - Planning for multiple agents
  - Planning by multiple agents
  - Coordinating plans of multiple agents
  - Planning and coordinating
  - Distributed continual planning

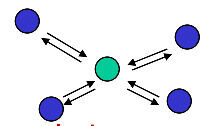


Why coordinate?

- Competing objectives (limited shared resources)
  - Shared parts and machines in factory
  - Battery power/energy
  - Market (goods, jobs)
- Shared objectives requiring joint actions
  - Carrying a beam
  - Joint sensing



Decentralized Decision-Making?



- Why centralize?
  - centralized computation often faster
  - centralized information can give better solutions
  - communicate only twice (gather problem info, issue results)
- Why decentralize?
  - competing objectives (self-interest)
  - control is already distributed
  - communication constraints/costs (b/w, delay, privacy)
  - computation constraints (parallel processing)

7

Criteria for Multi-Agent Planning

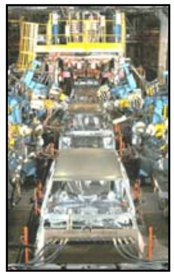
- computation costs
- communication costs
- plan quality
- flexibility (commitment)
- robustness
- scalability

8

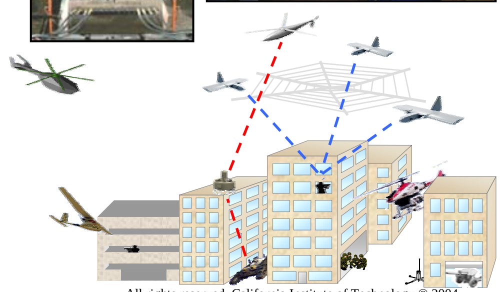
Applications 1

Why decentralize?

- competing objectives (self-interest)
- control is already distributed
- communication constraints/costs
- computation constraints



- Industry
  - car assembly
  - factory management
  - workforce management
- Military
  - distributed sensors
  - unmanned vehicles
  - troop/asset management

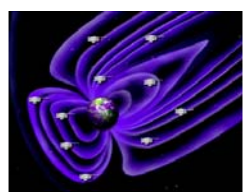


9

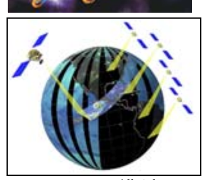
Applications 2

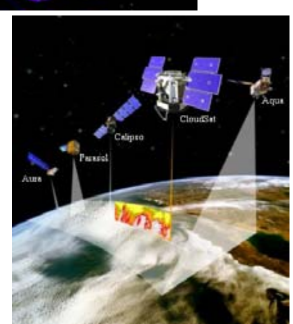
Why decentralize?

- competing objectives (self-interest)
- control is already distributed
- communication constraints/costs
- computation constraints



- Space
  - multiple rovers
  - spacecraft constellation
  - Earth orbiters
  - Mars network
  - DSN antenna allocation





10

Applications 3

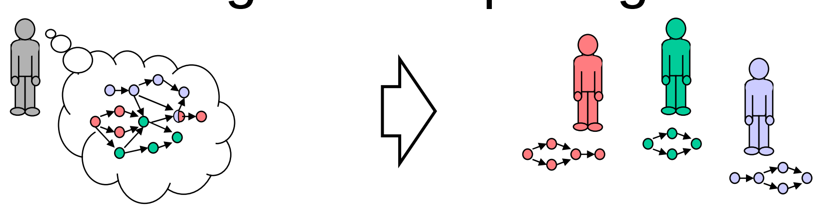
Why decentralize?

- **competing objectives (self-interest)**
- **control is already distributed**
- **communication constraints/costs**
- **computation constraints**

- Games
  - RTS (e.g. Starcraft, Age of Empires)
  - MMORPG (e.g. Ultima Online, Everquest, DAOC)
- Trading
  - supply chain management
  - B2B

11

Planning for Multiple Agents

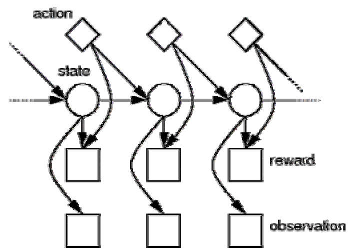


- Centralized planning, decentralized execution
- Planning requires
  - concurrent activity
  - temporal expressivity
- Many planners can be used for this
  - SHOP, MIPS, TLPlan, LPG, ASPEN, *etc.*

12

Markov Decision Processes (MDPs)

- POMDPs – partially observable MDPs
  - *S* – states
  - *A* – actions, transition probabilities from s_i to s_j for a_k
  - *O* – observations, probabilities of obtaining observation o_m when transitioning from s_i to s_j for action a_k
  - *V* – value function maps state history to a real number
- Extensions of MDPs for multiple agents
  - joint action
  - separate reward functions
  - observability by team
  - communication costs



	Individually Observable	Collectively Observable	Collectively Partially Observable	Non-observable
No Comm.	MMDP		Dec-POMDP POIPSG	
General Comm.		Xuan-Lesser	COM-MTDP	
Free Comm.				

