Improving the Robustness of Team Collaboration through Analysis of Qualitative Interactions

Matthew Klenk and **Daniel G. Bobrow**
Johan de Kleer
Palo Alto Research Center
Palo Alto, CA
{klenk,bobrow,dekleer}@parc.com

Wendy Mungovan and **Jorge Tierno**
Barnstorm Research
Cambridge, MA
{first.last}@barnstormresearch.com

Abstract

Members of effective teams must have knowledge about each others future actions. Typically, this is done through messages or precomputed divisions of labor. The first requires ongoing communication between the agents and the latter constrains the autonomy of the individual agents. We introduce *coordination rules* that facilitate collaboration between autonomous agents when communication is lost. By envisioning the results of all possible plan executions for each agent, we identify which decisions result in the greatest increase of within-team uncertainty. If removing this action does not significantly reduce the expected utility of the plan, we create a coordination rule, a statement that the agent will or will not take a particular action in some possible future. Coordination rules facilitate collaboration by improving state estimation and prediction by teammates. To accomplish this, we make the following contributions. First, we identify qualitative interactions by representing the space of decisions made by the agents in *plan-with-options* and their consequences in a *factored envisionment* that compactly represents multi-agent simulations. Second, we define two classes of within-team uncertainty metrics and how they are computed over the envisionment. Third, we present an evaluation of the effects of coordination rules on action selection in three scenarios. In all three scenarios, coordination rules enabled extended planning horizons and reduced planning times with no significant effect on plan quality.

1 Introduction

Coordinated goal-oriented behavior is a hallmark of intelligence. Groups that need to coordinate actions include trained teams (e.g., athletes executing a play), familiar novices (e.g., a family arranging childcare and meals), or complete strangers (e.g., drivers navigating an intersection). Deciding what to do is difficult in these scenarios due to ambiguity introduced by other agents' actions. Coordination can be achieved by constraining autonomy (e.g., athletes executing a play), division of labor (e.g., dad cooks dinner while mom picks up the kids), and social norms (e.g., at a four-way stop, the car on the right gets to go first). For improved performance, coordination is achieved by considering other agents' goals and reasoning about possible futures. This preserves the ability of the agents to respond autonomously to failures and opportunities that arise during plan execution while still coordinating their behavior, even in absence of communication. We define *coordination rules* as restrictions to the space of actions available to individual agents during execution. We demonstrate that when communication is not possible coordination rules facilitate collaboration by reducing within-team uncertainty.

To enable coordinated behavior, it is necessary to take into account the uncertainty of the dynamics of the world as well as the decisions of other agents. We use a standard planning model to represent the dynamics of the world and introduce *plan-with-options* as a model of contingent plans to represent the space of decisions left to the agent. *Envisioning* from qualitative reasoning [Kuipers, 1994] [Weld and de Kleer, 1989] is a multi-trajectory simulation process that given a model analyzes all qualitatively distinct futures for a scenario. In the multi-agent setting, many of the distinctions captured by traditional approaches are not relevant. Therefore, we introduce *interaction-based factoring* that creates a compact representation of possible futures. To automatically construct coordination rules, we define two classes of within-team uncertainty metrics and how they are computed over the envisionment. Finally, we evaluate these ideas over a set of scenarios.

The contributions of this paper include:

- Definition of coordination rules as constraints on an agents' actions to facilitate collaboration
- Exposition of interaction-based factoring algorithm that produces a compact representation of a multi-agent envisionment
- Description and discussion of two classes of within-team uncertainty metrics
- Evaluation of the effects of coordination rules on the action selection process in three scenarios

This is a cognitive systems [Langley, 2012] problem because it focuses on agents representing and reasoning about other agents' behaviors within a planning and execution system. Furthermore, our contributions combine representations and algorithms from artificial intelligence (AI) planning and qualitative reasoning. Finally, we show how an automated analysis of qualitative interactions can facilitate collaborative behavior.

