A Logic for Decidable Reasoning about Services

Yilan Gu
Dept. of Computer Science
University of Toronto

Mikhail Soutchanski
Dept. of Computer Science
Ryerson University

July 16, 2006
Outline

- Motivations
- Preliminaries
- Specification of the modified situation calculus for services
- Decidable reasoning about actions in this logic
- Discussions and future work
Shopping Online

Clients (buyers)

Requests (E.g., buy/return books)

Web Servers (E.g., Amazon)

Arrangement

Inventory

Shipping
Motivations

- Usually suppliers (Web servers) could not get complete information (OWA)
- Need composition of atomic services to achieve the clients’ requests
- Integrating Semantic webs with Web services
- Representing the dynamics
  - What needs to be represented?
    - atomic services (i.e., actions), dynamic environment (such as what books are available currently), the effect of service action
  - Expectations:
    - Represent actions for large/infinite domains (such as people, weight, time)
    - Be able to represent knowledge such as “there exist some ...”
- For composite services and the environment,
  - What do we care about? (reasoning)
    - Whether the composite services are possible to be executed successfully?
    - Whether certain properties/goals can be satisfied after the execution?
  - Expectations: efficient reasoning (here, decidability)
The Situation Calculus

• A first-order logic language
  – Represent actions and effects in a natural way
  – Very compact

• Three sorts:
  – Actions: \textit{buyBook}(x,y), \textit{returnBook}(x,y), …
  – Situations: \( S_0 \), \( do(a,s) \), \( do([a_1,\ldots,a_n],s) \)
  – Objects: things other than actions and situations.
    E.g., places, names, numbers, etc.

• Fluents: system features whose truth values may vary.
  E.g., \textit{instore}(x,s), \textit{boughtBook}(x,y,s), \textit{bought}(x,y,s)…
Basic action theory $\mathcal{D}$

- A set of first-order axioms to model actions and effects
- Precondition axioms for actions $\mathcal{D}_{ap}$:
  \[
  \text{Poss}(\text{buyBook}(x,y),s) \equiv \text{client}(x) \land \text{book}(y) \land \text{instore}(y,s)
  \]
- Successor state axioms $\mathcal{D}_{ss}$:
  \[
  \text{bought}(x,y,\text{do}(a,s)) \equiv a = \text{buybook}(x,y) \lor a = \text{buyCD}(x,y)
  \]
  \[
  \text{bought}(x,y,s) \land \neg (a = \text{returnbook}(x,y) \lor a = \text{returnCD}(x,y))
  \]
- Axioms for initial database $\mathcal{D}_{S0}$:
  - Knowledge known to be true in the situation $S_0$
  - Non-changeable facts
  - Open world assumption
Reasoning about Actions

• E.g., \( (\exists x)(\forall y)(\forall y') boughtBook(x,y,S) \land boughtBook(x,y',S) \supset y=y' \)

• Key reasoning mechanism -- regression operator \( R \)
  – Successor state axioms support regression in a natural way
  If \( F(x_1,\ldots,x_n,do(a,s)) \equiv \Psi_F(x_1,\ldots,x_n,a,s) \), then
      \[
      R[F(t_1,\ldots,t_n,do(a',S))] = R[\Psi_F'(t_1,\ldots,t_n,a',S)].
      \]

• Important properties for regression
  – \( D \models W \equiv R[W] \)
  – \( D \models W \text{ iff } D_{S_0} \cup D_{una} \models R[W] \)

\[
\begin{array}{c}
\text{\( W_0(S_0) \)} \\
\xrightarrow{R} \\
\xrightarrow{R} \\
\xrightarrow{R}
\end{array}
\]
Disadvantages of the Situation Calculus

Advantage: representing actions and effects very compactly.

Disadvantage: reasoning for actions in general is undecidable under the open world assumption (OWA).