P-complete

NP-complete

NEXP-complete

PSPACE-complete

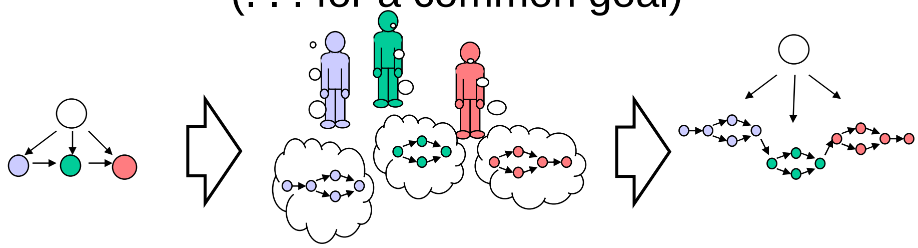
13

References – MDPs for Agents

- MDPs – Boutilier, JAIR, 1999
- MMDP – Boutilier, IJCAI '99
- Dec-POMDP – Bernstein *et al.*, UAI '00
- Xuan & Lesser, Agents '01, AAMAS '02
- COM-MTDP, Pynadath & Tambe, AAMAS '02, JAIR '02
- POIPSG, Kaebling *et al.*, UAI '01

14

Planning by Multiple Agents
(. . . for a common goal)

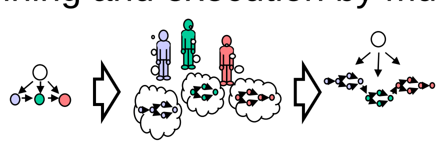


- Cooperative
- Does not necessarily require
  - concurrent activity
  - temporal expressivity
- Overlaps with parallel algorithms/processing

15

Distributed NOAH (Corkill, 1979)

- Planning and execution by multiple agents

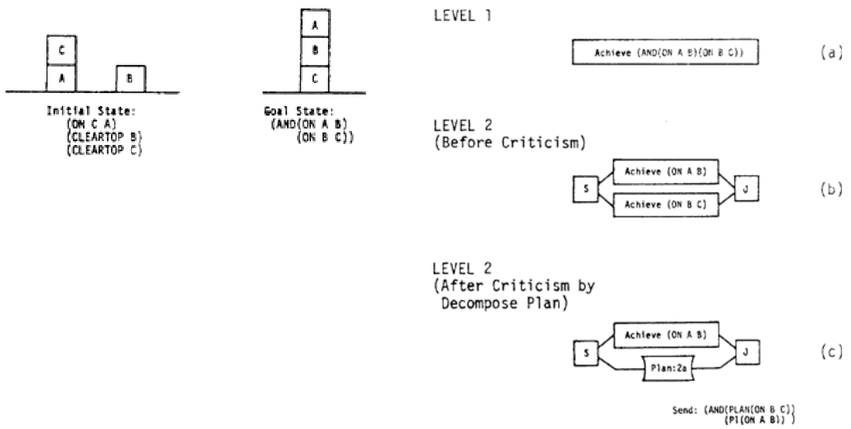


- Hierarchical planning
  - distribute conflict resolution (critic)
  - distribute world model
  - distribute resolution of deadlock
  - distribute elimination of redundant actions

16

Distributed NOAH

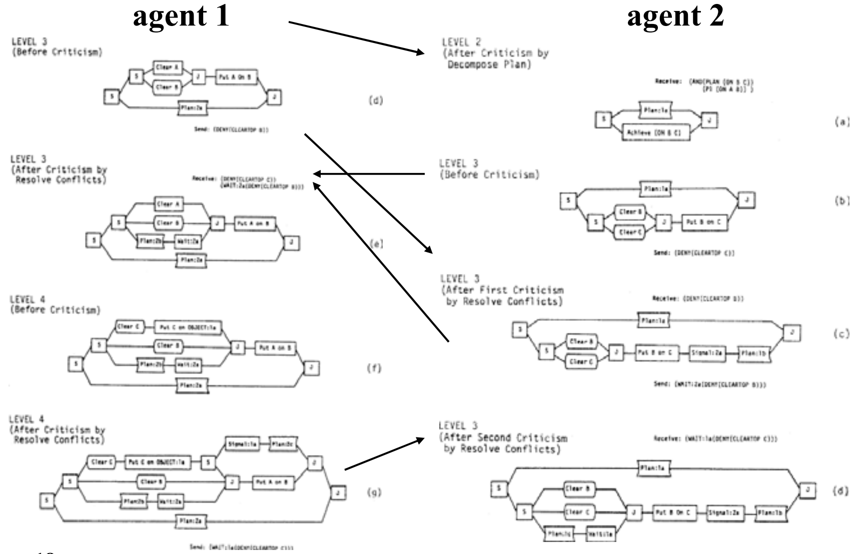
agent 1



17

Distributed NOAH

agent 1 **agent 2**



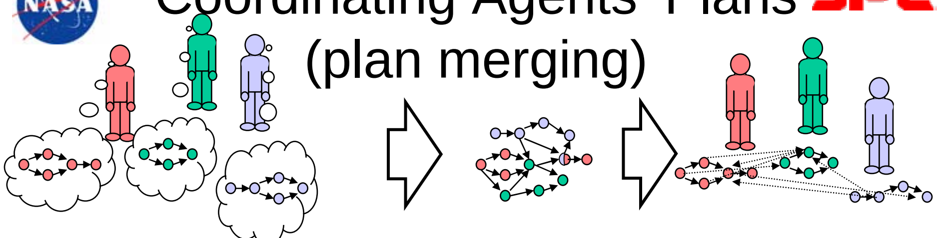
18

COLLAGE
(Lansky, 1991)

- Planning by multiple agents
- Distribute planning by partitioning into sub-problems
- Partially-ordered plan fragments with CSP-style binding constraints on action-parameter variables
- Action decomposition
- Planning as constraint satisfaction

19

Coordinating Agents' Plans (plan merging)



