Multiagent Planning as an Emerging Behavior in Agent Societies

Roberto Micalizio

Cristina Baroglio Matteo Baldoni

Dipartimento di Informatica
Università di Torino

NorMAS 2014
Outline

- **Motivations and Idea**
  - Multiagent planning as Social Computing

- **Background**
  - Classical Planning
  - Social Commitments & Goals

- **Social Continual Planning** by examples
Motivations

- Multiagent planning: synthesis of plans for a number of agents in a given *team*
  - each agent reaches its own goals
  - the agent plans are altogether consistent (i.e., no deadlock, no open preconditions, correct usage of resources)

- Multiagent planning as *distributed problem solving*:
  - agents are homogeneous
  - agents can trust each other
  - agents can inspect each other their beliefs
  - agents do not change over time (the team is fixed at the beginning)
    $\Rightarrow$ agents are not really autonomous

These assumptions are unpractical when agents constitute a *society* rather than a team
Motivations

- Multiagent planning: synthesis of plans for a number of agents in a given team
  - each agent reaches its own goals
  - the agent plans are altogether consistent (i.e., no deadlock, no open preconditions, correct usage of resources)

- Multiagent planning as *distributed problem solving*:
  - agents are homogeneous
  - agents can trust each other
  - agents can inspect each other their beliefs
  - agents do not change over time (the team is fixed at the beginning)
  \[\Rightarrow\] *agents are not really autonomous*

These assumptions are unpractical when agents constitute a *society* rather than a team
Multiagent Planning as Social Computing

Idea:
- Enrich the (classical) BDI planning agent with social capabilities
- The planning system is thought of as a normative system:
  - social norms define the constraints within which agents can operate
  - an agent’s plan must be “socially acceptable”

How to get there:
- use of social commitments for modeling agent interactions

Why?
- commitments have a normative power
  \(\Rightarrow\) an agent can create expectations on the behaviors of others just relying on the active commitments
- commitments are tightly related to goals [Telang et al. 2011]
  \(\Rightarrow\) a planning agent can be driven by the commitments it is responsible for
- commitments enable practical reasoning, that can be seen as a form of planning
Multiagent Planning as Social Computing

**Idea:**
- Enrich the (classical) BDI planning agent with *social capabilities*
- The planning system is thought of as a *normative system*:
  - social norms define the constraints within which agents can operate
  - an agent’s plan must be “socially acceptable”

**How to get there:**
- use of *social commitments* for modeling agent interactions

**Why?**
- commitments have a normative power
  - an agent can create expectations on the behaviors of others just relying on the active commitments
- commitments are tightly related to goals [Telang et al. 2011]
  - a planning agent can be driven by the commitments it is responsible for
- commitments enable *practical reasoning*, that can be seen as a form of planning
Multiagent Planning as Social Computing

Idea:
- Enrich the (classical) BDI planning agent with *social capabilities*
- The planning system is thought of as a *normative system*:
  - social norms define the constraints within which agents can operate
  - an agent’s plan must be “socially acceptable”

How to get there:
- use of *social commitments* for modeling agent interactions

Why?
- commitments have a normative power
  - an agent can create expectations on the behaviors of others just relying on the active commitments
- commitments are tightly related to goals [Telang et al. 2011]
  - a planning agent can be driven by the commitments it is responsible for
- commitments enable *practical reasoning*, that can be seen as a form of planning
Multiagent Planning as Social Computing

**Idea:**
- Enrich the (classical) BDI planning agent with *social capabilities*
- The planning system is thought of as a *normative system*:
  - social norms define the constraints within which agents can operate
  - an agent’s plan must be “socially acceptable”

**How to get there:**
- use of *social commitments* for modeling agent interactions

**Why?**
- commitments have a normative power
  - an agent can create expectations on the behaviors of others just relying on the active commitments
- commitments are tightly related to goals [Telang et al. 2011]
  - a planning agent can be driven by the commitments it is responsible for
- commitments enable *practical reasoning*, that can be seen as a form of planning
Multiagent Planning as Social Computing

**Idea:**
- Enrich the (classical) BDI planning agent with *social capabilities*
- The planning system is thought of as a *normative system*:
  - social norms define the constraints within which agents can operate
  - an agent’s plan must be “socially acceptable”

