FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Methods and Algorithms for Managing Inconsistency in Multi-Context Systems

Principles of Knowledge Representation and Reasoning — Doctoral Consortium
Toronto, Canada, 2010

Peter Schüller
Faculty Mentors: Thomas Eiter and Michael Fink

KBS Knowledge-Based Systems Group
WWTF Wiener Wissenschafts-, Forschungs- und Technologiefonds

Vienna University of Technology
Institut of Information Systems
Knowledge Based Systems Group

Multi-Context Systems: Subject of Investigation

- MCSs provide a formalism for interlinking knowledge bases.
- What is a multi-context system?
  A collection of contexts:
  $M = (C_1, \dots, C_n)$
- What is a context C_i?
  $C_i = (L_i,$ a logic
  $kb_i,$ the context's knowledge base
  $br_i)$ a set of bridge rules
- What is a logic L_i?
  $L = (\mathbf{KB}_L,$ set of well-formed knowledge bases
  $\mathbf{BS}_L,$ set of possible belief sets
  $\mathbf{ACC}_L)$ acceptability function $\mathbf{KB}_L \longrightarrow 2^{\mathbf{BS}_L}$
  Given a knowledge base, $\mathbf{ACC}_L$ answers:
  Which belief sets are accepted?
  ⇒ captures many (nonmonotonic) formalisms
  ⇒ allows multiple extensions (e.g., Reiter's default logic)
- What is a bridge rule?

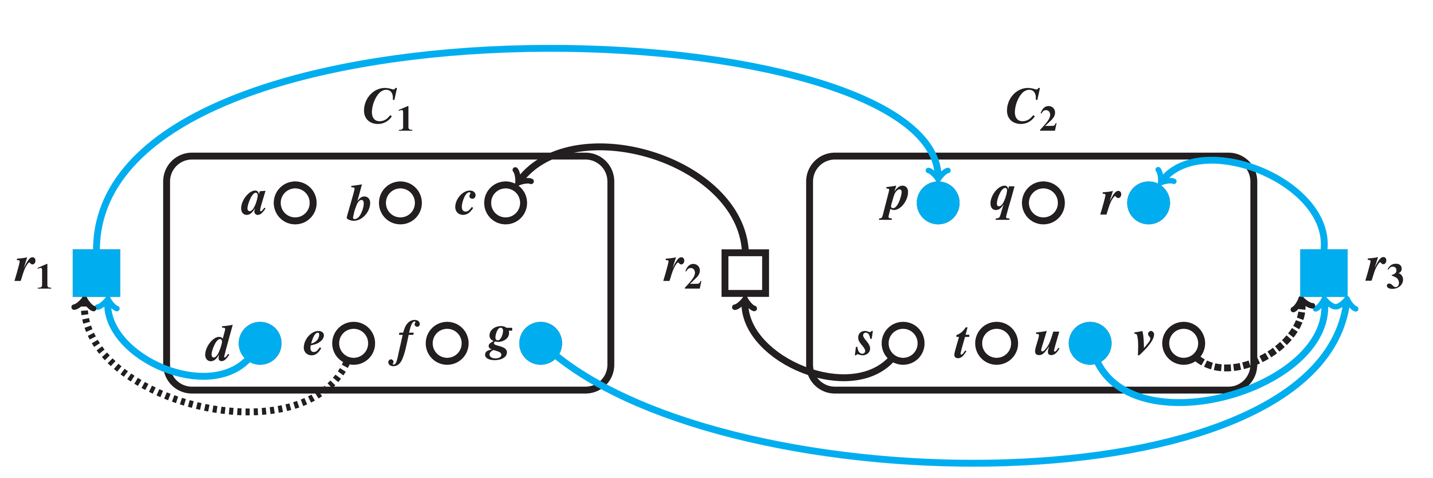
$$(k : s) \leftarrow (c_1 : p_1), \dots, (c_j : p_j), \\ \mathbf{not}\ (c_{j+1} : p_{j+1}), \dots, \mathbf{not}\ (c_m : p_m).$$

- How about semantics?

Equilibrium = stable belief state $S = (S_1, \dots, S_n)$, s.t.