- Pre-existing separately developed plans
- Goal is to resolve conflicts over states and resources and avoid redundant action
- Solutions are commitments in the form of
  - temporal constraints (requiring wait, signal actions)
  - subplan choices (e.g. drive or take taxi)
  - choices of effects on resources/states (e.g. use machine A instead of B)
- Assumes execution by agents, so need
  - concurrent action
  - temporal expressivity
- Can be centralized by communicating plans
- Much work
  - plan merging (Georgeff '83, Ephrati & Rosenschein '94, Tsamardinos, *et al*. '00)
  - hierarchical plan merging (Clement & Durfee, '99, Cox & Durfee, '03)

20

Plan Merging

Given the candidate plans of the agents, consider all possible combinations of plans, executed in all possible orderings (interleavings or even simultaneous)

Generate all possible reachable sequences of states

For any illegal (inconsistent or otherwise failure) states, insert constraints on which actions are taken or when to ensure that the actual execution cannot fail

21

Plan Merging Algorithm-1

Each action has pre-conditions, post-conditions, and during-conditions (optional)

- Compare an agent's actions against each action of the other agents ($O(n^2a)$ comparisons) to detect contradictions between pre, post, and during conditions
- If none, pair of actions commute and can be carried out in any order.
- If some, determine if either can precede the other (post-conditions of one compatible with pre-conditions of other)
- All simultaneous or ordered executions not safe are deemed "unsafe"

22

Plan Merging Algorithm-2

Ignore actions that commute with all others

Complete safety analysis by propagation

- **Beginning actions a and b is unsafe if either consequent situation (adding post-conds of a to b, or b to a) leads to an unsafe ordering**
- **Beginning a and ending b is unsafe if ending a and ending b is unsafe**
- **Ending a and ending b is unsafe if both of the successor situations are unsafe**

Plan Merging Algorithm-3

In planning, assumption is that plan step interactions are exception

Therefore, dropping commuting actions leaves very few remaining actions

Examining possible orderings and inserting synchronization actions (messages or clock-times) therefore becomes tractable

Iterative Plan Formation

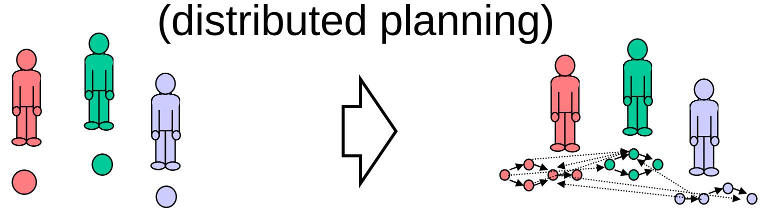
Sometimes, forming plans first and then coordinating them fails because of choices in initial plans formed

Instead, iterate between formation and coordination to keep alternatives alive

25

Planning and Coordinating

(distributed planning)



- Same as prior case (coordinating agents' plans), but planning has not completed up front
- Opportunity to resolve conflicts as plans are being refined
- Should compare to prior case where plans developed without communication and then coordinated
- Decentralized decision-making
  - communication costs can dominate

26

Plan Combination Search

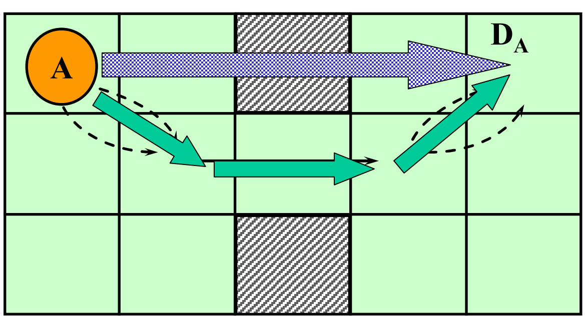
Given initial propositions about the world

1. Agents form successor states by proposing changes to current propositions caused by one action (or no-op)
2. Successor states are ranked using A* heuristic by all agents, and best choice is found and further expanded

Agents are simultaneously committing to a plan (corresponding to actions in solution path) and synchronizations (when actions are taken relative to each other)