1. Can we get rid of the disadvantage?
2. Can we specify the Semantic Web features in a natural way?

Solution: Consider a fragment of first-order logic $C^2$. 
Description Logics v.s. $C^2$

- **Description logics**
  - Base of OWL
  - Different varieties
  - $\mathcal{ALCQIO}(\sqcap, \sqcup, \neg, |, id)$

- **$C^2$** – a fragment of FOL
  - At most two variables $x, y$
  - No function symbols
  - Add counting quantifiers $\exists \geq n, \exists \leq n$

$\mathcal{ALCQIO}(\sqcap, \sqcup, \neg, |, id)$ v.s. $C^2$

- Concept names $\Leftrightarrow$ unary predicates
  
  \[ \text{instore} --- \text{instore}(x) \]

- Role names $\Leftrightarrow$ binary predicates
  
  \[ \text{boughtBook} --- \text{boughtBook}(x,y) \]

- E.g., $\exists \geq n R.C \Leftrightarrow \exists \geq n y.R(x,y) \land C(y)$, $\forall R.C \Leftrightarrow \forall y.R(x,y) \supset C(y)$
  
  \[ \neg C \Leftrightarrow \neg C(x), \neg Cl \sqcap C2 \Leftrightarrow C1(x) \land C2 (x) \]
Decidability of DLs and $C^2$

- [Borgida 1996] $\text{ALCQIO}(\sqcap, \sqcup, \neg, |, id)$ plus cross-product $\Leftrightarrow C^2$.
- We showed that: $C1 \times C2 = (R \sqcup \neg R)_{C2} \sqcap ((R \sqcup \neg R)_{C1})^{-}$.
- [Grädel et al., Pacholski et al. 1997] $C^2$ is decidable even under OWA.

\[ \text{ALCQIO}(\sqcap, \sqcup, \neg, |, id) \Leftrightarrow C^2, \text{ the translation algorithm is linear to the size of the given formula.} \]

\[ \text{ALCQIO}(\sqcap, \sqcup, \neg, |, id) \text{ is decidable even under OWA.} \]

- Other advantages
  - The features in Semantic Webs can be easily represented in $C^2$.
  - The reasoning in $C^2$ can also be easily translated into DLs.
  - May use current existing efficient DL reasoners for $C^2$ formulas.
The Decidable Situation Calculus $\mathcal{L}_{SC}^{DL}$

- **Sorts:**
  - Terms of *objects* are either variable $x$, variable $y$, or constants
  - Action functions have at most two arguments
  - Variable symbol $a$ of sort *action* and symbol $s$ of sort *situation* are the only additional variable symbols

- **Fluents with either two or three arguments:**
  - (Dynamic) concepts $\text{instore}(x,s)$, …
  - (Dynamic) roles $\text{boughtBook}(x,y,s)$, $\text{boughtCD}(x,y,s)$, $\text{bought}(x,y,s)$, …

- **Facts with either one or two arguments:**
  - (Static) concepts $\text{person}(x)$, $\text{client}(x)$, $\text{book}(x)$, $\text{cd}(x)$, …
  - (Static) roles $\text{hasCreditcard}(x,y)$, …

- **Logic:** add counting quantifiers $\exists \geq n$, $\exists \leq n$
The Basic Action Theory of $\mathcal{L}_{SC}^{DL}$

- **Precondition axioms:**
  - The RHS is $C^2$ if the situation argument $s$ is suppressed

- **Success state axioms:**
  - Allow counting quantifiers
  - Variables $a$ and $s$ are free in the RHS of the axioms
  - Moreover, $x, y, a$ and $s$ are the only variables (both free and quantified)

- **Axioms for initial databases:** (with OWA)
  - Each axiom is $C^2$ if $S_0$ is suppressed

**Purpose:** to ensure the regression result is $C^2$ regardless $S_0$. 
Extensions of the Basic Action Theory

• Allowing specify certain features similar to DLs

• Acyclic TBox axioms:
  – Dynamic ones: \( C(x,s) \equiv \Phi_C(x,s) \) (\( C \) — defined dynamic concept)
  – Static ones: \( C(x) \equiv \Phi_C(x) \) (provided in the \( D_{S0} \))
  – The RHS is \( C^2 \) when the situation argument \( s \) is suppressed
    E.g., \( valCust(x,s) \equiv person(x) \land (\exists y^\geq 3)\ bought(x,y,s) \)
    \( client(x) \equiv person(x) \land (\exists y)\ hasCreditcard(x,y) \)
  – Reasoning: use lazy unfolding for Dynamic ones