**How to get there:**
- use of *social commitments* for modeling agent interactions

**Why?**
- commitments have a normative power
  - an agent can create expectations on the behaviors of others just relying on the active commitments
- commitments are tightly related to goals [Telang et al. 2011]
  - a planning agent can be driven by the commitments it is responsible for
- commitments enable *practical reasoning*, that can be seen as a form of planning
Multiagent Planning as Social Computing

**Idea:**
- Enrich the (classical) BDI planning agent with *social capabilities*
- The planning system is thought of as a *normative system*:
  - social norms define the constraints within which agents can operate
  - an agent’s plan must be “socially acceptable”

**How to get there:**
- use of *social commitments* for modeling agent interactions

**Why?**
- commitments have a normative power
  - an agent can create expectations on the behaviors of others just relying on the active commitments
- commitments are tightly related to goals [Telang et al. 2011]
  - a planning agent can be driven by the commitments it is responsible for
- commitments enable *practical reasoning*, that can be seen as a form of planning
Background: Classical Planning

- a single-agent *planning domain* $D : \langle P, S, A, R \rangle$
  - $P$ is the (finite) set of atomic propositions
  - $S \subseteq 2^P$ is the set of possible states
  - $A$ is the (finite) set of actions
  - $R \subseteq S \times A \times S$ is a transition relation

- a single-agent *planning problem* $Pr : \langle D, I, G \rangle$
  - $D$ is the a planning domain
  - $I \subseteq S$ initial state
  - $G \subseteq S$ goal state

- a *solution* $\pi$ for $Pr$ is a sequence of actions $\langle a_1, \ldots, a_n \rangle$ such that:
  - $a_1$ is applicable to the initial state $I$
  - $a_i$ is applicable to the state resulting after the application of $a_{i-1}$ (for $i : 2..n$)
  - $G$ holds after the application of $a_n$
Background: Classical Planning

- a single-agent **planning domain** $D : \langle P, S, A, R \rangle$
  - $P$ is the (finite) set of atomic propositions
  - $S \subseteq 2^P$ is the set of possible states
  - $A$ is the (finite) set of actions
  - $R \subseteq S \times A \times S$ is a transition relation

- a single-agent **planning problem** $Pr : \langle D, I, G \rangle$
  - $D$ is the a planning domain
  - $I \subseteq S$ initial state
  - $G \subseteq S$ goal state

- a **solution** $\pi$ for $Pr$ is a sequence of actions $\langle a_1, \ldots, a_n \rangle$ such that:
  - $a_1$ is applicable to the initial state $I$
  - $a_i$ is applicable to the state resulting after the application of $a_{i-1}$ (for $i : 2..n$)
  - $G$ holds after the application of $a_n$
Background: Classical Planning

- a single-agent *planning domain* $D : \langle P, S, A, R \rangle$
  - $P$ is the (finite) set of atomic propositions
  - $S \subseteq 2^P$ is the set of possible states
  - $A$ is the (finite) set of actions
  - $R \subseteq S \times A \times S$ is a transition relation

- a single-agent *planning problem* $Pr : \langle D, I, G \rangle$
  - $D$ is the a planning domain
  - $I \subseteq S$ initial state
  - $G \subseteq S$ goal state

- a *solution* $\pi$ for $Pr$ is a sequence of actions $\langle a_1, \ldots, a_n \rangle$ such that:
  - $a_1$ is applicable to the initial state $I$
  - $a_i$ is applicable to the state resulting after the application of $a_{i-1}$ (for $i : 2..n$)
  - $G$ holds after the application of $a_n$
Background: Commitments and Goals

Life cycle of a commitment

Life cycle of a goal
the relation between commitments and goals has been captured by a set of rules [Telang et al. 2011]:

- **structural rules**: complete and deterministic, describe how commitment and goal states evolve

- **pragmatical rules**: describe patterns of practical reasoning over commitments and goals; these rules are neither complete nor deterministic
Background: Pragmatical Rules

\[
\frac{\text{guard}}{S_1 \rightarrow S_2}
\]

- \textit{guard} is a condition over an agent beliefs and over the active commitments
- \(S_1 \rightarrow S_2\) is a state transition defining how goals and commitments change
Background: Pragmatical Rules

\[
\frac{guard}{S_1 \rightarrow S_2}
\]

- *guard* is a condition over an agent beliefs and over the active commitments
- \( S_1 \rightarrow S_2 \) is a state transition defining how goals and commitments change

- Pragmatical Rules are divided into:
  - rules from **goals to commitments**
    
    \[
    \langle G^A, C^N \rangle \quad create(C) \quad ENTICE
    \]
  - rules from **commitments to goals**
    
    \[
    \langle G^N, C^D \rangle \quad consider(G), activate(G) \quad DELIVERY
    \]
Reasoning about Goal and Commitments via Social Continual Planning

Main idea:

- Interleave planning phases with execution and negotiation phases.

