Making strategies more explicit (but not too much)

Jan Broersen

Information and Computer Science Department
Utrecht University, The Netherlands

Logics for Dynamics of Information and Preferences
Seminar session on Strategies
Monday October 26th, 2009

Abstract

We argue that ATL is too weak to express several important multi-agent system verification properties. In particular, many important aspects of agent interaction cannot be suitably modeled in ATL since it lacks the possibility to talk about strategies more explicitly and, in particular, reason conditionally on strategy performance. Our answer to this observation (that is also made by several other authors) will be to extend ATL with stit-reasoning capability. We extend the basic interactive reasoning objective of ATL, as expressed by the super-additivity axiom, and enable assumption-guarantee type reasoning, stit-reasoning, etc. We present a general semantics, which is not finitely axiomatizable and then focus on suitable fragments that are. Also, we compare the logic's suitability for multi-agent system verification with verification formalisms based on dynamic logic. We will give a number of arguments in favor of stit-formalisms. But, which paradigm to use ultimately depends on what kind of properties we want to verify.

Outline

Reasons for becoming more explicit about strategies

A general strategic stit semantics

Fragments with bad and good behavior

Discussion: dynamic logic versus stit

Challenges

Assumption-guarantee reasoning

- Assumption-guarantee reasoning introduced in [Conjoining specifications, Abadi and Lamport '95]
- Verification of modular systems by reasoning about behavioral properties of components modulo the given behavior of other components.
- Central question: how to conjoin specifications that have been proven correct modulo each others behavior?
- MOCHA (model checker for ATL): assume-guarantee reasoning, by putting the 'assume' part in the model description language (called 'reactive modules')
- We want to put expressivity for the conditional part in ATL itself (and not in the reactive modules). We want to do reasoning, not only model checking.

Regularity, ability and the qualification problem

- ATL obeys regularity: ability = ensuring a condition, whatever the others do
- Regularity is very strong: apparently I cannot open that door
- Problem is well-known in the area of reasoning about action and change: it is called the qualification problem [Ginsberg and Smith 1988]
- So, we have to conditionalize on the 'normal circumstances' (e.g., nobody holding the door) to specify abilities
- Work with Paolo Turrini: a coalition logic for abilities modulo the moves of other agents (leaping ahead to the discussion: with conditionalization on action as in Dynamic Logic)

Deontic conditionals

- Deontic logic is about reasoning which strategies are ‘good’ and which are ‘bad’ from a normative perspective.
- [Horty 2001], [Kooi and Tamminga 2009]: obligations conditional on moves of (other) agents.
- CTD reasoning requires we reason modulo execution of actions.
- Dynamic version Chisholm: forbidden to kill, but if you kill, you have to kill gently (Forrester’s gentle murderer)
- leaping ahead to the discussion: difficult in dynamic deontic logic, because there conditionalization is with respect to actions explicit in the modal boxes, leading, e.g., to problems with negation.
- leaping ahead to the discussion: In *stit*-type formalisms we can conditionalize using the standard material implication.

Reasoning underlying game theoretic solution concepts

The reasoning underlying game theoretic solution concepts always involves conditionalization on moves of other agents.

- Strictly dominating choice: whatever the others do (consider the other agents as in decision theory)
- Nash: simultaneous best response to best response
- Minimax: alternating best response to best response
- Rationalizability: iterated elimination dominated strategies

Outline

Summary results on strategic *stit* (the logic G.STRAT)

- Solution to Horty's problem of defining a satisfactory semantics for strategic *stit* as such.
- A fairly standard normal modal semantics for strategic *stit* (replacing an earlier attempt).
- Embedding of many well-known logics, like PL, CTL, LTL, CTL*, CL, ATL, ATL*, XSTIT, Xu's *stit*, other *stit* logics from philosophy, like Horty's.
- identification of 'bad' and 'good' behaved fragments.