2 Representing Plans and Futures

Our scenarios involve agents with different capabilities, team-level evaluation, and uncertainty in our own actions, teammates' decisions, and adversary intentions. The Planning Domain Definition Language (PDDL) provides a standard mechanism for describing the dynamics of the world and scenarios [McDermott *et al.*, 1998]. We use PDDL to represent the preconditions and effects of actions. Uncertain effects are modeled using multiple versions of the same action along with associated probabilities.

An *envisionment* [Weld and de Kleer, 1989] is a graph of qualitative states with edges indicating the possible successors. Edges are labeled by the action or set of actions, in the multi-agent case, that change the state. In this context, it is analogous to the state-action space of planning [Ghallab *et al.*, 2004].

Consider a scenario with two rovers, a survey rover and an extraction rover, that are collaborating to extract minerals. The world has a base location and four possible mining locations. The survey rover can move between any of the locations, use its sensors to identify if minerals are present, and use its sensors to guide the extraction rover's drilling action. The full envisionment has 200 states.

In the worst case, the number of states in the envisionment is exponential in the size of the set of ground predicates, or facts, in the scenario. Therefore, we introduce two concepts that facilitate representing large scenarios.

2.1 Plan with Options

One cause for the large state space is that afforded action will result in a possible state transition. The AI planning community reduces the set of decisions required to reason about large state spaces by using domain knowledge in the form of hierarchical task networks (HTNs) [Erol *et al.*, 1994]. We leverage the same idea here for reasoning about other agents' behavior. HTNs represent tasks hierarchically and HTN planning involves decomposing tasks into subtasks and finally primitive actions that can be executed by the agent. An HTN *method* specifies the decomposition of a task. If multiple methods apply to the same task, then there are multiple plans that satisfy the task. We define the *plan-with-options* to be all possible decompositions for a goal task, as well as the current sequence of actions each agent is pursuing. Therefore, instead of reasoning over every possible action available to every agent, we consider the space of decomposition decisions faced by the other agents. This provides the agent autonomy to respond to changing conditions while remaining committed to the collaborative activity.

Consider the rover example, in which plan-with-options can be used to constrain the search pattern of the agents. For example, the survey rover may only visit the locations in a particular sequence with the option of returning home after visiting each one, and the drilling rover must follow the same pattern and only drill when co-located with the survey rover. Instead of having to generate successor states from all pairs of rover locations for the extraction rover's drilling action, we only generate successor states for the drilling action from states in which the rovers are co-located.

To implement plan-with-options, we define HTNs for each agent. We model options with multiple decompositions for same task. We translate the HTN into PDDL [Alford *et al.*, 2009] and SAPA, a domain-independent planner, [Do and Kambhampati, 2003] to generate all possible sequences of states.

2.2 Interaction-based Factoring

The other main driver in the complexity of the envisionment is the explicit representation of temporal differences between agents' actions. If the survey rover is headed toward location 1 and the extraction rover is headed toward location 2, standard representations would include the following states:

1. Both rovers en route to their respective locations
2. Survey rover at location 1 and extraction rover en route to location 2
3. Survey rover sensing at location 1 and extraction rover en route to location 2
4. Survey rover en route to location 1 and extraction rover at location 2
5. Rover 1 at location 1 and extraction rover at location 2
6. Rover 1 sensing at location 1 and extraction rover at location 2

It does not matter which arrives first, unless they are interacting with each other from their respective locations. Therefore it should be possible to represent the survey rover's actions with three states (en route, at location 1, and sensing at location 1) and the extraction rover's action as two states (en route and at location 2). While this appears to only remove a single state, consider what happens when we allow the drilling rover to drill when it reaches its destination. In the baseline case, there will be three additional states (one for 4, 5, and 6), but, when considered separately, there would only be a single additional state.