- H_i is calculated from bridge rules applicable wrt. S
- each context accepts S_i using $kb_i \cup H_i$: $S_i \in \mathbf{ACC}_i(kb_i \cup H_i)$

MCS Example: Semantics



$$r_1 = (2 : p) \leftarrow (1 : d), not(1 : e).$$
$$r_2 = (1 : c) \leftarrow (2 : s).$$
$$r_3 = (2 : r) \leftarrow (1 : g), (2 : u), not(2 : v).$$

- Does C_1 accept $\{d, g\}$ without inputs?
  ⇒ check if $\{d, g\} \in \mathbf{ACC}_1(kb_1)$
- Does C_2 accept $\{u\}$ with inputs $\{p, r\}$?
  ⇒ check if $\{u\} \in \mathbf{ACC}_2(kb_2 \cup \{p, r\})$
- If both is true,
  $(\{d, g\}, \{u\})$ is an equilibrium of M!

MCS Example: Inconsistency

Scenario: "Mentor C_1 and students C_2 and C_3 write a paper p"

$$kb_1 = \{Intuitive \sqsubseteq Readable, \exists contains.theorems \sqsubseteq ManyTheorems\}$$
$$kb_2 = \{theorems(X) \leftarrow writeTheorems(X).\}$$
$$kb_3 = \{intuition(X) \leftarrow makeIntuitive(X).\}$$

Bridge rules:

$$r_1 = (1 : contains(p, theorems)) \leftarrow (2 : theorems(p)).$$
$$r_2 = (1 : Intuitive(p)) \leftarrow (3 : intuition(p)).$$
$$r_3 = (2 : writeTheorems(p)) \leftarrow \mathbf{not}\ (1 : ManyTheorems(p)).$$
$$r_4 = (3 : makeIntuitive(p)) \leftarrow (1 : ManyTheorems(p)).$$

This MCS has no equilibrium! — why?

C_1 accepts:	Effect:
$ManyTheorems(p)$	C_2 does not write theorems… …C_1 does not accept $ManyTheorems(p)$
$\neg ManyTheorems(p)$	C_2 writes theorems… …C_1 accepts $ManyTheorems(p)$

