Planning Over Multi-Agent Epistemic States: A Classical Planning Approach

Christian Muise*, Vaishak Belle†, Paolo Felli*, Sheila McIlraith†
Tim Miller*, Adrian R. Pearce*, Liz Sonenberg*

*Department of Computing and Information Systems, University of Melbourne
†Department of Computer Science, University of Toronto

To Be Announced! Synthesis of Epistemic Protocols
August 19, 2015
2015 marks the 50th anniversary of the **Shakey Project**, which resulted in such innovations as A* search and STRIPS planning.

We’ve seen **substantial advances** in AI Automated Planning:

- compact state and transition system encodings
- advances in heuristic search and SAT
- plans with hundreds of actions
- ... hundreds of objects
- ... $2^{1000}$ states
- plan computation ranging from milliseconds to a few hours
2015 marks the 50th anniversary of the **Shakey Project**, which resulted in such innovations as A* search and STRIPS planning.

We’ve seen **substantial advances** in AI Automated Planning:

- compact state and transition system encodings
- advances in heuristic search and SAT
- plans with hundreds of actions
- ... hundreds of objects
- ... $2^{1000}$ states
- plan computation ranging from milliseconds to a few hours

Top-performing planners exploit **heuristic search**, though planning-tailored **SAT-based planners** are closing the gap.

Most of these advances have been in **classical planning**.
The Task

Synthesize plans, from the perspective of a single agent, that necessitate reasoning about the (nested) beliefs of other agents.

Desiderata:

- planning is at the belief level,
- goals and actions involve nested beliefs and respect KD45,
- agents have differing beliefs and differing capabilities, including the ability to perform communication actions and actions that are not public,
- an agent can reason as if it were another agent.
Example Goal: Deception

Make Bob believe that Sue believes the switch is off, when in fact Bob believes that it is on:

\{ B_{Bob} B_{Sue} \neg \text{switch\_on}, B_{Bob} \text{switch\_on} \} 

Example Action: Gossiping

Pre-condition for share(Bob, secret, roomA) includes that Bob believes the secret:

\( B_{Bob} \text{secret} \)

Effects who perceives the gossip (and who is aware of this):

\( \text{in}(Sue, roomA) \rightarrow B_{Sue} \text{secret}, \)

\( B_{Joe} \text{in}(Sue, roomA) \land \text{in}(Joe, roomA) \rightarrow B_{Joe} B_{Sue} \text{secret}, \ldots \)
Contribution

We formally characterize a notion of Multi-agent Epistemic Planning (MEP), and demonstrate how to solve a rich subclass of these problems using state-of-the-art classical planning techniques.
Outline

1. How To Perform Epistemic Planning?
2. Background & Notation
3. Restricted Perspectival Multi-Agent Epistemic Planning
4. Projecting to Others
5. Implementation
6. Summary
Outline

1 How To Perform Epistemic Planning?

2 Background & Notation

3 Restricted Perspectival Multi-Agent Epistemic Planning

4 Projecting to Others

5 Implementation

6 Summary
Option 1: Construct a Customized Planner

- (pro) Can directly handle epistemic reasoning
- (con) Rigid (forced to use one planner)
How To Perform Epistemic Planning?

Option 1: Construct a Customized Planner

- (pro) Can directly handle epistemic reasoning
- (con) Rigid (forced to use one planner)

Option 2: Cleverly Encode as Classical Planning

- (con) Must devise an encoding that ensures certain properties
  - Consistent belief
  - Logical closure
  - Etc...
- (pro) Planner agnostic – leverage every advance made!
How To Perform Epistemic Planning?

Option 1: Construct a Customized Planner
- **(pro)** Can directly handle epistemic reasoning
- **(con)** Rigid (forced to use one planner)

Option 2: Cleverly encode as Classical Planning
- **(con)** Must devise an encoding that ensures certain properties
  - Consistent belief
  - Logical closure
  - Etc...
- **(pro)** Planner agnostic – leverage every advance made!
Outline

1. How To Perform Epistemic Planning?
2. Background & Notation
3. Restricted Perspectival Multi-Agent Epistemic Planning
4. Projecting to Others
5. Implementation
6. Summary
Classical Planning

Planning Problem $\langle F, G, I, O \rangle$

$F$: set of fluent atoms
$G$: set of fluents describing the goal condition
$I$: set of fluents describing the initial state
$O$: set of operators of the form $\langle Pre, eff^+, eff^- \rangle$