Factored envisioning separates the envisionment into non-interacting components [Clancy and Kuipers, 1997]. Originally applied through causal analysis to qualitative constraints, the idea was extended to battlespace planning by analyzing the spatio-temporal trajectories of each agent [Hinrichs *et al.*, 2011]. We extend this idea with *interaction-based factoring* in which decisions about when agents are reasoned about individually versus as a group are made using the planning domain model. The core idea is that if an action includes multiple agents in its arguments, then those agents are *interacting* in that action. Factored envisionments consist of *factors*, which are envisionments of a subset of the agents and *factored states* that exclude facts related to other agents.

Algorithm

Algorithm 1 outlines our interaction-based factoring approach. It works with two stages, both of which operate recursively. In the first stage, we identify subsets of agents that are able to take actions together in the current state. That is, there exists an action for which the preconditions are met and each of the agents is a participant. For each subset, including the original set, we create a factored state, which includes the facts related to those agents and objects in the world. From this factored state, we create an envisionment using all the actions that include all of the actions in which the subset of agents are interacting. This envisionment is a factor. For each state in each factor, if further subsets of agents from that factor are afforded actions, we create additional factors from each of those factored states. In the second stage, we identify sets of factored states from across the factors of the previous stage called *join state sets*. Join state sets include a single factored state from each factor being joined. Each join state set results in a new factored state if there are new actions that can be taken by the set of agents in the joint state set. This recursive factoring and joining approach continues until there are no further actions that result in new factors to be created.

Algorithm 1 Factored envisioning algorithm

```
procedure fenvision($S_{init}, D_{full}, Agents$)
    $FS \leftarrow \emptyset$
    for all $D_i, A_i \in D, A$ do
        $FS_i$ = factor_state($A_i, S_{init}$)
        if envision($FS_i, D_i$) then
            add_child($S_{init}, FS_i$)
            $FS$.add($FS_i$)
            if factorable($A_i$) then
                for all $js \in$ all_successors($FS_i$) do
                    fenvision($js, D_{full}, A_i$)
                end for
            end if
        end if
    end for
    for all $D_i, A_i \in D, A$ do
        for all $jg \in$ possible_join_graphs($FS, A_i$) do
            for all $jss \in$ identify_join_state_sets($JG_i, D_i$) do
                $js \in$ create_joint_state($jss$)
                envision($js, D_i$)
                for all $s \in$ all_successors($js$) do
                fenvision($js, D_{full}, A_i$)
                end for
            end for
        end for
    end for
```

Consider our rover example again, the envisionment includes three types of factors: one for each rover and one for the rovers together. A single trajectory will involve spliting from the initial state as both rovers move to different locations, joining when they collaborate to drill the minerals, and separating again as they travel back to home. The complexity due to the locations of the rovers alone goes from 25 states (one for every combination of rover locations) to 15 states in the factored envisionment (five locations for the survey rover factor, five locations for the extraction rover factor, and five states for each co-location). In the full envisionment, the state changes for sensing and drilling actions are multiplied by 25, but in factored envisionment, the sensing actions are in the survey rover factor and the drilling actions occur in the ex- traction rover factor. This results in an exponential reduction in qualitative states in the factored envisionment.

Factoring Evaluation

As shown in the example, there is a significant savings when moving to the factored envisionment. Theoretically, if the state-action space of two individual agents are M and N, then their combined envisionment would include $M \times N$ states. If there was no interaction, their factored envisionment would include $M + N$ states. If they were interacting in every state, then the factored envisionment would still be on the order of $M \times N$ states.

Most collaborative problems have a mixture of interaction and independence. To evaluate the impacts of factoring on the size of the state graph, we create a series of scenarios with increasing complexity (i.e., the number of agents and actions available). These scenarios involved a set of aircraft striking a set of defended targets. Scenarios were made more complex by adding aircraft, locations, and targets.

Figure 1: The effects of factoring on the size of the envisionment on air strike scenarios.