⇒ no stable belief state

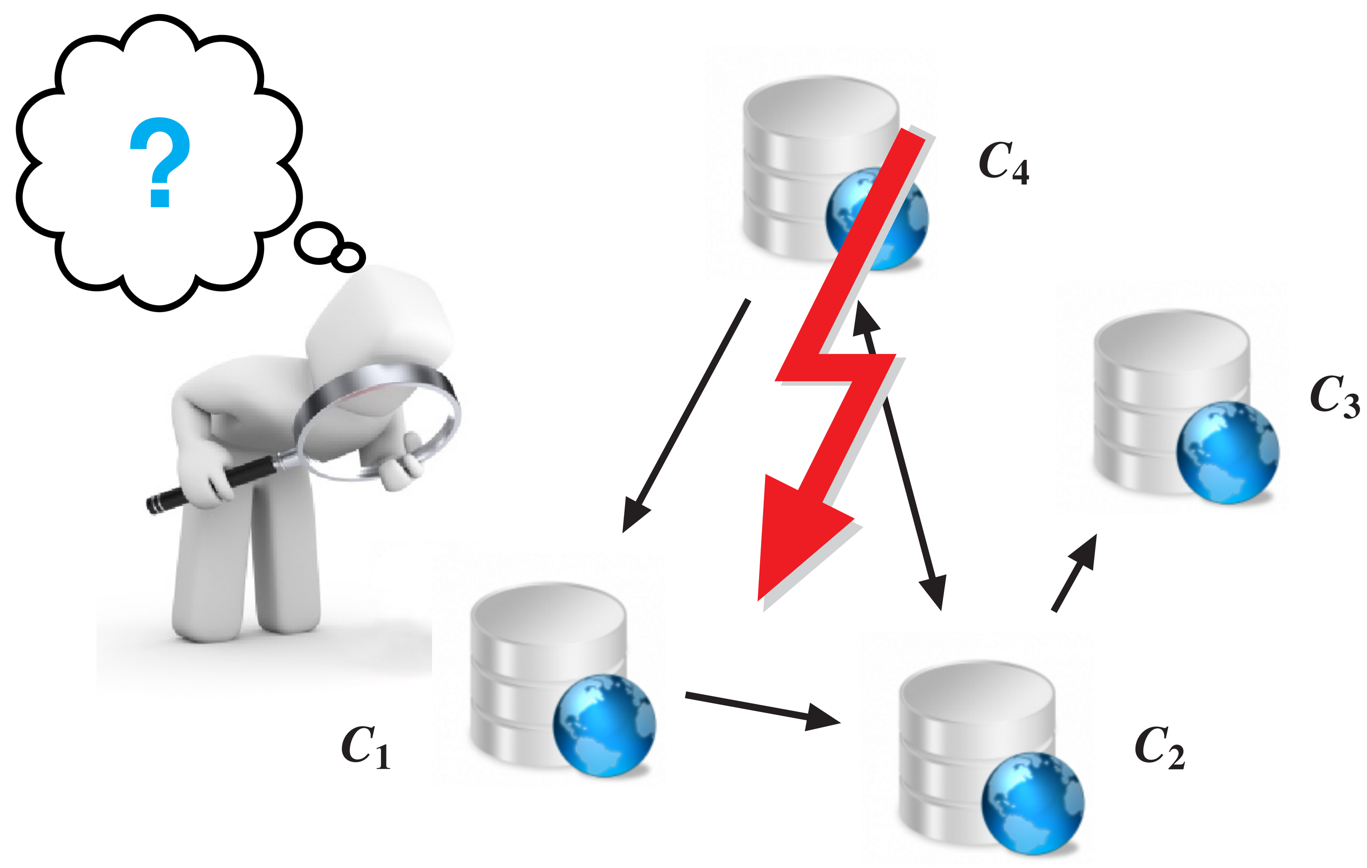
Inconsistency Management Approach

No equilibrium = no useful information!
⇒ Analyze inconsistency to gain information

1. Explain inconsistency in a given MCS
2. Reason about explanations ⇒ find causes and repairs
3. (Semi-)automatically repair the MCS
   ⇒ Obtain useful information in the presence of inconsistency

Basic situation:

- Most inconsistencies arise due to unexpected interactions
  Reasons: ⋆ systems are often connected by ad hoc links
  ⋆ large and complicated contexts and systems
- We can identify reasons for inconsistency by bridge rules
  because bridge rules model the links between contexts
  Disregarded: debugging context internals (use traditional methods)
  ⇒ Assumptions: context is consistent without input



Progress to Date

✓ Framework for explaining inconsistency:

> "Which rules must be deactivated/must fire, to gain consistency?"
> Diagnosis: (D_1, D_2), $D_1, D_2 \subseteq br_M$
> s.t. $M[br_M \setminus D_1 \cup heads(D_2)]$ is consistent.

e.g. $(\{r_1\}, \emptyset)$, or $(\emptyset, \{r_3\})$ in student/mentor example

> "Which rules must be present/not fire, to produce inconsistency?"
> Inconsistency Explanation: (E_1, E_2), $E_1, E_2 \subseteq br_M$
> s.t. for all (R_1, R_2) where $E_1 \subseteq R_1 \subseteq br_M$ and $R_2 \subseteq br_M \setminus E_2$,
> $M[R_1 \cup heads(R_2)]$ is inconsistent.

e.g. $(\{r_1, r_3\}, \{r_1, r_3\})$ in student/mentor example

✓ Experimental prototype
using dlvhex (ASP extension with external atoms)
✓ Complexity analysis
✓ Modularity properties

Current and Future Work

- Partially known MCSs and inconsistency
  (Trust/Policy Contexts will usually hide certificates and/or rules)
- Distributed calculation of inconsistency explanations
- Reasoning about explanations and repair of inconsistencies
- Query answering in the presence of inconsistency
  (related to paraconsistency, belief revision, belief merging)
- Distributed algorithms for inconsistency management

References

- Giunchiglia, F., and Serafini, L. Multilanguage hierarchical logics, or: How we can do without modal logics. *Artificial Intelligence* 65(1):29–70. 1994.
- Gerhard Brewka and Thomas Eiter. Equilibria in heterogeneous nonmonotonic multi-context systems. In *AAAI*, pg 385–390, 2007.
- Thomas Eiter, Michael Fink, Peter Schüller, and Antonius Weinzierl. Finding explanations of inconsistency in nonmonotonic multi-context systems. In *KR*, 2010.

This work was supported by the Vienna Science and Technology Fund (WWTF) under grant ICT 08-020.

Contact: {schueller,eiter,fink}@kr.tuwien.ac.at