Operator elements

$Pre$: set of fluents for the precondition
$eff^+$: set of conditional effects that add a fluent
$eff^-$: set of conditional effects that delete a fluent

$\left( \langle C^+, C^- \rangle \rightarrow I \right)$: conditional effect that fires when $C^+$ holds and $C^-$ does not hold
E.g., **PickupBlock**

If the agent is strong and the block is not slippery, then the agent holds the block: i.e., $\text{eff}^+$ contains

$$\left( \langle \{\text{strong}\}, \{\text{slippery}\} \rangle \rightarrow \text{holding\_block} \right)$$

If the block is big, then the agent’s hand will no longer be free (i.e., we should delete the $\text{hand\_free}$ fluent): i.e., $\text{eff}^-$ contains

$$\left( \langle \{\text{big\_block}\}, \emptyset \rangle \rightarrow \text{hand\_free} \right)$$

**Note:** We distinguish between $C^+/C^-$ and $\text{eff}^+/\text{eff}^-$ in our work so that our encoding is more legible
Outline

1. How To Perform Epistemic Planning?

2. Background & Notation

3. Restricted Perspectival Multi-Agent Epistemic Planning

4. Projecting to Others

5. Implementation

6. Summary
Restricted Perspectival MEP

Two restrictions:

- No disjunctive beliefs
- Bound depth of nested belief – KD45$_n$
General Approach

1. Restrict state, conditions and goals to *Proper Doxastic Knowledge Bases*
General Approach

1. Restrict state, conditions and goals to *Proper Doxastic Knowledge Bases*

2. Enforce KD45\(_n\) and realize belief revision by appealing to a solution to the *Ramification Problem*
General Approach

1. Restrict state, conditions and goals to *Proper Doxastic Knowledge Bases*
2. Enforce $KD45_n$ and realize belief revision by appealing to a solution to the *Ramification Problem*
3. Encode conditions for mutual awareness as conditional effects
General Approach

1. Restrict state, conditions and goals to Proper Doxastic Knowledge Bases
2. Enforce KD45\textsubscript{n} and realize belief revision by appealing to a solution to the Ramification Problem
3. Encode conditions for mutual awareness as conditional effects

**Result:** A classical planning problem we can input to any off-the-shelf solver.

**Theorem\*:** The approach is always sound and complete for problems that do not require disjunction

\* stated informally
1. Restrict to Proper Doxastic Knowledge Bases

**Definition: Proper Doxastic Knowledge Base (PDKB)**

A set or conjunction of *Restricted Modal Literals.*

No disjunction!
1. Restrict to Proper Doxastic Knowledge Bases

**Definition: Proper Doxastic Knowledge Base (PDKB)**

A set or conjunction of *Restricted Modal Literals*.

**Definition: Restricted Modal Literal (RML)**

\[ \phi ::= p \mid \neg \phi \mid B_{ag} \phi \]

- \(ag \in Ag\): A particular agent
- \(p \in P\): An original fluent without belief

E.g., “Sue believes it is raining”: \(B_{Sue} raining\)
1. Restrict to Proper Doxastic Knowledge Bases

**Definition: Proper Doxastic Knowledge Base (PDKB)**

A set or conjunction of *Restricted Modal Literals*.

**Definition: Restricted Modal Literal (RML)**

\[ \phi ::= p \mid \neg \phi \mid B_{ag} \phi \]

* \( ag \in Ag \): A particular agent
* \( p \in P \): An original fluent without belief

E.g., “Sue believes it is raining”: \( B_{Sue} \text{raining} \)

**Encoding**

- Create a fluent for every RML (bounded depth and agents)
- States, preconditions, effects, & goals are PDKBs.

* - PEKB [Lakemeyer & Lespérance, 2012]
2. Enforce $\text{KD45}_n$ and Belief Revision via Ramifications

Treat $\text{KD45}_n$ axioms and belief revisions as indirect effects of actions.
Treat $\text{KD45}_n$ axioms and belief revisions as indirect effects of actions.

**Potential Solutions to the Ramification Problem:**
2. Enforce KD45\(_n\) and Belief Revision via Ramifications

Treat KD45\(_n\) axioms and belief revisions as indirect effects of actions.