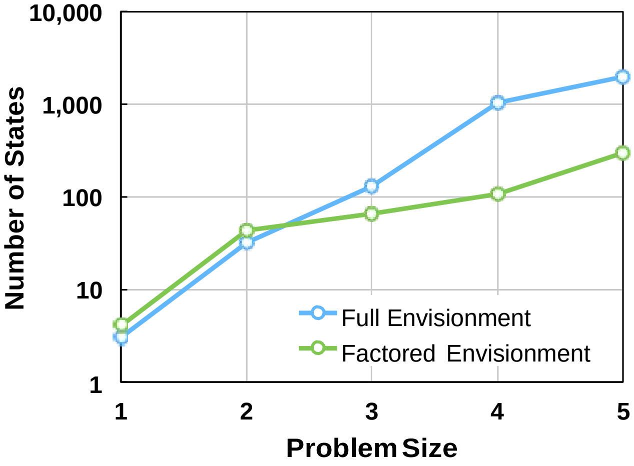


Figure 1 plots the problem complexity against the number of states. In the smaller scenarios, all of the agents are constantly interacting. In the first two problems, the factored envisionment is larger. This is due to the overhead of creating initial states for each factor. As the problems get larger, there is an apparent order of magnitude savings resulting from reasoning about non-interacting agents independently.

3 Generating Coordination Rules

Coordination rules reduce within-team uncertainty by restricting the space of possible actions for a particular agent. By analyzing the envisionment, the agent measures the amount of uncertainty introduced by each of its decisions. Next, it generates a coordination rule that eliminates the action with the greatest increase in within-team uncertainty in

Scenario	State	Action
SEAD	✓	✗
Patrol	✗	✓
Strike	✓	✓

Table 1: Results for the selection of the appropriate rule by each metric in each scenario.

which the remaining decisions still support a successful collaboration. When action probabilities are known we can evaluate the tradeoff between within-team uncertainty and overall expected utility.

To generate effective coordination rules, it is necessary to define within-team uncertainty. We identify two broad classes of uncertainty:

- State-based: These metrics measure the similarities between possible future states of the world
- Action-based: These metrics measure the similarities between possible action sequences for each agent.

These classes specify what is being compared in the metric (actions or states). It is necessary to determine the time period of interest. For example, when deciding which items to pickup in a grocery store, it is helpful to know what part of the store your partner is in for the next few minutes as it is likely you will run into each other soon. On the other hand, if you are working on a document without communication, you care about which sections of the document your colleague will have changed at the next possible communication point.

We used the following three scenarios with target coordination rules to evaluate these two metrics. For the state-based metric, we use the terminal states of the envisionment, and, for the action-based metric, we use the complete action trajectories.

- SEAD: Two aircraft must search two regions and destroy enemy air defenses. In addition to moving actions, there are sensing and attacking actions for both friendly and enemy units.
  - Rule: Aircraft 1 will visit target region 1 before target region 2.
- Patrol: Two aircraft with different sensors must identify and track targets within an area of interest. Collaboration is required for the search aircraft to hand off targets to the tracking aircraft.
  - Rule: Search aircraft will only follow targets within one of the regions on the return trip.
- Missile Strike: Two salvos of missiles are launched at high-priority and low-priority targets. The targets have countermeasures that may engage the salvos, and the salvos can change what they are targeting.
  - Rule: If each salvo was alive at the point of communication loss, the salvo that was targeting the high-priority target will engage the low-priority target.

Table 1 indicates if each metric was able to identify the expected coordination rule. State-based uncertainty metrics are most effective when there are clear aspects of the state that are relevant to mission decisions (e.g., the status of a particular target). Action-based metrics work best in scenarios with repeated patterns of activity (e.g., moving between different regions in a patrol).

4 Impacts on Action Selection

We claim that coordination rules support autonomous behavior by facilitating action selection by teammates that cannot directly communicate with one another. To evaluate this, we use decentralized partially observable Markov decision processes (Dec-POMDP) [Bernstein *et al.*, 2000] to formalize the collaborative action selection problem. Coordination rules result in a smaller state-action space, and therefore we expect off-the-shelf solvers to create plans faster and with longer horizons after applying coordination rules. Given that coordination rules remove options, they may reduce the value of the optimal solution. On the other hand, they may also improve the value of approximate solutions by simplifying the action-selection process.