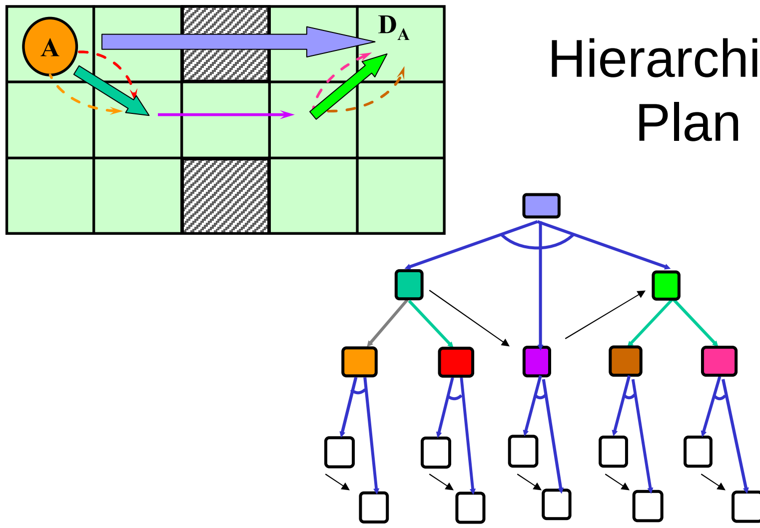
27

Hierarchical Example

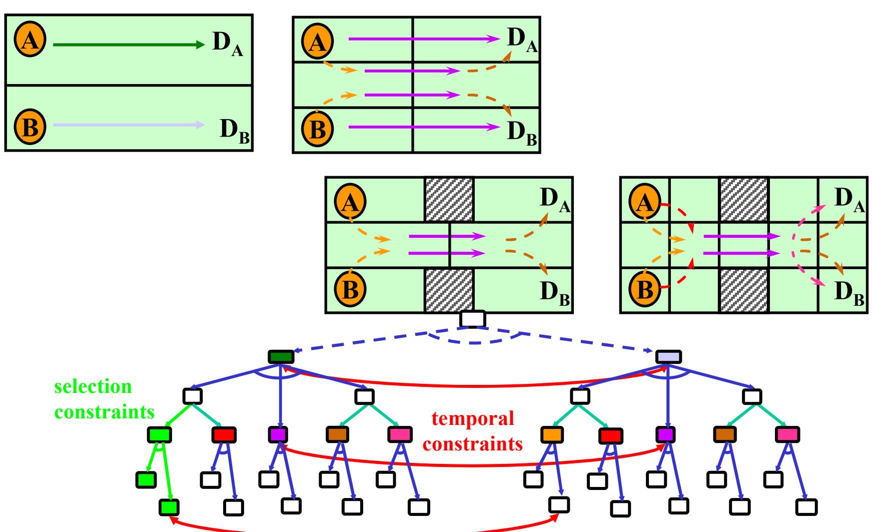


28

Hierarchical Plan



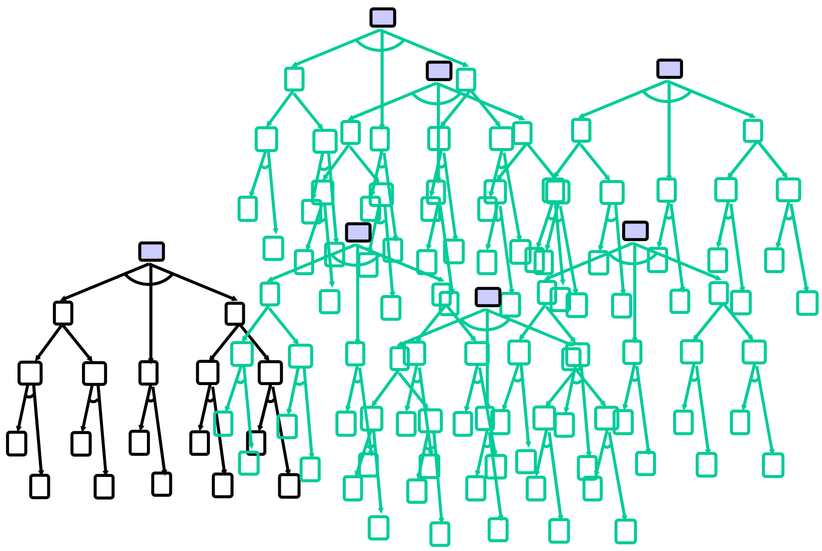
Multi-level Coordination & Planning (Clement & Durfee, 1999)



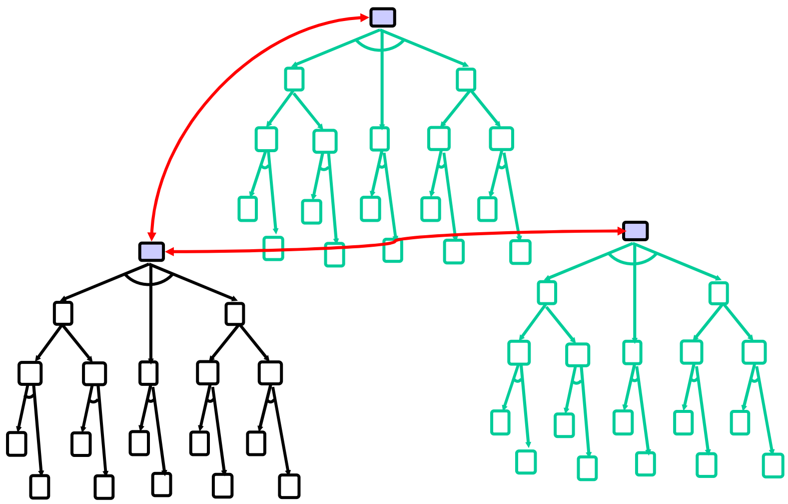
Hierarchical Coordination Search

1. **Initialize the current abstraction level to most abstract**
2. **Agents exchange descriptions of their plans and goals at the current level**
3. **Remove plans or plan steps with no potential conflicts. If nothing left, done. If conflicts should be resolved at this level, skip next step.**
4. **Set the current level to the next deeper level, and refine all remaining plans (steps). Goto 2.**
5. **Resolve by: (i) put agents in a total order; (ii) current top agent sends its plans to others; (iii) lower agents change plans to avoid conflicts with received plans; (iv) next lower agent becomes top agent**