- The planning phase involves both:
  - "Physical" actions: directly change the world
  - Pragmatical actions: (indirectly) change the social state

- During the execution phase:
  - A physical action is directly performed by an agent
  - A pragmatical action triggers a negotiation with others

- Negotiation involves operations on commitments and it is driven by pragmatical rules.
Reasoning about Goal and Commitments
via Social Continual Planning

Main idea:

- Interleave *planning* phases with *execution* and *negotiation* phases.

- The *planning phase* involves both:
  - "physical" actions: directly change the world
  - Pragmatical actions: (indirectly) change the social state

- During the *execution phase*:
  - A physical action is directly performed by an agent.
  - A pragmatical action triggers a negotiation with others.

- *Negotiation* involves operations on commitments and it is driven by pragmatical rules.
Reasoning about Goal and Commitments via Social Continual Planning

Main idea:

- Interleave \textit{planning} phases with \textit{execution} and \textit{negotiation} phases
- The \textbf{planning phase} involves both:
  - “physical” actions: directly change the world
  - \textbf{pragmatical actions}: (indirectly) change the social state
- During the \textbf{execution phase}:
  - A physical action is directly performed by an agent
  - A pragmatical action triggers a negotiation with others
- \textbf{Negotiation} involves operations on commitments and it is driven by pragmatical rules
Reasoning about Goal and Commitments
via Social Continual Planning

Main idea:

- Interleave **planning** phases with **execution** and **negotiation** phases
- The **planning phase** involves both:
  - "physical" actions: directly change the world
  - pragmatical actions: (indirectly) change the social state
- During the **execution phase**:
  - A physical action is directly performed by an agent
  - A pragmatical action triggers a negotiation with others
- **Negotiation** involves operations on commitments and it is driven by pragmatical rules
Pragmatical Rules to Define Agent’s Strategy

- pragmatical rules **from commitments to goals** define the *strategy* of an agent (i.e., *when to trigger a planning phase*)
Pragmatical Rules to Define Agent’s Strategy

- pragmatical rules **from commitments to goals** define the *strategy* of an agent (i.e., *when to trigger a planning phase*)
- e.g.

\[
\langle G^N, C^D \rangle \quad \text{consider}(G), \text{activate}(G) \quad \text{DELIVERY}
\]

“an honest agent activates a goal $G$ when $G$ appears as a consequent of a detached commitments it responsible for”

(but are all agents honest?)
pragmatic rules **from goals to commitments** are thought of as pragmatic actions
Pragmatical Rules as Pragmatical Actions

• pragmatical rules **from goals to commitments** are thought of as pragmatical actions

\[
\frac{\langle G^A, C^N \rangle}{create(C)} \implies \text{ENTICE} \quad \iff \quad \text{ENTICE} (G, C)
\]

- :precondition $\langle G^A, C^N \rangle$
- :effect $create(C)$
Pragmatical Rules as Pragmatical Actions

- Pragmatical rules **from goals to commitments** are thought of as pragmatical actions

\[
\begin{align*}
\langle G^A, C^N \rangle & \xrightarrow{\text{create}(C)} \text{ENTICE} \\
\text{ENTICE} (G, C) & \quad :\text{precondition} \langle G^A, C^N \rangle \\
& \quad :\text{effect} \text{create}(C)
\end{align*}
\]

**ISSUE**
- how to determine over which goals and commitments these actions are defined?
Pragmatical Rules as Pragmatical Actions

- Pragmatical rules **from goals to commitments** are thought of as pragmatical actions.

\[<G^A, C^N> \xrightarrow{create(C)} \text{ENTICE} \]

\[\text{ENTICE} (G, C)\]

\[: precondtion <G^A, C^N>\]

\[: effect create(C)\]

**ISSUE**
- how to determine over which goals and commitments these actions are defined?

**SOLUTION**
- blackboard of services
Example: World-Wide Delivery Service

Problem: sending a parcel from Oklahoma City (Oklahoma) to Bertinoro (Italy)

four shipping agencies:

