Argumentation and Temporal Persistence

E. Hadjisoteriou A. Kakas
Email: csp7he2@cs.ucy.ac.cy, antonis@cs.ucy.ac.cy

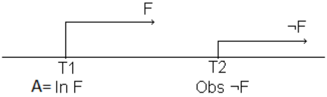
University of Cyprus, Dept. of Computer Science,
Kallipoleos 75, 1678 Nicosia, Cyprus
http://www.ucy.ac.cy

September 2011

Outline

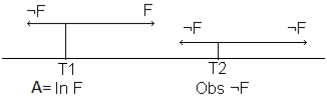
Motivation

- Understand Temporal Persistence via Argumentation
  - Study this in the specific content of Language $\mathcal{E}$
  - Our analysis is applicable more widely
- Link to Frame Problem and Qualification Problem
- Builds on earlier work: Language $\mathcal{E}$[2] and Argumentation[1]
- Motivation
  - Not all domains are consistent
- Extension: Introduce new arguments for
  1. backwards persistence
  2. persistence from observations



Motivation

- Understand Temporal Persistence via Argumentation
  - Study this in the specific content of Language $\mathcal{E}$
  - Our analysis is applicable more widely
- Link to Frame Problem and Qualification Problem
- Builds on earlier work: Language $\mathcal{E}$[2] and Argumentation[1]
- Motivation
  - Not all domains are consistent
- Extension: Introduce new arguments for
  1. backwards persistence
  2. persistence from observations



Outline

Language $\mathcal{E}$

- Action Language
- Propositions:
  - c-propositions (A initiates/terminates F when C)
  - h-propositions (A happens at T)
  - t-propositions (L holds at T)
- Models are truth assignments to Fluents that satisfy:
  - Persistence: Truth Values of Fluents remains the same in time intervals where no relevant action occurs
  - A fluent F changes its truth value only via occurrence of an initiation or a termination point

Language $\mathcal{E}$

- Action Language
- Propositions:
  - c-propositions (A initiates/terminates F when C)
  - h-propositions (A happens at T)
  - t-propositions (L holds at T)
- Models are truth assignments to Fluents that satisfy:
  - Persistence: Truth Values of Fluents remains the same in time intervals where no relevant action occurs
  - A fluent F changes its truth value only via occurrence of an initiation or a termination point

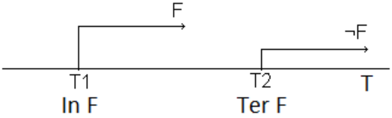
Language $\mathcal{E}$

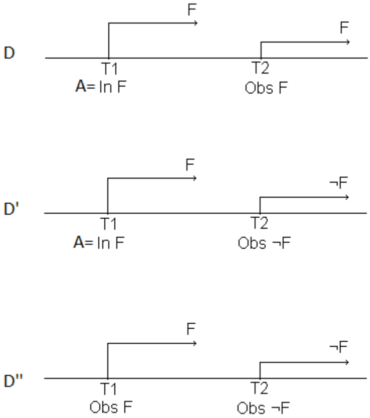
- Action Language
- Propositions:
  - *c*-propositions (A initiates/terminates F when C)
  - *h*-propositions (A happens at T)
  - *t*-propositions (L holds at T)
- Models are truth assignments to Fluents that satisfy:
  - Persistence: Truth Values of Fluents remains the same in time intervals where no relevant action occurs
  - A fluent F changes its truth value only via occurrence of an initiation or a termination point

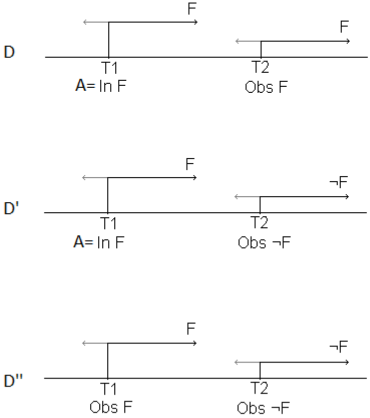
Language $\mathcal{E}$

- Action Language
- Propositions:
  - *c*-propositions (A initiates/terminates F when C)
  - *h*-propositions (A happens at T)
  - *t*-propositions (L holds at T)
- Models are truth assignments to Fluents that satisfy:
  - Persistence: Truth Values of Fluents remains the same in time intervals where no relevant action occurs
  - A fluent F changes its truth value only via occurrence of an initiation or a termination point

- Priorities
  - $PG[F, T; T1] < NG[F, T; T2]$, for all $T1 < T2$
  - ...