Strategic *stit*

Definition (Syntax)

Well-formed formulas of the language $\mathcal{L}_{\text{G.STRAT}}$ are defined by:

$$\varphi \quad := \quad p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [A \text{ sstit}]\varphi \mid \varphi U^{ee}\varphi$$

Definition (Syntactic abbreviations)

$$\Box\varphi \equiv_{def} [\emptyset \text{ sstit}]\varphi \qquad X\varphi \equiv_{def} \bot U^{ee}\varphi$$

$$\psi U\varphi \equiv_{def} \varphi \vee (\psi \wedge (\psi U^{ee}\varphi)) \qquad \psi U_w\varphi \equiv_{def} \neg\varphi U\neg\psi$$

$$F\varphi \equiv_{def} \top U\varphi \qquad G\varphi \equiv_{def} \neg F\neg\varphi$$

Idea behind the semantics

- Picture multi-agent system behavior as a bundle of histories through static multi-agent system states
- Strategies, for agent sub-groups in the system are possible sub-bundles of histories.
- The temporal modality $\varphi U^{ee} \varphi$ has a standard interpretation on individual histories
- Introduce dynamic states, that tell you what is the 'actual' history and what are the 'actual' strategies presently 'run' by the agents (here comes in the *stit*-view)
- The agency operator $[A \text{ sstit}]\varphi$ is interpreted as a relation between histories 'jumping' to histories that 'could have been the actual history' given the agent's present strategy. If φ holds for all these histories, his strategy ensures φ.
- Historical necessity $\Box\varphi$ is like the universal path quantifier in CTL. It says that something is settled (not possible to be influenced by a present choice of some agent).

Semantic Structures

Definition (Semantic structures)

$\mathcal{F} = \langle S, H, \{sT(a) \mid a \in Ags\}, R_X, \{R_A \mid A \subseteq Ags\}\rangle$ such that:

1. S is a non-empty set of (static) states. Elements of S are denoted s, s', etc.
2. H is a non-empty set of histories. Histories are sets of states (implicitly) ordered by the next state relation R_X over dynamic states. Elements of H are denoted h, h', etc.
3. $sT(a)$ is for each $a \in Ags$ a non-empty set of strategies. Strategies are sets of histories. Strategies in $sT(a)$ are denoted α_a, β_a, etc.
4. dynamic states are tuples $\langle s, h, \alpha_a, \alpha_b, \ldots \alpha_k\rangle$, where:
   - 4.1 $s \in h$
   - 4.2 $\forall x \in Ags, h \in \alpha_x$

   The substructure $\alpha_a, \alpha_b, \ldots \alpha_k$ is called a 'strategy profile'. We use '$\vec{\alpha}$' as a concise notation.

Semantic Structures, cont.

Definition (Semantic structures, cont.)

1. R_X is a 'next time' relation over dynamic states. That is, $\langle s, h, \alpha_a, \alpha_b, \ldots \alpha_k \rangle R_X \langle s', h', \beta_a, \beta_b, \ldots \beta_k \rangle$ if and only if $h = h'$, $\forall x \in Ags, \alpha_x = \beta_x$, and R_X is serial and deterministic. For $s, t \in h$ we write $s < t$ in case we have $\langle s, h, \alpha_a, \alpha_b, \ldots \alpha_k \rangle R_X^+ \langle t, h, \alpha_a, \alpha_b, \ldots \alpha_k \rangle$, where R_X^+ denotes the transitive closure of the relation R_X.
2. The R_A are 'effectivity' equivalence classes over dynamic states such that $\langle s, h, \alpha_a, \alpha_b, \ldots \alpha_k \rangle R_A \langle s', h', \beta_a, \beta_b, \ldots \beta_k \rangle$ if and only if $s = s'$, and $\forall x \in A, \alpha_x = \beta_x$. Furthermore:
   - 2.1 $R_X \circ R_\emptyset \subseteq R_A \circ R_X$ for any A (no action constitutes a choice between histories that are undivided in next states)

Truth conditions

Definition (Truth, validity, logic)