- **AmericanTrucks**: operates only in north America
- **EuropeanTrucks**: operates only in Europe
- **BlueVector** (flight company): blue connections
- **RedVector** (flight company): red connection
Conclusions

**Social Continual Planning:**

- practical reasoning as a form of planning
- agent’s autonomy is preserved
  - an agent can adopt local optimization strategies
  - each agent can use the planner that suits it most
- commitments support flexible planning solutions
  - help agents take advantage of the opportunities available in a given time
  - help agents find alternative solutions when something wrong happens
multiagent planning = local agents’ planning + social state

Thank you!
Example: World-Wide Delivery Service

Problem: sending a parcel from Oklahoma City (Oklahoma) to Bertinoro (Italy)

four shipping agencies:

- **AmericanTrucks**: operates only in north America
- **EuropeanTrucks**: operates only in Europe
- **BlueVector** (flight company): blue connections
- **RedVector** (flight company): red connection
Physical Actions

A subset of physical actions for the truck agencies

\textbf{load}(\textit{?t} - truck ?\textit{p} - parcel ?\textit{l} - location)

\textit{precondition}: at(?t, ?l) \land at(?p, ?l)

\textit{effect}: \lnot at(?p, ?l) \land loaded(?p, ?t)

\textbf{drive}(\textit{?t} - truck ?\textit{l1}, ?\textit{l2} - location)

\textit{precondition}: at(?t, ?l1)

\textit{effect}: \lnot at(?t, ?l1) \land at(?t, ?l2)

\textbf{deliver}(\textit{?t} - truck ?\textit{p} - parcel ?\textit{l} - location)

\textit{precondition}: at(?t, ?l) \land loaded(?p, ?t) \land dest(?p, ?l)

\textit{effect}: \lnot loaded(?p, ?t) \land at(?p, ?l) \land delivered(?p)
<table>
<thead>
<tr>
<th>agent</th>
<th>service</th>
<th>price</th>
</tr>
</thead>
<tbody>
<tr>
<td>AmericanTrucks</td>
<td>at(p, Oklahoma) ∧ delivered(p)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, New York) ∧ delivered(p)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, San Francisco) ∧ delivered(p)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>EuropeanTrucks</td>
<td>at(p, Rome) ∧ delivered(p)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, Paris) ∧ delivered(p)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, Bertinoro) ∧ delivered(p)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>BlueVector</td>
<td>at(p, Rome)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, Paris)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, New York)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>RedVector</td>
<td>at(p, Rome)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>at(p, San Francisco)</td>
<td>$?x</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Pragmatical Actions

From the point of view of AmericanTrucks (AmT):

```
entice_delivery(?a - agent ?p - parcel ?l - location)
:precondition
:effect create(C)
```

```
entice_at(?a - agent ?p - parcel ?l - location)
:effect create(C)
```

These new actions are made available to an off-the-shelf planner.
Solving the Problem

- AmericanTrucks has to deliver parcel $p_1$, initially located in Oklahoma City, to Bertinoro

```
entice_delivery(AmT, EuT, {at(p1, Bertinoro), delivery(p1)}, $?x)
```

- The planner finds a trivial plan: “ask EuropeanTrucks to deliver $p_1$”
- The execution of such a pragmatic action triggers a negotiation phase between AmericanTrucks and EuropeanTrucks
Solving the Problem

As an effect of the negotiation...

Social State

- $CC(\text{AmT, EuT, } \{\text{at}(p1, \text{Bertinoro}), \text{delivery}(p1)\}, \$100)$
- $CC(\text{EuT, AmT, } \text{at}(p1, \text{Rome}),\{\text{at}(p1, \text{Bertinoro}), \text{delivery}(p1)\})$

- AmericanTrucks has now a new goal: $\text{at}(p1, \text{Rome})$
- A new planning phase is activated
Solving the Problem

A new trivial plan is found:

\[
\text{entice}_\text{at}(\text{AmT}, \text{BlueV}, \text{at}(p1, \text{Rome}), $?x)
\]

which triggers a new negotiation phase:

Social State

\[
\text{CC}(\text{AmT}, \text{EuT}, \{\text{at}(p1, \text{Bertinoro}), \text{delivery}(p1)\}, $100)
\]

\[
\text{CC}(\text{EuT}, \text{AmT}, \text{at}(p1, \text{Rome}), \{\text{at}(p1, \text{Bertinoro}), \text{delivery}(p1)\})
\]

\[
\text{CC}(\text{AmT}, \text{BlueV}, \text{at}(p1, \text{Rome}), $500)
\]

\[
\text{CC}(\text{BlueV}, \text{AmT}, \text{at}(p1, \text{New York}), \text{at}(p1, \text{Rome}))
\]
Solving the Problem

