SITUATION CALCULUS WITH ACTIONS AND OTHER EVENTS

John McCarthy
Computer Science Department
Stanford University
jmc@cs.stanford.edu
http://www-formal.stanford.edu/jmc/
2005 Nov 17

A slogan for AI: Whatever a person can do, he should
able to make a computer do for him.

Almost all of my papers are on the above-mentioned w
page.

This lecture proposes Events Primary Sequential situat
calculus, EPS sitcalc for short.

SITUATION CALCULUS

- Proposed 1963 for formalizing effects of actions.
- Improved 2002 to include occurrence axioms.
- http://www-formal.stanford.edu/jmc/sitcalc.html

ACTIONS ARE EVENTS

• In EPS situation calculus, events are primary and acti
by actors are a kind of event. EPS is sequential.

• The logic is first order logic without causal operat
Second order formulas are used for circumscription.

• An *event* e has some *effect axioms* formalizing

$$\text{<preconditions>} \rightarrow Holds(fluent, Result(e,s))$$

• An *internal event* e also has an *occurrence axiom*

$$\text{<preconditions>} \rightarrow Occurs(e,s),$$

and an axiom for the *next* situation

$$Occurs(e,s) \rightarrow Next(s) = Result(e,s).$$

- Some phenomena previously axiomatized with *dom* *constraints* are often more accurately and convenie axiomatized by using *internal events*. When both ve are blocked the room becomes stuffy.

- We minimize change a situation at a time. A v becoming blocked and the room becoming stuffy oc in different situations.

- When the theory is used for projection of the cor quences of sequences of events, the nonmonotonic r soning is done one situation at a time.

- When an event e is not governed by an occurre axiom, we have branching time, i.e. non-determini When $Occurs(e, s)$ holds, we have linear time.

- Processes that don't settle down cannot be trea
with state constraints. The buzzer is an example,
the stuffy room elaborated to buzz is another.

$Result^*, Next, Next^*$

- $Result^*(e, s)$ gives the situation resulting from e ter the events formalized to occur have happened. example, if vent1 is closed, then $Result^*(Block2, s)$ $Result(Getstuffy, Result(Block2, s))$.

- When what occurs in a situation is determined, ther a next situation satisfying

$$Occurs(e, s) \rightarrow Next(s) = Result(e, s).$$

- When other actions are asserted to occur $Next^*(s$ sometimes wanted. $Result^*$ and $Next^*$ are undefined w the system doesn't settle down as in the buzzer or buzzing stuffy room.

A BUZZER—1

A simple buzzer consists of a relay operating a br
switch. When the relay isn't energized, current can f
through the switch operating the relay. When the
lay operates it opens the switch, cutting off the curr
through the relay. The system then oscillates, i.e. buzz

The buzzer has only internal events—four of them. Op
ating and releasing the relay, and operating and releas
the switch.

A BUZZER—2

Effect axioms:

$$Holds(On(R), Result(Onn(R), s))$$
$$\neg Holds(On(R), Result(\mathit{Offf}(R), s))$$
$$Holds(On(Sw), Result(Onn(Sw), s)$$
$$\neg Holds(On(Sw), Result(\mathit{Offf}(Sw), s)).$$

Occurrence axioms:

$$\neg Holds(On(Sw), s) \wedge Holds(On(R), s)$$
$$\rightarrow Occurs(\mathit{Offf}(R), s)$$
$$Holds(On(Sw), s) \wedge \neg Holds(On(R), s)$$
$$\rightarrow Occurs(Onn(R), s))$$
$$Holds(On(R), s) \wedge Holds(On(Sw), s)$$
$$\rightarrow Occurs(\mathit{Offf}(Sw), s)$$
$$\neg Holds(On(R), s) \wedge \neg Holds(On(Sw), s)$$
$$\rightarrow Occurs(Onn(Sw), s)$$

THE STUFFY ROOM

A room has two vents, vent1 and vent2. The vents
be opened or closed. When both vents are closed,
room is, or becomes stuffy. Matt Ginsberg proposed
scenario in 1988 to show that simply minimizing cha
gives an unintended model, namely a model in which w
one vent is closed, the other opens, which avoids chang
the stuffiness of the room.

We formalize this using the internal events of the ro
becoming stuffy or unstuffy.

We then elaborate the scenario to express that when
room is stuffy, Pat then opens a vent.

THE STUFFY ROOM—simple

Effect axioms:

$$Holds(Blocked1, Result(Block1, s))$$
$$Holds(Blocked2, Result(Block2, s))$$
$$\neg Holds(Blocked1, Result(Unblock1, s))$$
$$\neg Holds(Blocked2, Result(Unblock2, s))$$
$$Holds(Stuffy, Result(Getstuffy, s))$$
$$\neg Holds(Stuffy, Result(Ungetstuffy, s))$$

Occurrence axioms:

$$Holds(Blocked1, s) \wedge Holds(Blocked2, s) \wedge \neg Holds(Stuffy, s) \rightarrow Occurs(Getstuffy, s)$$

and

$$(\neg Holds(Blocked1, s) \vee \neg Holds(Blocked2, s)) \wedge Holds(Stuffy, s) \rightarrow Occurs(Ungetstuffy, s)$$

ELABORATING THE STUFFY ROOM

The first elaboration says that when Pat finds the ro
stuffy he unblocks vent2. We have

$$Holds(Stuffy, s) \rightarrow Occurs(Does(Pat, Unblock2), s).$$

A second elaboration in which Mike finds the room c
when there is an unblocked vent and blocks vent2 is
pressed by

$$Holds(Unstuffy, s) \rightarrow Occurs(Does(Mike, Block2), s)$$

(

With both elaborations, we get an oscillation; Pat
blocks vent2 and Mike blocks it again.