Truth of a G.STRAT-formula φ in a dynamic state $\langle s, h, \vec{\alpha} \rangle$ of a model $\mathcal{M} = \langle S, H, \{sT(a) \mid a \in Ags\}, R_X, \{R_A \mid A \subseteq Ags\}, \pi \rangle$ is defined as (suppressing the model denotation '$\mathcal{M}$'):

$$
\begin{array}{lcl}
\langle s, h, \vec{\alpha} \rangle \models p & \Leftrightarrow & \langle s, h, \vec{\alpha} \rangle \in \pi(p) \\
\langle s, h, \vec{\alpha} \rangle \models \neg\varphi & \Leftrightarrow & \text{not } \langle s, h, \vec{\alpha} \rangle \models \varphi \\
\langle s, h, \vec{\alpha} \rangle \models \varphi \wedge \psi & \Leftrightarrow & \langle s, h, \vec{\alpha} \rangle \models \varphi \text{ and} \\
 & & \langle s, h, \vec{\alpha} \rangle \models \psi \\
\langle s, h, \vec{\alpha} \rangle \models [A \text{ sstit}]\varphi & \Leftrightarrow & \text{if } \langle s, h, \vec{\alpha} \rangle R_A \langle s, h', \vec{\beta} \rangle \\
 & & \text{then } \langle s, h', \vec{\beta} \rangle \models \varphi \\
\langle s, h, \vec{\alpha} \rangle \models \psi U^{ee} \varphi & \Leftrightarrow & \exists t \text{ with } s, t \in h \text{ and } s < t, \\
 & & \text{such that} \\
 & & (1)\ \langle t, h, \vec{\alpha} \rangle \models \varphi \text{ and} \\
 & & (2)\ \forall r \text{ with } s < r < t \text{ we} \\
 & & \text{have } \langle r, h, \vec{\alpha} \rangle \models \psi
\end{array}
$$

Outline

A product logic as a fragment

Definition (the sub-logic AIV.STRAT^{a-temp})

$AIV.STRAT^{a-temp}$ is the logic such that AIV.STRAT^{a-temp} $\subset$ G.STRAT that results from restricting the syntax by (1) dropping the until operator, and (2) restricting the set of operators "$[A$ sstit$]\varphi$ with $A \subseteq Ags$", to the set "$[P$ sstit$]\varphi$ with $P = Ags \setminus a$ and $a \in Ags$". The groups P are called ‘anti-individuals’ by [Herzig and Schwarzentruber AIML 2008].

Theorem

For $|Ags| \geq 3$, the satisfiability problem of the fragment AIV.STRAT^{a-temp} is undecidable, and there is no standard finite Hilbert-style axiomatization.

A well-known and well-behaved fragment

Theorem

The logic ATL is the fragment of the logic G.STRAT determined by the definitions

$\langle\langle A\rangle\rangle X\varphi \equiv_{def} \Diamond[A \ sstit]X\varphi$,

$\langle\langle A\rangle\rangle G\varphi \equiv_{def} \Diamond[A \ sstit]G\varphi$,

$\langle\langle A\rangle\rangle(\varphi U\psi) \equiv_{def} \Diamond[A \ sstit](\varphi U\psi)$

A new strategic *stit* fragment

Definition (the sub-logic CTL.STIT)

CTL.STIT is the logic such that CTL.STIT $\subset$ G.STRAT that results from restricting the syntax by having operators $[A \text{ sstit}]\varphi$ only appear in the combinations:
$[A \text{ sstit}]X\varphi$,
$[A \text{ sstit}]G\varphi$, and
$[A \text{ sstit}](\varphi U \psi)$.

- Agency and time are coupled.
- theorem: XSTIT and ATL are sub-fragments
- Turns the product of the general semantics into something that resembles a ‘flow product’ [Gabbay, Shehtman 1999, unpublished]
- conjecture: the fragment CTL.STIT is well-behaved

Original design strategy for the one-shot fragment XSTIT

- We build up the semantics and the syntax ensuring we stay within the Sahlqvist class, thereby ensuring completeness (*).
- Effects in next states (without which we would not have (*))
- Formal comparison with *stit* formalisms from philosophy is not straightforward:
  - XSTIT models are a superset of standard *stit* models
  - The XSTIT language is neither a sub- or superset of the standard temporal *stit* language of, e.g. Horty.
- XSTIT models are easily viewed as two dimensional normal simulations of models of Alternating Time Temporal Logic and Coalition Logic ($s \in E(s')$ iff $s' \; R_{\Box} \circ R_A \; s$). CL is embedded by $[A]\varphi := \Diamond[A \text{ xstit}]\varphi$.
- We stay within normal multi-modal / two-dimensional modal logic.