AmericanTrucks

- load(AmTruck27, p1, OC)
- drive(AmTruck27, OC, NY)
- unload(AmTruck27, p1, OC)

CC(BlueV, AmT, at(p1, New York), at(p1, Rome)),

CONDITIONAL
Solving the Problem

AmericanTrucks

- load(AmTruck27, p1, OC)
- drive(AmTruck27, OC, NY)
- unload(AmTruck27, p1, OC)

CC(BlueV, AmT, T, at(p1, Rome))

DETACHED
Solving the Problem

**AmericanTrucks**
- load(AmTruck27, p1, OC)
- drive(AmTruck27, OC, NY)
- unload(AmTruck27, p1, OC)
- CC(BlueV, AmT, T, at(p1, Rome))
  - DETACHED

**BlueVector**
- embark(BV5, p1, NY)
- fly(BV5, NY, RM)
- disembark(BV5, p1, RM)
- CC(EuT, AmT, at(p1, Rome), {at(p1, Bertinoro), delivery(p1)})
  - CONDITIONAL
- CC( AmT, BlueV, at(p1, Rome), $500)
  - CONDITIONAL
Solving the Problem

AmericanTrucks

- load(AmTruck27, p1, OC)
- drive(AmTruck27, OC, NY)
- unload(AmTruck27, p1, OC)
- CC(BlueV, AmT, T, at(p1, Rome))

BlueVector

- embark(BV5, p1, NY)
- fly(BV5, NY, RM)
- disembark(BV5, p1, RM)
- CC(EuT, AmT, T, {at(p1, Bertinoro), delivery(p1)})
- CC(AmT, BlueV, T, $500)

SATISFIED

DETACHED
Solving the Problem

AmericanTrucks
- load(AmTruck27, p1, OC)
- drive(AmTruck27, OC, NY)
- unload(AmTruck27, p1, OC)
- CC(BlueV, AmT, T, at(p1, Rome))
  \textit{SATISFIED}
- pay(BlueV, $500)

BlueVector
- embark(BV5, p1, NY)
- fly(BV5, NY, RM)
- disembark(BV5, p1, RM)
- CC(EuT, AmT, T, \{at(p1, Bertinoro), delivery(p1)\})
  \textit{SATISFIED}
- CC(BlueV, AmT, T, $500)
  \textit{SATISFIED}

EuropeanTrucks
- load(EuTruck13, p1, RM)
- drive(EuTruck13, RM, BR)
- deliver(EuTruck13, p1, BR)
- CC(AmT, EuT, T, $100)
  \textit{DETACHED}
Solving the Problem

AmericanTrucks
- load(AmTruck27, p1, OC)
- drive(AmTruck27, OC, NY)
- unload(AmTruck27, p1, OC)
- CC(BlueV, AmT, T, at(p1, Rome))
- pay(BlueV, $500)
- pay(EuT, $100)

BlueVector
- embark(BV5, p1, NY)
- fly(BV5, NY, RM)
- disembark(BV5, p1, RM)
- CC(EuT, AmT, T, {at(p1, Bertinoro), delivery(p1)})

EuropeanTrucks
- load(EuTruck13, p1, RM)
- drive(EuTruck13, RM, BR)
- deliver(EuTruck13, p1, BR)
- CC(AmT, EuT, T, $100)
- CC(BlueV, AmT, T, $500)

SATISFIED
Reasoning about Goal and Commitments via Continual Planning

- Given an agent $x$, its configuration is $S_x : \langle B, C, G \rangle$ [Telang]:
  - $B$: set of beliefs about the world state (including beliefs about itself and others)
  - $C$: set of commitments of the form $C(x, y, s, u)$ (public)
  - $G$: set of goals of the form $G(x, p, r, q, s, f)$ (private)

- Extended agent configuration $S_x : \langle B, C, G, A_x, A_{gc}^x, R_{cg}^x \rangle$:
  - $A_x$: set of primitive actions for agent $x$ (change a portion of the world)
  - $A_{gc}^x$: set of actions corresponding to pragmatical rules from goals to commitments (change the social state)
  - $R_{cg}^x$: set of reactive rules corresponding to pragmatical rules from commitments to goals (trigger planning phases)