Potential Solutions to the Ramification Problem:

<table>
<thead>
<tr>
<th>Option 1: Derived Predicates</th>
</tr>
</thead>
<tbody>
<tr>
<td>(pro) Supported by some planners</td>
</tr>
<tr>
<td>(con) Not a general solution</td>
</tr>
</tbody>
</table>
2. Enforce KD45\textsubscript{n} and Belief Revision via Ramifications

Treat KD45\textsubscript{n} axioms and belief revisions as indirect effects of actions.

**Potential Solutions to the Ramification Problem:**

**Option 1: Derived Predicates**
- **(pro)** Supported by some planners
- **(con)** Not a general solution

**Option 2: Ramification Actions**
- **(pro)** Heuristic Guidance
- **(con)** Not supported by planners
2. Enforce KD45\(_n\) and Belief Revision via Ramifications

Treat KD45\(_n\) axioms and belief revisions as indirect effects of actions.

**Potential Solutions to the Ramification Problem:**

<table>
<thead>
<tr>
<th>Option 1: Derived Predicates</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>(pro)</em> Supported by some planners</td>
</tr>
<tr>
<td><em>(con)</em> Not a general solution</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option 2: Ramification Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>(pro)</em> Heuristic Guidance</td>
</tr>
<tr>
<td><em>(con)</em> Not supported by planners</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option 3: Compile into Ancillary Conditional Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>(pro)</em> Standard Approach [Pinto, 1999]</td>
</tr>
<tr>
<td><em>(con)</em> Size for non-binary constraints</td>
</tr>
</tbody>
</table>
Anatomy of an Ancillary Effect

If $eff^+$ (or $eff^-$) contains some conditional effect $cond_1$ then we will add to $eff^+$ (or $eff^-$) another effect $cond_2$:

$$cond_1 = (\langle C_1^+, C_1^- \rangle \rightarrow l_1)$$
$$cond_2 = (\langle C_2^+, C_2^- \rangle \rightarrow l_2)$$

$$(\langle C_1^+, C_1^- \rangle \rightarrow l_1) \in eff^+ (or \ eff^-) \Rightarrow (\langle C_2^+, C_2^- \rangle \rightarrow l_2) \in eff^+ (or \ eff^-)$$
If you add belief, forget about its negation (belief update).

\[(\langle C^+, C^- \rangle \rightarrow l) \in \text{eff}^+ \]

\[\Rightarrow\]

\[(\langle C^+, C^- \rangle \rightarrow \neg l) \in \text{eff}^- \]

\[(\langle \{at\_a\_room_1\}, \emptyset \rangle \rightarrow B_{a secret_b}) \in \text{eff}^+ \]

\[\Rightarrow\]

\[(\langle \{at\_a\_room_1\}, \emptyset \rangle \rightarrow \neg B_{a secret_b}) \in \text{eff}^- \]
KD45 Logical Closure

The state should always be deductively closed.

\[
(\langle C^+, C^- \rangle \rightarrow l) \in \text{eff}^+ \\
\Rightarrow \\
\forall l' \in \text{Closure}(l), (\langle C^+, C^- \rangle \rightarrow l') \in \text{eff}^+
\]

\[
(\langle \{\text{at}_a\_\text{room}_1\}, \emptyset \rangle \rightarrow B_a\neg\text{secret}_b) \in \text{eff}^+ \\
\Rightarrow (\langle \{\text{at}_a\_\text{room}_1\}, \emptyset \rangle \rightarrow \neg B_a\neg\text{secret}_b) \in \text{eff}^+
\]
Removing belief should also keep the state deductively closed.

\[
\begin{align*}
\langle \langle C^+, C^- \rangle \rightarrow l \rangle & \in eff^- \\
\Rightarrow \forall l' \in Closure(\neg l), \langle \langle C^+, C^- \rangle \rightarrow \neg l' \rangle & \in eff^-
\end{align*}
\]

\[
\begin{align*}
\langle \langle \emptyset, \emptyset \rangle \rightarrow \neg B_a\text{secret} \rangle & \in eff^- \\
\Rightarrow \langle \langle \emptyset, \emptyset \rangle \rightarrow B_a\neg\text{secret} \rangle & \in eff^-
\end{align*}
\]
Uncertain Firing

Be aware of any effect that may have fired.