Syntax XSTIT

The XSTIT syntax:

$$\varphi \quad := \quad p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Box\varphi \mid [a \text{ xstit}]\varphi \mid X\varphi$$

‘Next’ is not defined as an abbreviation...

Frames XSTIT

A frame is a tuple $\mathcal{F} = \langle S, H, R_\Box, \{R_A \mid A \subseteq Ags\}, R_X\rangle$ such that:

- S is a non-empty set of (static) states. Elements of S are denoted s, s', etc.
- H is a non-empty set of histories. Histories are sets of states ordered by a next state relation. Elements of H are denoted h, h', etc.
- R_X is a next state relation, obeying seriality and determinism.
- dynamic states are tuples $\langle s, h\rangle$, with $s \in S$ and $h \in H$ and $s \in h$.
- $R_\Box$ is a 'historical necessity' relation over dynamic states such that $\langle s, h\rangle R_\Box \langle s', h'\rangle$ if and only if $s = s'$
- The R_A are 'effectivity' relations over dynamic states obeying appropriate Sahlqvist first-order conditions.

Semantics of XSTIT

Validity $\mathcal{M}, \langle s, h \rangle \models \varphi$, of a formula φ in a dynamic state $\langle s, h \rangle$ of a model $\mathcal{M} = \langle S, H, R_{\Box}, \{R_A \mid A \subseteq Ags\}, R_X, \pi \rangle$ is defined as:

$$
\begin{array}{lll}
\mathcal{M}, \langle s, h \rangle \models p & \Leftrightarrow & \langle s, h \rangle \in \pi(p) \\
\mathcal{M}, \langle s, h \rangle \models \neg\varphi & \Leftrightarrow & \text{not } \mathcal{M}, \langle s, h \rangle \models \varphi \\
\mathcal{M}, \langle s, h \rangle \models \varphi \wedge \psi & \Leftrightarrow & \mathcal{M}, \langle s, h \rangle \models \varphi \text{ and } \mathcal{M}, \langle s, h \rangle \models \psi \\
\mathcal{M}, \langle s, h \rangle \models \Box\varphi & \Leftrightarrow & \langle s, h \rangle R_{\Box} \langle s', h' \rangle \text{ implies } \mathcal{M}, \langle s', h' \rangle \models \varphi \\
\mathcal{M}, \langle s, h \rangle \models X\varphi & \Leftrightarrow & \langle s, h \rangle R_X \langle s', h' \rangle \text{ implies } \mathcal{M}, \langle s', h' \rangle \models \varphi \\
\mathcal{M}, \langle s, h \rangle \models [A \text{ xstit}]\varphi & \Leftrightarrow & \langle s, h \rangle R_A \langle s', h' \rangle \text{ implies } \mathcal{M}, \langle s', h' \rangle \models \varphi
\end{array}
$$

Satisfiability, validity on a frame and general validity are defined as usual.

Axioms of XSTIT

	S5 for $\Box$
	KD for each $[A \text{ xstit}]$
(Det)	$\neg X \neg \varphi \rightarrow X\varphi$
(C-Mon)	$[A \text{ xstit}]\varphi \rightarrow [A \cup B \text{ xstit}]\varphi$
($\emptyset \Rightarrow$ Sett)	$[\emptyset \text{ xstit}]\varphi \rightarrow \Box X\varphi$
(X-Eff)	$\Box X\varphi \rightarrow [A \text{ xstit}]\varphi$
(NCUH)	$[A \text{ xstit}]\varphi \rightarrow X\Box\varphi$
(Indep-G)	$\Diamond[A \text{ xstit}]\varphi \wedge \Diamond[B \text{ xstit}]\psi \rightarrow \Diamond([A \text{ xstit}]\varphi \wedge [B \text{ xstit}]\psi)$ for $A \cap B = \emptyset$