Top-Down Coordination

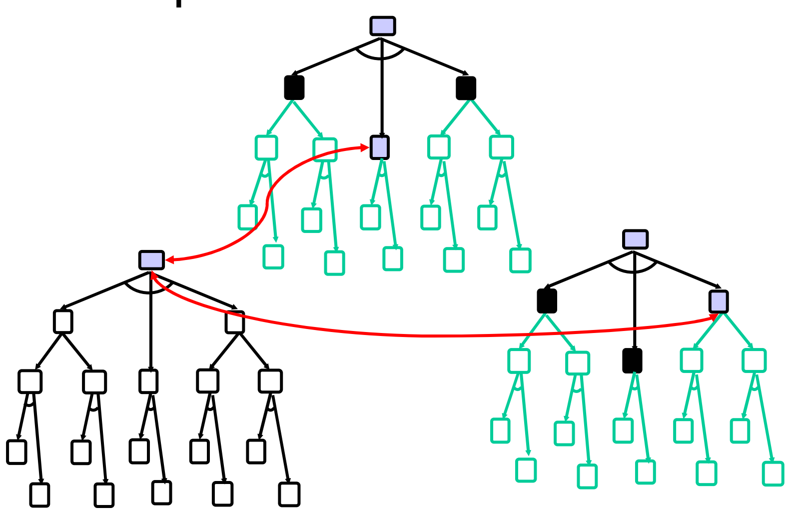


Top-Down Coordination



33

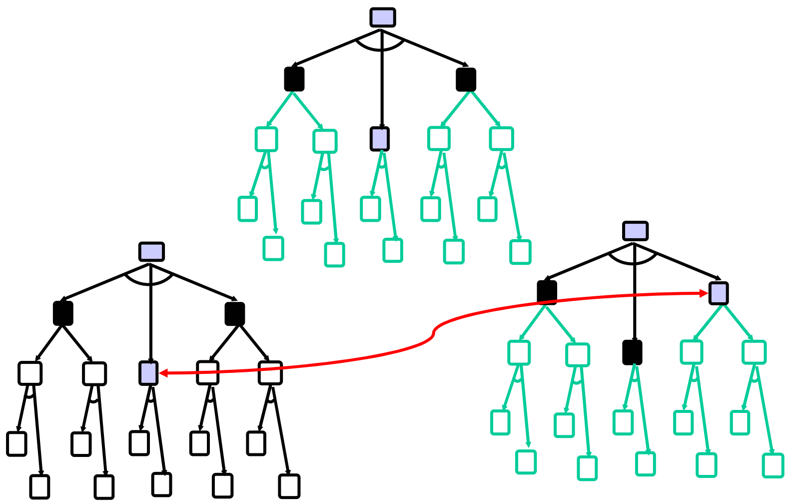
Top-Down Coordination



34

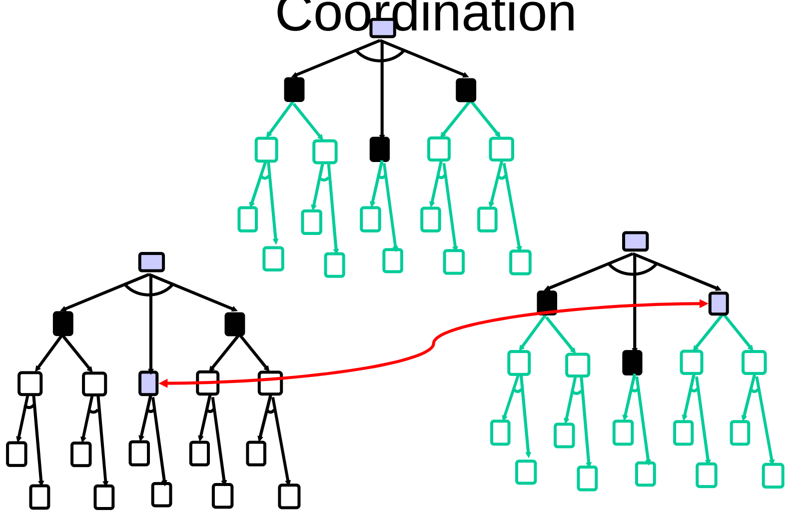
All rights reserved, California Institute of Technology © 2004