\[
(\langle C^+, C^- \rangle \rightarrow l) \in \text{eff}^+
\]

\[
\Rightarrow
\]

\[
(\langle \emptyset, \{\neg \phi \mid \phi \in C^+\} \cup C^- \rangle \rightarrow \neg l) \in \text{eff}^-
\]

\[
(\langle \{\text{at}_a\_room}_1\}, \emptyset \rangle \rightarrow B_a\text{secret}_b) \in \text{eff}^+
\]

\[
\Rightarrow (\langle \emptyset, \{\neg \text{at}_a\_room}_1\} \rangle \rightarrow \neg B_a\text{secret}_b) \in \text{eff}^-
\]
Uncertain Firing

Be aware of any effect that *may* have fired.

\[
\begin{align*}
(\langle C^+, C^- \rangle \rightarrow l) &\in eff^+ \\
\Rightarrow \\
(\langle \emptyset, \{\neg \phi \mid \phi \in C^+ \} \cup C^- \rangle \rightarrow \neg l) &\in eff^-
\end{align*}
\]

\[
\begin{align*}
(\langle \{at_a\_room_1\}, \emptyset \rangle \rightarrow B_a secret_b) &\in eff^+ \\
\Rightarrow (\langle \emptyset, \{\neg at_a\_room_1\} \rangle \rightarrow \neg B_a secret_b) &\in eff^-
\end{align*}
\]

Similar to the conformant planning technique:

\[
p \rightarrow q \Rightarrow !K!p \rightarrow !K!q
\]
3. Encode Conditioned Mutual Awareness as Conditional Effects

**Idea:** Given a condition $\mu_i$ for agent $i$ to witness an action, add the conditional effects for us to update our belief about agent $i$. 
3. Encode Conditioned Mutual Awareness as Conditional Effects

**Idea:** Given a condition $\mu_i$ for agent $i$ to witness an action, add the conditional effects for us to update our belief about agent $i$.

\[
\begin{align*}
(\langle C^+, C^- \rangle \rightarrow l) \in \text{eff}^+ \\
\Rightarrow \\
(\langle B_i^+ C^+ \cup \neg B_i^+ C^- \cup \{ B_i \mu_i \}, \emptyset \rangle \rightarrow B_i l) \in \text{eff}^+
\end{align*}
\]

\[
\begin{align*}
(\langle C^+, C^- \rangle \rightarrow l) \in \text{eff}^- \\
\Rightarrow \\
(\langle B_i^+ C^+ \cup \neg B_i^+ C^- \cup \{ B_i \mu_i \}, \emptyset \rangle \rightarrow \neg B_i l) \in \text{eff}^+
\end{align*}
\]
Conditioned Mutual Awareness: Examples

Possible values for $\mu_i$

- **in_room_i**: Agent $i$ observes the effect if they are in the room (i.e., physically present).
  - **True**: Agent $i$ always observes the effect
  - **False**: Agent $i$ never observes the effect

$$\mu_c = \text{True}$$

$$(\langle \emptyset, \{\neg \text{at_a_room}_1\} \rangle \rightarrow \neg B_a \text{secret}_b) \in \text{eff}^-$$

$$\Rightarrow (\langle \{\neg B_c \neg \text{at_a_room}_1\}, \emptyset \rangle \rightarrow \neg B_c \neg B_a \text{secret}_b) \in \text{eff}^+$$
Outline

1. How To Perform Epistemic Planning?
2. Background & Notation
3. Restricted Perspectival Multi-Agent Epistemic Planning
4. Projecting to Others
5. Implementation
6. Summary
**Idea:** Project our beliefs about the world and action effects to reason *as if* we were another agent.

- Can be repeated to depths greater than 1
- Represents our view of how another will reason
- Works in concert with Conditioned Mutual Awareness
Projecting to Others

**Idea:** Project our beliefs about the world and action effects to reason *as if* we were another agent.

**Example:** “Does Sue believe that Bob believes that Sue believes ...”

Sue is the root agent. [Bob, Sue] is the *virtual agent* the root agent will reason about. We want Sue reasoning as if she was Bob reasoning as if he was Sue.

---

**Projecting a PDKB state**

\[ \text{Proj}(s, \vec{Ag}) : \text{Projecting state } s \text{ for agent sequence } \vec{Ag} : \]

\[
\begin{align*}
\begin{cases}
\{ \phi \mid B_i \phi \in s \} & \text{if } \vec{Ag} = [i] \\
\text{Proj(Proj}(s, [i]), \vec{Ag}') & \text{if } \vec{Ag} = [i] + \vec{Ag}'
\end{cases}
\end{align*}
\]
Idea: Project our beliefs about the world and action effects to reason as if we were another agent.