NONMONOTONIC REASONING IN SITCALC

• Projection is the easy case of nonmonotonic reason about the effects of events.

• When we project, we can circumscribe in each sit tion successively. It gives the same results as Shoha chronological minimization but is much simpler tech cally. It doesn't suit the stolen car scenario in whic fact about the future is given.

• We minimize the predicates $Occurs$, $Prevents$, $Chan$ etc. Strictly speaking, we circumscribe $(\lambda e)Occurs($ and $(\lambda f\ e)Prevents(f, e, s)$, $(\lambda e\ f)Changes(e, f, s)$.

NONMONOTONIC REASONING—2

$$
\begin{aligned}
&Foo' \leq_s Foo \equiv (\forall vars)(Foo'(vars, s) \rightarrow Foo(vars, s)) \\
&(Foo' <_s Foo) \equiv (Foo' \leq_s Foo) \wedge \neg(Foo' =_s Foo), \\
&Foo' =_s Foo \equiv (\forall vars)(Foo'(vars) \equiv Foo(vars, s)),
\end{aligned}
$$

(

where $vars$ stands for a list of the entities varied as
is minimized. Then the circumscription of $Foo(var$
takes the form

$$
\begin{aligned}
&Axiom(Foo, vars) \wedge (\forall foo'\ vars')(Axiom(foo', vars') \\
&\qquad \rightarrow \neg(foo' <_s Foo)).
\end{aligned}
$$

(

This spells out to

$$
\begin{aligned}
&Axiom(Foo, vars) \wedge (\forall foo'\ vars')\\
&(Axiom(foo', vars') \wedge ((\forall vars)(foo'(vars, s)\\
&\qquad\qquad \rightarrow Foo(vars, s))\\
&\rightarrow (\forall vars)(Foo(vars, s) \equiv foo'(vars, s)))).
\end{aligned}
$$

(

Call this formula $Circ(Axiom; Foo; vars; s)$.

The general frame axioms are

$$\neg Changes(e, p, s) \rightarrow (Holds(p, Result(e, s)) \equiv Holds(p,$$

(

for propositional fluents and

$$\neg Changes(e, f, s) \rightarrow Value(f, Result(e, s)) = Value(f,$$

(

for general fluents.

NARRATIVES

• A narrative is a set of situations, event, and asserti
about situations and maybe assertions about events.

• A *simple narrative* consists of two sequences $(S_1, S_2,$
and $(E_1, E_2, \ldots)$, where $S_{i+1} = Result(E_i, S_i)$ for each

• Unfortunately, real narratives, whether historical or
tional, are rarely if ever simple.

SOME PHILOSOPHY

• Assume a deterministic world—if you like with stoch tic processes and quantum processes. That doesn't g free will.

• Some entities, including people and chess progra make choices.

• Making a choice involves considering the consequen of alternative actions, e.g. using a non-determinist the like situation calculus. This is minimal free will.

• Thus deterministic entities use non-deterministic th ries.

• Do the philosophy as you like, but this is how AI has be done.

FREE WILL IN A DETERMINIST WORLD

• We can make our theory of a process more determi by adding occurrence axioms. We can do it if we le more or adopt rules for deciding on actions.

• Human free will may consist of using a non-determi theory to decide deterministically on an action.

Here's a minimal example of using a non-determinist t ory within a determinist rule.

$$
\begin{aligned}
&Occurs(Does(John,\\
&\qquad \textbf{if } \text{Prefers}(John, Result(Does(John, a1), s),\\
&\qquad\qquad\qquad Result(Does(John, a2), s))\\
&\qquad \textbf{then } a1\\
&\qquad \textbf{else } a2\\
&\qquad\quad), s).
\end{aligned}
\qquad (
$$

• Here Prefers($John, s1, s2$) is to be understood as ass ing that John prefers situation $s1$ to $s2$.

• Do animals, even apes, make decisions based on co paring anticipated consequences? If not, can apes trained to do it? Chess programs do. According Dan Dennett, some recent experiments suggest that a sometimes consider the consequences of alternate tions.

• We envisage an extended theory of free will that treat whether an action was done freely and whethe merits blame or praise.

CONCLUSIONS AND REMARKS

• This formalism is preliminary. It needs to be elabora
to allow concurrent and continuous events.

• Sequential processes, as treated in EPS, are worth se
rate formalization, because most common sense narra
and planning fit within the sequential case.

• The eventual formalism must permit elaborating a
quential theory by adding a few or many concurren
continuous processes. On the other hand, specializat
to the sequential case also needs to be a simple operat
on a theory allowing concurrent events.

• For the future: It would be more Newton-like to
that a proccess continues until something interrupts

OTHER WORK

Events that are not actions have been previously used–
least by Fangzhen Lin, Sheila McIlraith, and Javier Pi

Occurrence axioms are even more important in the tre
ment of concurrent events in situation calculus—to
the subject of another article.
http://www-formal.stanford.edu/jmc/freewill2.html a
these ideas to formalizing simple deterministic free w

This work benefited from discussions with Eyal A
Tom Costello, Ron Fadel, Hector Levesque, Vladimir L
chitz, Fangzhen Lin, Sheila McIlraith, Leora Morgenst
Aarati Parmar, Raymond Reiter, and Tran Son and
comments of three anonymous referees.

This research was partly supported by SRI Subcontr
No. 34-000144 under SPAWAR Prime Contract No. N
00-C-8018.