Top-Down Coordination



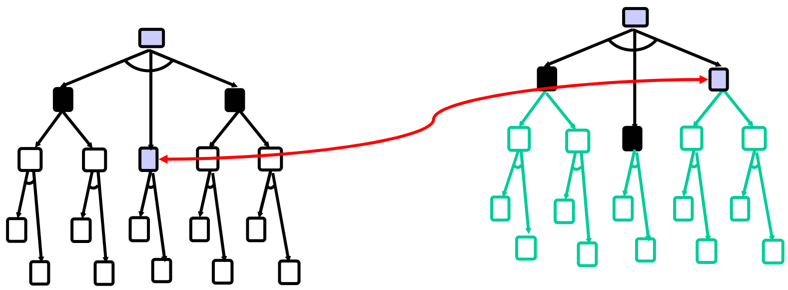
35

Top-Down Coordination



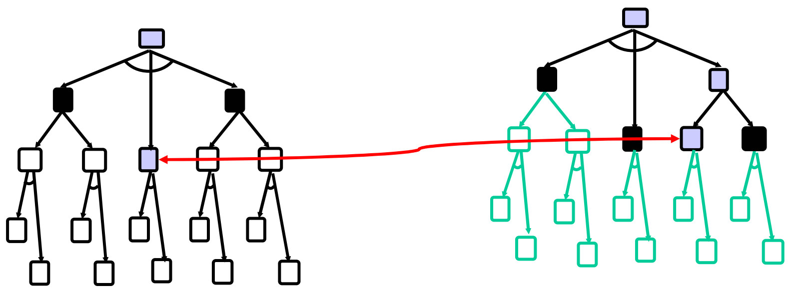
36

Top-Down Coordination



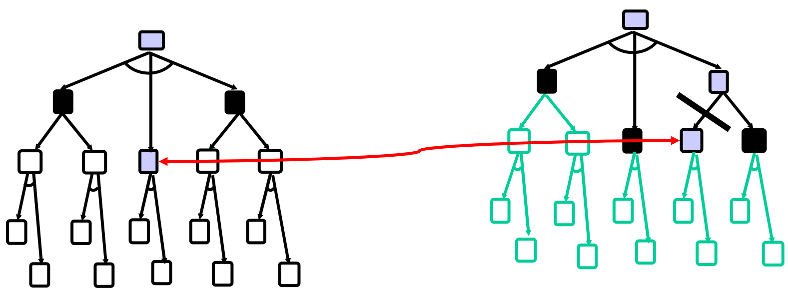
37

Top-Down Coordination



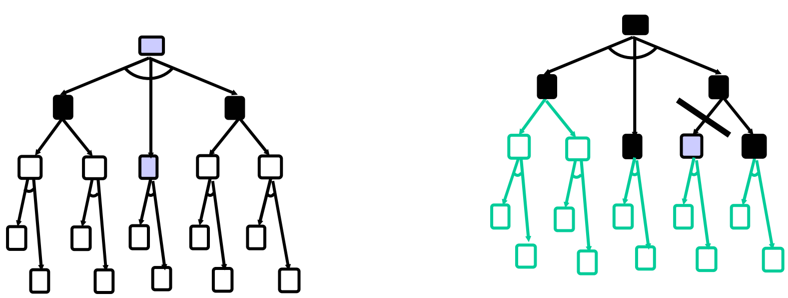
38

Top-Down Coordination



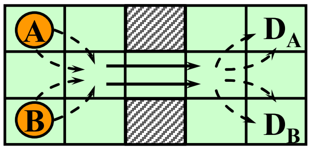
39

Top-Down Coordination



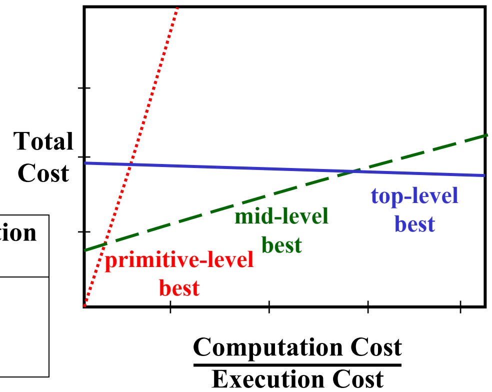
40

Coordinating at Abstract Levels Can Improve Performance



BFS algorithm

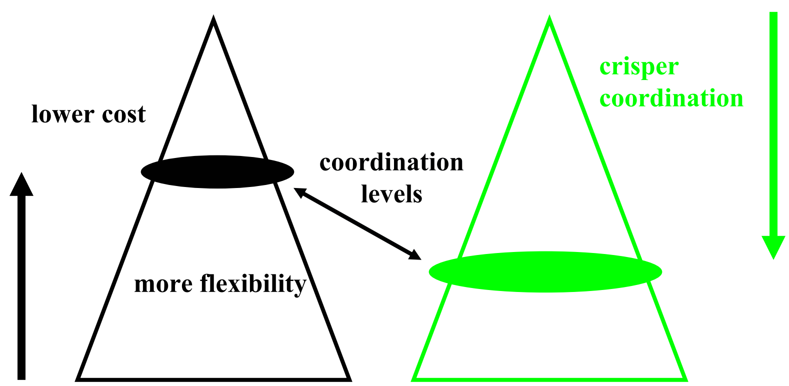
level	computation time	execution time
top	4	60
mid	159	40
primitive	2375	35



41

Tradeoffs

Choice of level at which coordination commitments are made matters!



42

Generalized Partial Global Planning

(GPGP, Decker & Lesser, 1995)

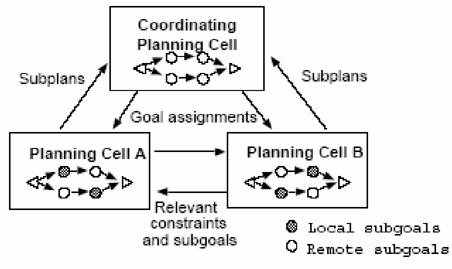
- Mechanisms to generalize PGP
  - updating non-local viewpoints
  - communicating results
  - handling redundancy of effort
  - resolve conflicts (hard constraints)
  - handle soft constraints ("optimize")
- Examines tradeoffs of using mechanisms according to
  - communication overhead
  - execution time
  - plan quality
  - missed deadlines

43

DSIPE

(desJardins & Wolverton, 1999)

- Distributed version of SIPE-2 planning system
- SIPE – mixed-initiative hierarchical (HTN) planning
- Centralized conflict resolution
- Creates common partial views of subplan
- Synchronization and plan-merging
- Irrelevance reasoning on pre-conditions and effects to limit communication



44

Shared Plans
(Grosz & Kraus, 1996)

- Model and theory of collaborative planning

$\text{Int.To}(G, \alpha, T_i, T_\alpha, C_\alpha)$

(1) $[\text{basic.level}(\alpha) \wedge \text{Bel}(G, \text{Exec}(G, \alpha, T_\alpha, \text{constr}(C_\alpha)), T_i) \wedge \text{Commit}(G, \alpha, T_\alpha, T_i, C_\alpha)] \otimes$