Example: “Does Sue believe that Bob believes that Sue believes …”

Sue is the root agent. [Bob, Sue] is the virtual agent the root agent will reason about. We want Sue reasoning as if she was Bob reasoning as if he was Sue.

Projecting a PDKB state

Projecting an Action

Projecting effects for agent $i$ works in a similar fashion, but can only be done if the effect is uniform in $i$:

$C^- = \emptyset$ and all RMLs in the effect begin with $B_i$. 
Outline

1. How To Perform Epistemic Planning?
2. Background & Notation
3. Restricted Perspectival Multi-Agent Epistemic Planning
4. Projecting to Others
5. Implementation
6. Summary
Example Action Encoding: Sharing a Secret

!(:action share_bob_secret3_room2

:derive-condition at_?_room2

:precondition (and
  (at_bob_room2)
  [bob](secret3))

:effect (and
  (when (at_sue_room2)
    [sue](secret3))
  (when (at_ann_room2)
    [ann](secret3))))
Preliminary Evaluation

- Restricted Perspectival MEP problems specified in custom format and compiled for input to classical planners.
- Exploits the $SIW^+$ and $BFS_f$ planners [Lipovetzky & Geffner, 2012] (LAPKT planning library [Ramirez et al., 2015]).
- Simulates execution of an action sequence or generates a plan.
- Tested on several domains, including:
  - *Thief* domain model verified and pre-existing queries trivially solved (e.g., [Löwe, Pacuit, and Witzel, 2011])
  - *Corridor* domain [Kominis & Geffner, 2015]
  - *Grapevine* domain (combination of *Corridor* and *Gossip*)
  - Ask Christian about others!

- Time to generate plans in these more challenging domains remains small. Encoding and parsing times dominates. The max depth of nested belief, $d$, produces the bottleneck since the number of newly introduced fluents is exponential in $d$. 
Let $a$, $b$, and $c$ be three agents in a corridor of four rooms ($p_1, p_2, p_3, p_4$) from left to right. The agents can move from a room to a contiguous room, and when agent $i$ communicates some information, all the agents that are in the same room or a contiguous room, will hear what was said. E.g., if $i$ expresses his knowledge about $q$ in room $p_3$, all agents in rooms $p_2$, $p_3$ and $p_4$ will come to know it. $a$, $b$ and $c$ are initially in rooms $p_1$, $p_2$ and $p_3$, respectively, and $a$ has to find out the truth value of a proposition $q$ and let $c$ know without $b$ learning it.
**Grapevine Domain** *(Corridor + Gossip)*

There are two rooms with all agents starting in the first room. Every agent believes their own secret to begin with, and the agents can either move between rooms or broadcast a secret they believe. Movement is always observed by all, but the sharing of a secret is only observed by those in the same room. Goals include private communication and also misconception (e.g., \{B_a s_b, B_b \neg B_a s_b\})*
## Experiments

| Problem    | $|Ag|$ | $d$ | $|F|$ | $|\bar{o}|$ | Time (s) |
|------------|------|-----|------|-------|----------|
|            |      |     |      |       | Plan     | Total    |
| Corridor   | 3    | 1   | 70   | 5     | 0.01     | 0.11     |
|            | 7    | 1   | 150  | 5     | 0.01     | 0.21     |
|            | 3    | 3   | 2590 | 5     | 0.05     | 6.85     |
| Grapevine  | 4    | 1   | 216  | 10    | 0.04     | 0.27     |
|            | 3    | 2   | 774  | 4     | 0.09     | 1.84     |
|            | 4    | 2   | 1752 | 4     | 0.70     | 6.61     |

$Ag$: The number of agents included  
$d$: Maximum depth of nested belief  
$F$: Number of compiled fluents  
$\bar{o}$: Computed plan
Multi-agent Epistemic Planning With Proper Doxastic Knowledge Bases

Positive Condition

Negative Condition

Condition Effect

Mutual Awareness

Effect Type

Positive Effects

Negative Effects

ID | Pos Cond | Neg Cond | Effect | Source
---|----------|----------|--------|-------
89631 | mike in room_mike | sue in room_mike | ⊗_mike ⊗_sue light_on | (40604) Closure
11672 | bob in room_bob | mike in room_mike | ⊗_bob ⊗_mike light_on | (63000) Commonly Known
49907 | mike in room_mike | sue in room_mike | ⊗_mike ⊗_sue light_on | (81220) Commonly Known
48577 | bob in room_bob |  | ⊗_bob light_on | (42873) Closure