4.1 Decentralized Partially Observable Markov Decision Process

A Dec-POMDP is defined by a set of agents, a set of states (including an initial state distribution), a set of actions for each agent, a transition function that specifies the probability of each state given a set of actions taken by the agents, a reward function that specifies the reward for being in a state, the set of observations for each agent, and an observation model that specifies the probability of each observation for each agent given a state. A solution is a set of policies, one per agent, which maps each sequence of observations to an action for each agent.

4.2 Method

We used the Multi Agent Decision Problem Toolbox [Spaan and Oliehoek, 2008] (http://fransoliehoek.net/madp) to evaluate the effects of coordination rules. This is an open source toolbox implements a variety of Dec-POMDP solvers and benchmark problems. We selected three scenarios with the following coordination rules):

- Intersection: Two vehicles at an intersection with actions to move forward or wait. If both cars move forward at the same time, they crash resulting in a negative reward, and if both cars make it through, there is a positive reward.
  - Rule: Car 1 will wait.
- Recycling robots: Two robots either recycle a small amount, a large amount or recharge. While the robots can recycle small pieces independently, they must collaborate to recycle a large piece, which results in the largest reward.
  - Rule: Robot 1 will not attempt to recycle large amounts.
- Missile strike: Same as in Section 3

For each of these scenarios, we generated policies for four planning horizons and used two solution methods:

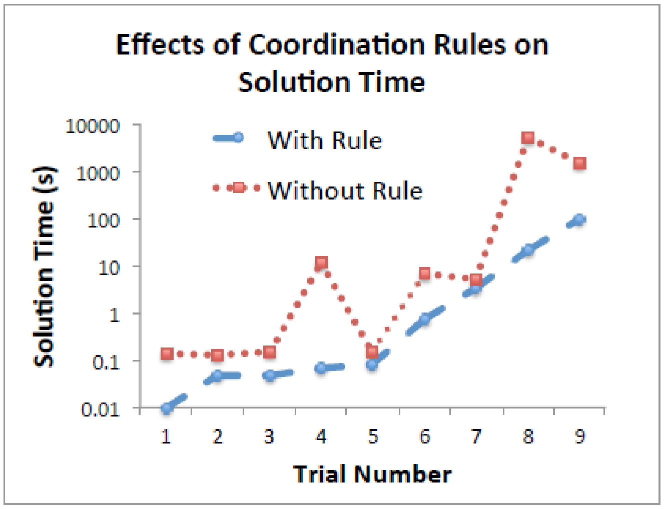
- Optimal: General Multi-Agent A-Star [Oliehoek *et al.*, 2008]
- Approximate: Forward Policy Search with Alternating Maximizatio [Emery-Montemerlo *et al.*, 2004]

We measured the amount of time required to generate a policy along with the expected utility of the policy.

4.3 Results

Coordination rules provided an order-of-magnitude time savings with minimal effects on solution quality. In each domain, the solver with the coordination rule was able to find a solution for a longer horizon than the uncoordinated case. Furthermore, Figure 2 illustrates that on problems with the same planning horizon that were solved in a non-trivial amount of time, the problems without coordination rules took significantly longer to solve. In the most extreme example, in the recycling robots domain with a planning horizon of four, the solution time with the coordination rule was 22 seconds. Without the rule, it took 86 minutes.

Figure 2: Comparison of planning times with and without coordination rules.



Regarding optimal solution quality, for each planning horizon that generated a policy in each condition, the policy with the coordination rules was on average 3 percent worse than the baseline policy. In the worst case, the coordination rule resulted in a solution that was 6 percent worse for the longest planning horizon on the recycling robots. This is because the coordination rule prevents the highest reward action from being taken.