F
F
D
T1
A= In F
T2
Obs F
F
¬F
D'
T1
A= In F
T2
Obs ¬F
F
¬F
D''
T1
Obs F
T2
Obs ¬F


D
F
F
T1
A= In F
T2
Obs F
D'
F
¬F
T1
A= In F
T2
Obs ¬F
D''
F
¬F
T1
Obs F
T2
Obs ¬F

Argumentation Basics

- Argumentation Framework $< Arg, A >$
  - Arg is a set of Arguments
  - A is an Attacking Relation on Arg
- Admissibility: Set of Arguments such that:
  - Non self attacking
  - Attacks all its attacks
- Models: Maximal admissible sets

Outline

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
    - Persistence
    - Local Generation Arguments
    - Local Observation Arguments:
    - Assumption at 0
- $< \rightarrow$ priority relation

For all time points t_1, t_2 and t such that $t_1 < t < t_2$,

$HoldsAt(f, t_2) \leftarrow HoldsAt(f, t)$ $\quad PFP[f, t_2; t]$

$HoldsAt(f, t_1) \leftarrow HoldsAt(f, t)$ $\quad PBP[f, t_1; t]$

$\neg HoldsAt(f, t_2) \leftarrow \neg HoldsAt(f, t)$ $\quad NFP[f, t_2; t]$

$\neg HoldsAt(f, t_1) \leftarrow \neg HoldsAt(f, t)$ $\quad NBP[f, t_1; t]$

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $< \rightarrow$ priority relation

$$
\begin{array}{ll}
HoldsAt(f, t+1) \leftarrow Initiation(f, t) & PG_F[f, t] \\
\neg HoldsAt(f, t) \leftarrow Initiation(f, t) & PG_B[f, t] \\
\neg HoldsAt(f, t+1) \leftarrow Termination(f, t) & NG_F[f, t] \\
HoldsAt(f, t) \leftarrow Termination(f, t) & NG_B[f, t]
\end{array}
$$

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $< \rightarrow$ priority relation

$$HoldsAt(f,t) \leftarrow Observation(f,t) \qquad PO[f,t]$$

$$\neg HoldsAt(f,t) \leftarrow Observation(\neg f,t) \qquad NO[f,t]$$

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $< \rightarrow$ priority relation

$HoldsAt(f, 0)$ $\quad PA[f, 0]$

$\neg HoldsAt(f, 0)$ $\quad NA[f, 0]$

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $< \rightarrow$ priority relation

If $t_1 < t_2$
$PFP[f, t^*; t_1] < NFP[f, t^*; t_2]$, $NFP[f, t^*; t_1] < PFP[f, t^*; t_2]$,
$PBP[f, t^*; t_2] < NBP[f, t^*; t_1]$, $NBP[f, t^*; t_2] < PBP[f, t^*; t_1]$,
$NFP[f, t_2; t_1] < PO[f, t_2]$, $PFP[f, t_2; t_1] < NO[f, t_2]$,
$NBP[f, t_1; t_2] < PO[f, t_1]$ and $PBP[f, t_1; t_2] < NO[f, t_1]$.

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $<\rightarrow$ priority relation

At 0
$PA[f, 0] < NO[f, 0]$ and $NA[f, 0] < PO[f, 0]$

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $< \rightarrow$ priority relation

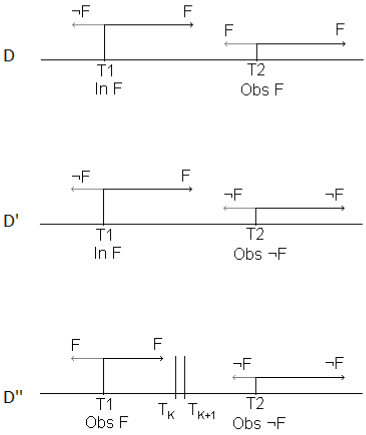
At t
$PG_B[f,t] < PO[f,t]$ and $NG_B[f,t] < NO[f,t]$

Argumentation Program $(B(D), A, <)$

- $B(D) \rightarrow$ background theory
- $A \rightarrow$ argument rules
  - Persistence
  - Local Generation Arguments
  - Local Observation Arguments:
  - Assumption at 0
- $< \rightarrow$ priority relation

At $t+1$
$PG_F[f,t] < NO[f,t+1]$ and $NG_F[f,t] < PO[f,t+1]$

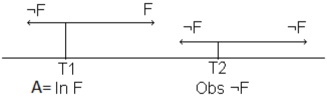
Note that there are no priorities between conflicting forward and backward persistence


D
¬F
F
T1
In F
F
F
T2
Obs F
D'
¬F
F
T1
In F
¬F
¬F
T2
Obs ¬F
D''
F
F
T1
Obs F
T_K
T_K+1
¬F
¬F
T2
Obs ¬F