(2) $[\neg\text{basic.level}(\alpha) \wedge$
(2a) $[(\exists P, R_\alpha)$
(2a1) $\text{FIP}(P, G, \alpha, T_i, T_\alpha, R_\alpha, C_\alpha)] \otimes$
(2b) $[(\exists P, P_{elab}, T_{elab}, R_{elab})$
(2b1) $[\text{PIP}(P, G, \alpha, T_i, T_\alpha, C_\alpha) \wedge$
(2b2) $\text{FIP}(P_{elab}, G, \text{Elaborate\_Individual}(P, G, \alpha, T_i, T_\alpha, C_\alpha), T_i, T_{elab}, R_{elab}, C_{elab/\alpha}) \wedge$
(2b3) $\text{Int.To}(G, \text{Elaborate\_Individual}(P, G, \alpha, T_i, T_\alpha, C_\alpha), T_i, T_{elab}, C_{elab/\alpha})]]]$

Int.To – intend-to
FIP – full individual plan
PIP – partial individual plan

basic.level – leaf of recipe tree
Bel – believe
R – recipe

G – agent
α – action
T – time
C – constraints

45

Shared Plans

Type	Notation	Meaning
Modal Operators	Int.To	intend-to
	Int.Th	intend-that
	Pot.Int.To	potential intention-to
	Pot.Int.Th	potential intention-that
	Exec	ability to perform basic level actions
	Commit	commitment to basic level actions
	Do	performance of action
Meta-Predicates (Plans)	FIP	full individual plan
	PIP	partial individual plans
	SP	SharedPlans
	FSP	full SharedPlans
	PSP	partial SharedPlans
Meta-Predicates (Ability)	CBA	can bring about
	CBAG	can bring about group

46

Shared Plans

Meta-Predicates (subsidiary)	CONF	actions/propositions conflict
	GTD	get to do
	CC	can contract
	CCG	group of agents can contract
	BCBA	believe can bring about
	MBCBAG	mutually believe can bring about group
	WBCBA	weakly believe can bring about
	WMBCBAG	weakly mutually believe can bring about group
	MP	member of group performs action
	SGP	sub-group performs action
	FSPC	contracting in FSP
	PSPC	contracting in PSP
Act-types for Planning Actions	Select_Rec	agent selects (extends) recipe
	Select_Rec_GR	group of agents selects (extends) recipe
	Elaborate_Individual	agent extends partial plan
	Elaborate_Group	group of agents extends partial SharedPlan
Predicates (subsidiary)	single.agent	single-agent action
	multi.agent	group action
	basic_level	basic-level action
Functions	constr	constraints of a context
	recipe	recipes for action
	cost	cost of action
	econ	relativize cost (for benefit comparison)

47

Distributed Planning and Execution

Issues in when agents plan and coordinate, relative to each other, and relative to execution

Are often sequentialized

No sequential order works well in all cases

48

Post-Planning Coordination

Essentially, plan merging techniques

Dealing with execution problems can involve:

- Contingency preplanning: detecting multiagent contingency, and invoking already coordinated response
- Monitoring/replanning: detecting deviation and restarting the planning/coordination process

Obviously, localizing impacts minimizes fresh coordination; building a plan that permits localized adjustments can be important, but might be less efficient

49

Pre-Planning Coordination

Impose coordination constraints before planning is done; plans work within these

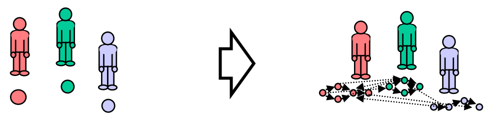
Example: Set the boundaries; define the roles

Social laws: Define what could be done and when, then leave it up to agents to plan within the legal limits

Cooperative state changing rules: Force agents planning decisions into cooperative behaviors

50

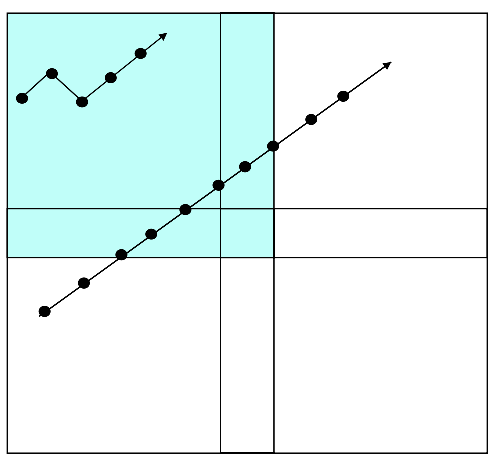
Distributed Continual Planning



- Same as prior case (distributed planning), but
  - plans are being executed at same time
  - goals may change
- At any given time, plans might only be partially coordinated, and execution results could cause chain reactions of further planning and coordination
- May break and re-make commitments
  - unexpected event/failure
  - goal change
- Must reach consensus (and deconflict) on plan segments before they are executed
  - real time guarantees?
  - what if not possible?
- In a sense, the coordinated plans are only evident after the fact, as they are continually being adjusted during execution

51

Example Application: Distributed Vehicle Monitoring



52

Partial Global Planning
(Durfee & Lesser, 1991)

1. **Task allocation: inherent**
2. **Local plan formulation: sequence of interpretation problem solving activities**
3. **Local plan abstraction: major plan steps (such as for time-region processing)**
4. **Communication: Use meta-level organization to know who is responsible for what aspects of plan coordination**

53

Partial Global Planning (cont)

5. **Partial global plan construction: Pieces of related plans (e.g., potentially tracking the same vehicle) are aggregated**
6. **Partial global plan modification: redundant or inefficient schedules are adjusted to improve collaborative performance**
7. **Communication planning: identification of partial results that should be gainfully exchanged, and when**