ID | Pos Cond | Neg Cond | Effect | Source
---|----------|----------|--------|-------
49988 | sue in room_sue |  | ⊗_sue ¬ light_on | (61097) Remove Negations
12292 | mike bob in room_bob | mike in room_mike | ⊗_mike ⊗_bob ¬ light_on | (67269) Remove Negations
68604 | bob in room_bob | sue in room_sue | ⊗_bob ¬ light_on | (34766) Remove Negations
### Lineage of Effect

<table>
<thead>
<tr>
<th>Type</th>
<th>ID</th>
<th>Pos Cond</th>
<th>Neg Cond</th>
<th>Effect</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>30329</td>
<td></td>
<td></td>
<td>light_on</td>
<td>Original Effect</td>
</tr>
<tr>
<td>+</td>
<td>51050</td>
<td>□_mike in_room_mike</td>
<td>□_mike light_on</td>
<td></td>
<td>(30329) Commonly Known</td>
</tr>
<tr>
<td>+</td>
<td>42628</td>
<td>□_mike in_room_mike</td>
<td>◊_mike light_on</td>
<td></td>
<td>(51050) Closure</td>
</tr>
<tr>
<td>-</td>
<td>63000</td>
<td>◊_mike \neg in_room_mike</td>
<td>□_mike \neg light_on</td>
<td></td>
<td>(42628) Uncertain Firing</td>
</tr>
<tr>
<td>+</td>
<td>11672</td>
<td>□_bob in_room_bob</td>
<td>◊_bob ◊_mike in_room_mike</td>
<td>◊_bob ◊_mike light_on</td>
<td>(63000) Commonly Known</td>
</tr>
</tbody>
</table>

### Negative Effects

<table>
<thead>
<tr>
<th>Neg Cond</th>
<th>Effect</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>◊_mike ◊_sue light_on</td>
<td></td>
<td>(40604) Closure</td>
</tr>
<tr>
<td>◊_bob ◊_mike light_on</td>
<td></td>
<td>(63000) Commonly Known</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Pos Cond</th>
<th>Neg Cond</th>
<th>Effect</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>49988</td>
<td>□_sue in_room_sue</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12292</td>
<td>□_mike □_bob in_room_bob</td>
<td>□_mike in_room_mike</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Outline

1. How To Perform Epistemic Planning?
2. Background & Notation
3. Restricted Perspectival Multi-Agent Epistemic Planning
4. Projecting to Others
5. Implementation
6. Summary
Conclusion

- Multi-agent planning settings often require us to model the nested belief of agents

- We leveraged a tractable fragment of epistemic reasoning to maintain consistency of agents’ belief

- Realized an automated planning system that deals effectively with the nested belief in a multi-agent setting
Future Work

Lots!
Demo, benchmarks, code, and slides available at:
http://www.haz.ca/research/pdkb/
Some References and Follow-on Work I

Computing social behaviours using agent models.
In International Joint Conference on Artificial Intelligence, IJCAI 2015, 2015.

F. Kominis and H. Geffner.
Beliefs in multiagent planning: From one agent to many.

G. Lakemeyer and Y. Lespérance.
Efficient reasoning in multiagent epistemic logics.

N. Lipovetzky and H. Geffner.
Width and serialization of classical planning problems.

B. Löwe, E. Pacuit, and A. Witzel.
DEL planning and some tractable cases.

Planning over multi-agent epistemic states: A classical planning approach.
In The 29th AAAI Conference on Artificial Intelligence, 2015.
In Workshop on Distributed and Multi-Agent Planning (DMAP’15), 2015.

Towards team formation via automated planning.
In International Workshop on Coordination, Organisation, Institutions and Norms in Multi-Agent Systems, 2015.

Leveraging fond planning technology to solve multi-agent planning problems.
In Workshop on Distributed and Multi-Agent Planning (DMAP’15), 2015.

Efficient reasoning with consistent proper epistemic knowledge bases.
In The International Conference on Autonomous Agents and Multiagent Systems, 2015.

J. Pinto.
Compiling ramification constraints into effect axioms.

M. Ramirez, N. Lipovetzky, and C. Muise.
Lightweight Automated Planning ToolKiT.