Outline

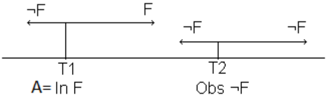
Qualification Problem[3]

- Domains have meaning even when we observe an unexpected failure of an action
- No conclusion between T_1 and T_2
  - Arguments for F and $\neg F$



Qualification Problem[3]

- Domains have meaning even when we observe an unexpected failure of an action
- No conclusion between T_1 and T_2
  - Arguments for F and $\neg F$



Outline

Results

- Recover and also extend Language $\mathcal{E}$
- The extended semantics can be explained by Language $\mathcal{E}$
- Fully recover Language $\mathcal{E}$

Forward persistence arguments = backwards persistence arguments

Theorem

Let D be a language $\mathcal{E}$ domain description and a countable number of h-propositions. Then:

- *For every language $\mathcal{E}$ model, M, of D there exists an admissible extension, E, of the corresponding argumentation program $\Delta \equiv (B(D), A, <)$ such that E corresponds to M, i.e. $E \models$ holds-at(f, T) if and only if $M(f, T) = true$ and $E \models \neg$holds-at(f, T) if and only if $M(f, T) = false$.*
- *There exists a complete admissible extension D of the corresponding argumentation program $\Delta \equiv (B(D), A, <)$.*

Results

- Recover and also extend Language $\mathcal{E}$
- The extended semantics can be explained by Language $\mathcal{E}$
- Fully recover Language $\mathcal{E}$

Add a generation point

Theorem

Let D be a domain description. For every maximally admissible extension E there exist a domain D' obtained from D by adding new events such that there exist a language $\mathcal{E}$ model, M, of D' that corresponds to E (i.e. $E \models holds\text{-}at(f, t)$ if and only if $M \models holds\text{-}at(f, t)$).

Results

- Recover and also extend Language $\mathcal{E}$
- The extended semantics can be explained by Language $\mathcal{E}$
- Fully recover Language $\mathcal{E}$

Assign higher priority to conflicting forward persistence arguments over backwards persistence arguments

Theorem

Add the following priorities rules when $t_1 < t_2$:
$PFP[f, t; t_1] > NBP[f, t; t_2]$ *and* $NFP[f, t; t_1] > PBP[f, t; t_2]$.
Then, every maximally admissible extension E, for any domain D corresponds to a model M of the language $\mathcal{E}$, of D.

Outline

Conclusion

- The extension of Language $\mathcal{E}$ enable us to:
  - recover and also extend Language $\mathcal{E}$
  - give a semantic meaning to domains that cannot be interpreted in Language $\mathcal{E}$
- Closer to original Event Calculus
- The formalization does not depend on Language $\mathcal{E}$
- Based on "Natural" Properties
- Advance Qualification Problem

Conclusion

- The extension of Language $\mathcal{E}$ enable us to:
  - recover and also extend Language $\mathcal{E}$
  - give a semantic meaning to domains that cannot be interpreted in Language $\mathcal{E}$
- Closer to original Event Calculus
- The formalization does not depend on Language $\mathcal{E}$
- Based on "Natural" Properties
- Advance Qualification Problem

Conclusion

- The extension of Language $\mathcal{E}$ enable us to:
  - recover and also extend Language $\mathcal{E}$
  - give a semantic meaning to domains that cannot be interpreted in Language $\mathcal{E}$
- Closer to original Event Calculus
- The formalization does not depend on Language $\mathcal{E}$
- Based on "Natural" Properties
- Advance Qualification Problem

Conclusion

- The extension of Language $\mathcal{E}$ enable us to:
  - recover and also extend Language $\mathcal{E}$
  - give a semantic meaning to domains that cannot be interpreted in Language $\mathcal{E}$
- Closer to original Event Calculus
- The formalization does not depend on Language $\mathcal{E}$
- Based on "Natural" Properties
- Advance Qualification Problem

Conclusion

- The extension of Language $\mathcal{E}$ enable us to:
    - recover and also extend Language $\mathcal{E}$
    - give a semantic meaning to domains that cannot be interpreted in Language $\mathcal{E}$
- Closer to original Event Calculus
- The formalization does not depend on Language $\mathcal{E}$
- Based on “Natural” Properties
- Advance Qualification Problem

Thanks for listening!!!

Questions ?

Antonis C. Kakas, Rob Miller, and Francesca Toni, *An argumentation framework of reasoning about actions and change*, LPNMR, 1999, pp. 78–91.

R Kowalski and M Sergot, *A logic-based calculus of events*, New Gen. Comput. **4** (1986), no. 1, 67–95.

M. Thielscher, *The Qualification Problem: A Solution to the Problem of Anomalous Models*, AIJ **131** (2001), no. 1–2, 1–37.