54

Partial Global Planning (cont)

8. **Mapping back to local plans: Partial global plan commitments are internalized**
9. **Local plan execution**

Cycle repeats as local plans change or new plans from other agents arrive. Always acting on local information means that there could be inconsistencies in global view, but these are tolerated

55

Shared Activity Coordination

(SHAC, Clement & Barrett, 2003)

– distributed continual planning algorithm

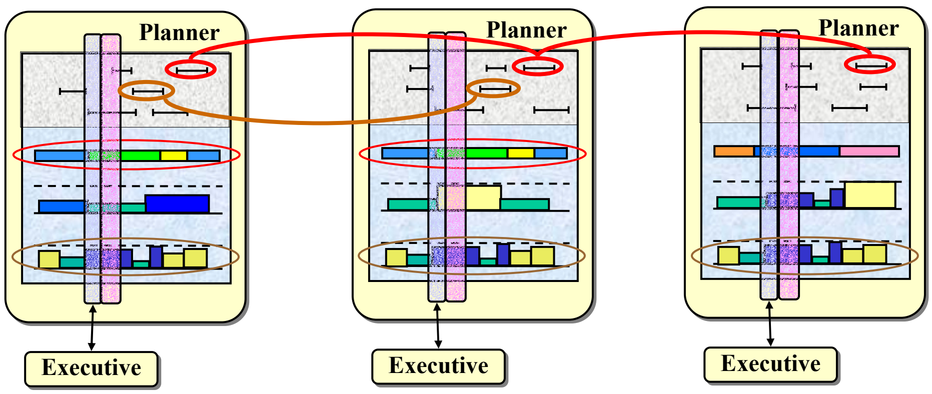
– framework for defining and implementing automated interactions between planning agents (a.k.a. coordination protocols/algorithms)

– software

- planner-independent interface
- protocol class hierarchy
- testbed for evaluating protocols

56

Shared Activity Coordination



Shared activities implement team plans, joint actions, and shared states/resources

Shared Activity Model

- **parameters** (string, integer, *etc.*)
  - **constraints** (e.g. agent4 allows start_time [0,20], [40,50])
- **decompositions** (shared subplans)
- **permissions** - to modify parameters, move, add, delete, choose decomposition, constrain
- **roles** - maps each agent to a local activity
- **protocols** - defined for each role
  - change constraints
  - change permissions
  - change roles
    - includes adding/removing agents assigned to activity

SHAC Algorithm

Given: a ***plan*** with multiple activities, including a set of ***shared_activities***, and a ***projection*** of ***plan*** into the future.

1. Revise ***projection*** using the currently perceived state and any newly added goal activities.
2. Alter ***plan*** and ***projection*** while honoring ***constraints*** and ***permissions of shared_activities***.
3. Release relevant near-term activities of ***plan*** to the real-time execution system.
4. For each shared activity in ***shared_activities***
   – apply each associated *protocol* to modify the activity
5. Communicate changes in ***shared_activities***.
6. Update ***shared_activities*** based on received communications.
7. Go to 1.

59

Market Mechanisms

- Used for resource/task allocation
- Plans share resources and tasks over time (another resource)
- Combinatorial auctions for bids over multiple resources
  - optimization techniques capture constraints and produce schedules
  - if during execution, auction/optimization may need to be repeated for unexpected events
  - difficult to motivate truthful bids and obtain optimal allocations, but no other technique gives such guarantees

60

Summary

- Multi-agent systems have many benefits (especially for robotics)
- Often hard to motivate decentralized decision-making unless agents are naturally self-interested
- Many applications, but appropriate architecture is not obvious
- Multi-agent planning problems and techniques
  - Planning for multiple agents (done?)
  - Planning by multiple agents (hard to motivate?)
  - Coordinating plans of multiple agents (many techniques)
  - Planning and coordinating
  - Distributed continual planning

  } communication costs are important
- Other new directions
  - flexibility and robustness
  - multi-agent uncertainty
  - real-time coordination
  - negotiation (self-interested agents)

61

References

- Clement, B. and Barrett, A. Continual Coordination through Shared Activities. *Proceedings of the Second International Conference on Autonomous Agents and Multi-Agent Systems*, 2003.
- Clement, B. and Durfee, E. Theory for Coordinating Concurrent Hierarchical Planning Agents Using Summary Information. *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pp. 495-502, 1999.
- Corkill, D. Hierarchical Planning in a Distributed Environment. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pp. 168-175, Tokyo, August 1979.
- Decker K. and Lesser, V. Designing a Family of Coordination Algorithms. In *Proceedings of the First International Conference on Multi-Agent Systems*, San Francisco, July 1995.
- desJardins, M. and Wolverton, M. Coordinating a distributed planning system. *AI Magazine*, 20(4):45–53, 1999.
- Durfee, E. and Lesser, V. Partial Global Planning: A Coordination Framework for Distributed Hypothesis Formation. *IEEE Transactions on Systems, Man, and Cybernetics, Special Issue on Distributed Sensor Networks*, SMC-21(5):1167-1183, September 1991.
- Grosz, B. and Kraus, S. Collaborative Plans for Complex Group Action. In *Artificial Intelligence*. 86(2), pp. 269-357, 1996.
- Lansky, A. Localized Search for Multi-agent Domains, *Proc. 12th Int'l Joint Conf. Artificial Intelligence (IJCAI '91)*, pp. 252-258, 1991.

62