Regarding approximate solution quality, the results support the claim that coordination rules may result in improved performance when used in conjunction with approximate solvers. In the missile strike scenario, the policies generated by the approximate solution method, were 1 percent better with the coordination rule than those without. This is interesting because the expected value of the optimal policy without the coordination rule is 1 percent better.

5 Related Work

Our work builds on research from the knowledge representation, planning and agents communities. An early approach to support multi-agent collaboration involves specifying joint commitments and mutual belief [Cohen and Levesque, 1990]. Joint commitments provide a mechanism for an agent to promise to another agent to either attain a state of mutual belief that the commitment has been satisfied or communicate that it is impossible. The key benefit of this framework is that it prevents agents from dropping out of collaborative behavior as soon as uncertainty arises as they know they will be told by their teammates if they are working toward impossible goals.

Building on this approach is the idea of SharedPlans [Grosz and Kraus, 1996]. SharedPlans were originally developed to further our understanding of dialog systems, but have been extended to be a general model of collaborative behavior [Grosz and Kraus, 1999]. SharedPlans evolve during execution and maintain the mutual beliefs of the collaborating agents and important dependencies between their future actions.

Woolridge and Jennings [Wooldridge and Jennings, 1999] present a definition of the cooperative problem-solving process that goes beyond the focus of this work to include steps for the recognition of a collaborative problem and for team formation. Following the other works, the author's emphasize the importance of joint commitments, but they also introduce the concept of *conventions*, which are understandings within the team when commitments may be abandoned. Coordination rules can act as a convention by ruling out particular actions under certain circumstances.

While the above approaches use logics to support collaborative behavior, others in the multi-agent community view collaboration as a series of decisions made by individual agents. These include the decentralized partially observable Markov decision processes [Bernstein *et al.*, 2000] and communicative multi-agent team decision processes (COM-MTDP) [Pynadath and Tambe, 2002]. Using a helicopter escort scenario, the authors show how, by analyzing a COM-MTDP, different communication policies can be compared. Our work contributes to these lines of work in two ways: (1) we construct the state action space through our plan-with-options domain representation and the envisionment process, and (2) we introduce a heuristic that reduces the state action space by using predictability to limit individual actions. Further study of the cost and benefits of the heuristic in general problems is an important aspect of future work.

6 Conclusion

As cognitive systems become increasing prevalent in our environment, our ability to collaborate with them as well as their ability to collaborate with each other will determine their utility to society. We have shown that plan-with-options provides a natural way to specify the space of coordinated behaviors without removing agent autonomy and, through interaction- based factored envisionment, we can compactly represent the effects of multi-agent plan executions. From this envision- ment, we defined two classes of within-team uncertainty met- rics, discussed how they work on different types of scenarios, and how they enable the selection of coordination rules. Finally, we demonstrated that coordination rules facilitate action selection by removing complexity from the state action space. This accelerates planning times and allows for longer planning horizons while having minimal effects on plan quality.

The generation and application of coordination rules should facilitate collaboration in many different multi-agent action selection formalisms. Therefore, in addition to exploring different scenarios, future work should address different action selection problem formulations and solution methods. Also, while we have defined two classes of uncertainty metrics, we have not defined automated selection criteria. Furthermore, there are clearly domain specific methods that should also be explored (e.g., reducing uncertainty at the expected point of future communication). Finally, to design systems of collaborating agents, action models will likely exist at multiple scales. In this work, we discuss briefly how qualitative evaluation of state-action space can be used to generate coordination rules that simplify the action selection problem when more detailed models are available. The automatic generation and curation of these multi-level models will be an essential problem for long-lived collaborative cognitive systems.

Acknowledgments

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

[Alford *et al.*, 2009] Ronald Alford, Ugur Kuter, and Dana S Nau. Translating HTNs to PDDL: A small amount of domain knowledge can go a long way. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1629–1634, 2009.