Outline

Ontological commitments of action logics

Logic \ Action	Names	Execution	Reach	Agency	Eff. Des.	Reali.
Modal Action Logic	Yes	conditional	one shot	none	open	next
Situation Calculus	Yes	conditional	one shot	none	open	next
Dynamic Logic	Yes	conditional	strategic	none	open	next
FO Dynamic Logic	No	conditional	strategic	none	closed	next
Coalition Logic	No	conditional	one shot	group	open	abs.
sub-game operator	No	conditional	one shot	group	closed	abs.
ATL	No	conditional	strategic	group	open	next
BIAT (Segerberg)	No	conditional	strategic	single	closed	next
PAL / DEL	No	cond. + det.	one shot	none	closed	next
Belnap *stit*	No	actual + cond.	one shot	multi	open	imm.
Group *stit*	No	actual + cond.	one shot	group	open	imm.
XSTIT	No	actual + cond.	one shot	group	open	next
Strategic STIT	No	actual + cond.	strategic	group	open	imm.

Comparison using fixed point semantics

- Central dynamic logic modality: $[a]\varphi$ (action name in the box)
- Use fixed points to increase expressivity: $\nu Z.\, \varphi \wedge [a][b]Z$
- In dynamic logic: $[(a; b)^*]\varphi$

Now the same information in *stit*

- Comparable central *xstit*-logic modality: $\Diamond[A \text{ xstit}](a \wedge \varphi)$ (agent name in the box)
- Use fixed points to increase expressivity: $\nu Z.\, \varphi \wedge \Diamond[A \text{ xstit}](a \wedge \Diamond[A \text{ xstit}](b \wedge Z))$
- In CTL.STIT: $\varphi \wedge \Diamond[A \text{ sstit}]G(\varphi \rightarrow \Diamond[A \text{ sstit}]X(a \wedge \Diamond[A \text{ sstit}]X(b \wedge \varphi)))$

Mirrors the main difference: *stit* actions are identified with effects, dynamic logic actions with the names inside boxes. However, there seems to be no essential difference in expressivity.

Additional advantage of *stit*: after removing the $\Diamond$ from the central modality we can reason directly about action using standard logic connectives.

More arguments

- The names of human agent actions usually refer directly to their effect.
- Planning community: HTN planning = planning based on abstract actions
- Giving action names too ‘early’ violates the planning principle of ‘least commitment’ (see the theory on partial-order planning)
- Some philosophers have argued actions are ‘derived’ entities. I agree.
- All depends on whether we approach the phenomenon of action from the bottom (starting with action objects with a name) or from the top (starting with abstract effects).

Reasoning about strategies

Human agents explain and describe strategies using finite sets of condition action pairs, like the following:

$$\{\text{ if } p_1 \text{ do } q_1,\ \text{ if } p_2 \text{ do } q_2, \ldots,\ \text{ if } p_n \text{ do } q_n\}$$

Now we can express in the logic that a group A performs this strategy as:

$$\begin{array}{l}
[A \text{ sstit}]G(\\
(p_1 \rightarrow [A \text{ sstit}]Xq_1)\wedge \\
(p_2 \rightarrow [A \text{ sstit}]Xq_2)\wedge \\
\vdots \\
(p_n \rightarrow [A \text{ sstit}]Xq_n))
\end{array}$$

Agent programs and strategies

Claim: if we abstract away from the detail, agent programs are strategy descriptions in terms of condition-action pairs.

Typical agent program rule: $\gamma \leftarrow \beta \mid \alpha$

- β a propositional logic formula to be evaluated against an agent's *belief base*,
- γ a propositional formula to be evaluated against an agent's declarative *goal base*, and
- α an action term allowing special programming constructs.

Approach: in stead of using a dynamic logic for the α-part, use strategic *stit* for agent program verification

Advantages of not having the action names inside the boxes

Action operator \ Logic	*stit*-logic	dynamic logic
Concurrency	logical conjunction	concurrency theory
Conditionalization on action	material implication	modal univ.-quantification
Action negation	logical negation	problematic

- Note that a semantics in terms of dynamic states is required to be able to interpret the action operators as standard logical connectives.
- Reflects directly that *stit* is a logic of action, while dynamic logic is a logic of programming. And that is also what dynamic logic was designed for.