• RBox axioms:
  – For taxonomic reasoning purpose
  – \( R1 \supset R2 \) for role \( R1, R2 \)
    E.g., \( boughtBook(x,y,s) \supset bought(x,y,s), boughtCD(x,y,s) \supset bought(x,y,s) \)
  – Correctly compiled in \( D_{SS} \). I.e., \( D \models (\forall x,y,s).R1(x,y)[s] \supset R2(x,y)[s] \)
Reasoning: Regression + Lazy Unfolding

• Expectations
  – Resulting formula should be $C^2$ if $S_o$ is suppressed
  – Be able to handle dynamic TBox axioms

• Reiter’s regression operator is not suitable: introduce new variables

• Formula $W$ that is regressable in $L^{DC}_{SC}$
  – The situation term in $W$ are ground
  – Variables in $W$ can only include $x, y$

• Modified regression operator $R$
  – When $W$ is not atomic, the operator is still defined recursively
  – Add $R[\exists^n v.W] = \exists^n v. R[W]$
  – Reuse variables $x$ and $y$ when $W$ is atomic (examples on the next slide)
  – When $W$ is a defined dynamic concept, use TBox axioms (lazy unfolding)
A Regression Example in $\mathcal{L}_{\text{SC}}^{\text{DL}}$

\[A1 = buyCD(\text{Tom, BackStreetBoys}),\]
\[A2 = buyBook(\text{Tom, HarryPotter}),\]
\[A3 = buyBook(\text{Tom, TheFirm})\]

\[\mathcal{R}[(\exists x).\text{valCust}(x, do([A1,A2,A3],S_0))]\]
\[= \mathcal{R}[(\exists x).\text{person}(x) \land (\exists \geq 3 y) \text{bought}(x, y, do([A1,A2,A3], S_0))] \text{ (lazy unfolding)}\]
\[= (\exists x).\text{person}(x) \land (\exists \geq 3 y) \mathcal{R}[\text{bought}(x, y, do([A1,A2,A3], S_0))]\]
\[= \ldots \text{ (recursively do regression using the successor state axioms)}\]
\[= (\exists x).\text{person}(x) \land (\exists \geq 3 y) [(x=\text{Tom} \land y = \text{TheFirm}) \lor\]
\[\ldots (x=\text{Tom} \land y = \text{HarryPotter}) \lor\]
\[bought(x,y,S_0) \]
Important Properties

Suppose $W$ is a regressable formula of $\mathcal{L}_{SC}^{DL}$ with the basic action theory $\mathcal{D}$

- The regression $R[W]$ terminates in a finite number of steps.
- $R[W]$ is a $C^2$ formula if $S_0$ is suppressed
- $\mathcal{D} \models W \equiv R[W]$
- $\mathcal{D} \models W$ iff $\mathcal{D}_S \cup \mathcal{D}_{una} \models R[W]$
- The problem whether is $\mathcal{D} \models W$ decidable
  - $\mathcal{D}_{S0} \cup \mathcal{D}_{una} \models R[W]$ is a decidable reasoning in $C^2$ when $S_0$ is suppressed everywhere
- The executability problems and projection problems are decidable in $\mathcal{L}_{SC}^{DL}$
Discussions and Future Work

• Conclusions
  – Formalize a decidable language suitable for Web services
  – Have compact powerful expression power

• Other related researches
  – [McIlraith and Son 2002] assumes that sufficient information is available
  – [Berardi et al. 2003] uses propositional dynamic logic to model services
    e-services $\rightarrow$ constants, fluents $\rightarrow$ F(s) (propositional fragment of the
    situation calculus)
    to capture the change of the world over time instead of caused by actions
  – [Baader et al. 2005] defines a service using a triple of sets of DL formulas

• Possible future work
  – Implementations
  – Consider the knowledge base progression/update problem in $\mathcal{L}_{SC}^{DL}$
  – Etc.