[Bernstein *et al.*, 2000] Daniel S Bernstein, Shlomo Zilberstein, and Neil Immerman. The complexity of decentralized control of markov decision processes. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 32–37. Morgan Kaufmann Publishers Inc., 2000.

[Clancy and Kuipers, 1997] Daniel J Clancy and Benjamin Kuipers. Model decomposition and simulation: A component based qualitative simulation algorithm. In *AAAI/IAAI*, pages 118–124, 1997.

[Cohen and Levesque, 1990] Philip R Cohen and Hector J Levesque. Intention is choice with commitment. *Artificial intelligence*, 42(2):213–261, 1990.

[Do and Kambhampati, 2003] Minh Binh Do and Subbarao Kambhampati. Sapa: A multi-objective metric temporal planner. *J. Artif. Intell. Res.(JAIR)*, 20:155–194, 2003.

[Emery-Montemerlo *et al.*, 2004] Rosemary Emery-Montemerlo, Geoff Gordon, Jeff Schneider, and Sebastian Thrun. Approximate solutions for partially observable stochastic games with common payoffs. In *Autonomous Agents and Multiagent Systems, 2004. AAMAS 2004. Proceedings of the Third International Joint Conference on*, pages 136–143. IEEE, 2004.

[Erol *et al.*, 1994] Kutluhan Erol, James Hendler, and Dana S Nau. Htn planning: Complexity and expressivity. In *AAAI*, volume 94, pages 1123–1128, 1994.

[Ghallab *et al.*, 2004] Malik Ghallab, Dana Nau, and Paolo Traverso. *Automated planning: theory & practice*. Elsevier, 2004.

[Grosz and Kraus, 1996] Barbara J Grosz and Sarit Kraus. Collaborative plans for complex group action. *Artificial Intelligence*, 86(2):269–357, 1996.

[Grosz and Kraus, 1999] Barbara J Grosz and Sarit Kraus. The evolution of sharedplans. In *Foundations of rational agency*, pages 227–262. Springer, 1999.

[Hinrichs *et al.*, 2011] Thomas R Hinrichs, Kenneth D Forbus, Johan de Kleer, Sungwook Yoon, Eric Jones, Robert Hyland, and Jason Wilson. Hybrid qualitative simulation of military operations. In *IAAI*, 2011.

[Kuipers, 1994] Benjamin Kuipers. *Qualitative reasoning: modeling and simulation with incomplete knowledge*. MIT Press, Cambridge, MA, USA, 1994.

[Langley, 2012] Pat Langley. The cognitive systems paradigm. *Advances in Cognitive Systems*, 1:3–13, 2012.

[McDermott *et al.*, 1998] Drew McDermott, Malik Ghallab, Adele Howe, Craig Knoblock, Ashwin Ram, Manuela Veloso, Daniel Weld, and David Wilkins. Pddl-the planning domain definition language. 1998.

[Oliehoek *et al.*, 2008] Frans A Oliehoek, Matthijs TJ Spaan, and Nikos A Vlassis. Optimal and approximate q-value functions for decentralized pomdps. *J. Artif. Intell. Res.(JAIR)*, 32:289–353, 2008.

[Pynadath and Tambe, 2002] David V. Pynadath and Milind Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of Artificial Intelligence Research*, pages 389–423, 2002.

[Spaan and Oliehoek, 2008] Matthijs TJ Spaan and Frans A Oliehoek. The multiagent decision process toolbox: software for decision-theoretic planning in multiagent systems. In *Proc. of the AAMAS Workshop on Multi-Agent Sequential Decision Making in Uncertain Domains (MSDM)*, pages 107–121, 2008.

[Weld and de Kleer, 1989] D.S. Weld and J. de Kleer. *Readings in Qualitative Reasoning about Physical Systems*. Morgan Kaufmann, 1989.

[Wooldridge and Jennings, 1999] Michael Wooldridge and Nicholas R Jennings. The cooperative problem-solving process. *Journal of Logic and Computation*, 9(4):563–592, 1999.