More advantages of not having the action names inside the boxes

We can study agent interaction properties as standard normal modal logic axioms and frame properties. Examples:

Coalition monotonicity: $\Diamond[A \text{ sstit}]\varphi \rightarrow \Diamond[B \text{ sstit}]\varphi$ for $A \subseteq B$.

This says that if a group A can strategically see to something, any supergroup can also strategically see to that same something.

The point here is that *stit*, as a framework, allows us to study such properties. And if we do not want some of them, we can adapt our semantics accordingly.

Regularity

Regularity: $\Diamond[A \text{ sstit}]\varphi \rightarrow \Box\langle\overline{A} \text{ sstit}\rangle\varphi$.

This says that if a group A can strategically see to it that φ, the complementary group of agents necessarily have to allow φ as a possibility, i.e., they cannot see to it that $\neg\varphi$.

maximality

Ags-**maximality**: $\Box\langle\emptyset \text{ sstit}\rangle\varphi \rightarrow \Diamond[\mathit{Ags} \text{ sstit}]\varphi$.

This says that if the empty set of agents cannot but allow for a possible outcome obeying φ, the whole group of agents *Ags* can ensure φ.

So, all things that are possible outcomes as such can actually be guaranteed by a choice of the grand coalition.

Note that general maximality $\Box\langle\overline{A} \text{ sstit}\rangle\varphi \rightarrow \Diamond[A \text{ sstit}]\varphi$ is not necessarily a desirable property. The present logic also does not satisfy it.

Independence of agency

Independence of agency:
$\Diamond[A \text{ sstit}]\varphi \wedge \Diamond[B \text{ sstit}]\psi \rightarrow \Diamond([A \text{ sstit}]\varphi \wedge [B \text{ sstit}]\psi)$ for $A \cap B = \emptyset$.

This says that agents cannot deprive other agents of choices of the same moment, i.e., choices possibly taken concurrently by other agents.

The properties are not independent. E.g., regularity follows from independence of agency together with coalition monotonicity.

Also the properties are not necessarily desirable for any application of the logic.

Strong super-additivity

Inspired by assumption-guarantee reasoning and the ideas on conjoining specifications, we may investigate:

$$\textit{(SSA)} \quad ([B \text{ sstit}]X\psi \to \Diamond[A \text{ sstit}]X\varphi \wedge [A \text{ sstit}]X\varphi \to \Diamond[B \text{ sstit}]X\psi) \to \Diamond([A \text{ sstit}]X\varphi \wedge [B \text{ sstit}]X\psi) \text{ for } A \cap B = \emptyset$$

Autonomy

One of the most central properties of multi-agent systems, as a modal interaction axiom:

Autonomy: $[A \text{ sstit}]X[B \text{ sstit}]\varphi \leftrightarrow [A \text{ sstit}]X\Box\varphi$ for $A \cap B = \emptyset$.

This says there is just only one way in which a group A can ensure that next a disjoint group B sees to it that φ: by seeing to it that next φ is settled.

In other words, group A cannot see to it that B makes a *deliberate* choice for something; the only way he can directly influence group B's choice is by ensuring that some property holds for *all* his choices.

Note that 'autonomy' as defined here is different from 'independence of agency'.

Yet more advantages of not having the action names inside the boxes

Modeling knowingly doing and intentionally doing.

- These are important notions is philosophy and in law (murder versus manslaughter, strict liability versus recklessness, and so on.)
- In *stit* we can directly apply standard possible world semantics for epistemic modalities and motivational modalities to dynamic states.
- I do not see how these could be studied in dynamic logic

Outline

Reasons for becoming more explicit about strategies

A general strategic stit semantics

Fragments with bad and good behavior

Discussion: dynamic logic versus stit

Challenges

Open questions / further research

- How do we incorporate epistemic modalities and modalities for intention and preference?
- What does it mean to do something knowingly, as a *group*? Is this type knowledge about dynamics of the ‘common’ type or of the ‘distributed’ type?
- Can we characterize (aspects of) Dynamic Epistemic Logic within *stit*? ‘Only’-stit (like ‘only knowing’, etc)?
- How do we make the link with agency and agent programming more concrete? Representation theorems?

Thanks

Thanks!