Generalizing Multi-Context Systems for Reactive Stream Reasoning Applications[1]

Stefan Ellmauthaler

Computer Science Institute
Leipzig University
Germany

3^{rd} Hybris Workshop
Dresden
November 20^{th} 2013

UNIVERSITÄT LEIPZIG

[1] This work has been presented at the ICCSW 2013 [?]

Outline

1. Motivation
2. Background
3. Preference-based Iterative Managed Multi-Context Systems
4. Reactive Bridge Rules
5. Conclusion & Future Work

Advertisement

Advertisement

3^{rd} Hybris Video

The video can be viewed at `http://goo.gl/eBEIlA`

Assisted Living (AL)

An Application for Artificial Intelligence

The Basic Idea

- Enhance an apartment with an AI which monitors the activities of daily living of the inhabitants.

Assisted Living (AL)

An Application for Artificial Intelligence

The Basic Idea

- Enhance an apartment with an AI which monitors the activities of daily living of the inhabitants.
- Coordinate services by outside health care providers.

The Basic Idea

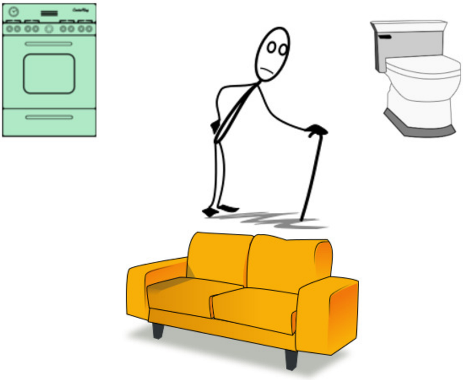
- Enhance an apartment with an AI which monitors the activities of daily living of the inhabitants.
- Coordinate services by outside health care providers.
- Provide supervision and assistance to ensure the inhabitants
  - health,
  - safety, and
  - well-being.

Assisted Living

An Example of AL in Action

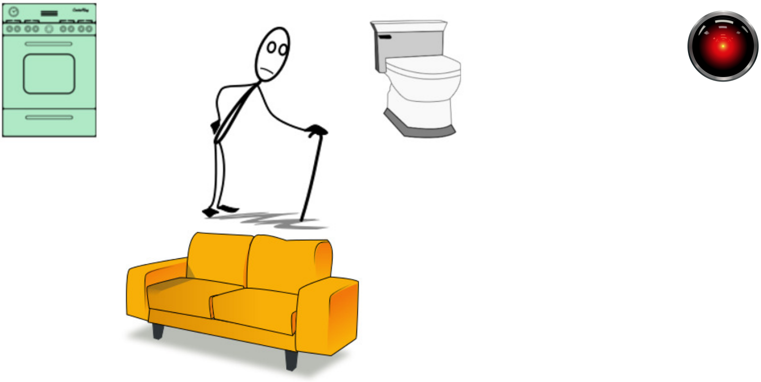
Assisted Living

An Example of AL in Action



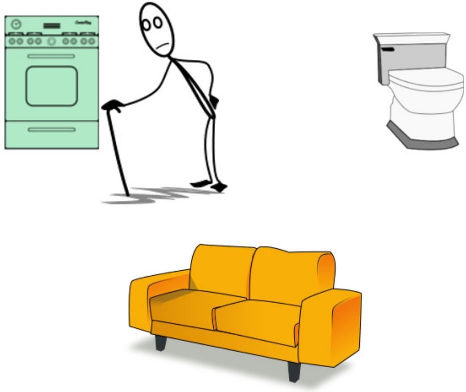
Assisted Living

An Example of AL in Action

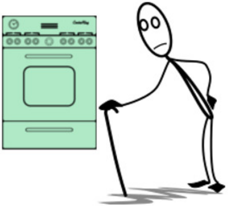


Assisted Living

An Example of AL in Action



What shall the AI believe?

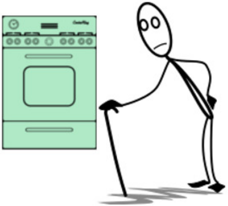


What shall the AI believe?

- The stove is active

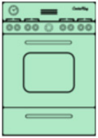
Assisted Living

An Example of AL in Action



What shall the AI believe?

- The stove is active
- Bob is cooking

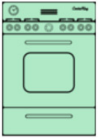


What shall the AI believe?

- The stove is active
- Bob is cooking

Assisted Living

An Example of AL in Action

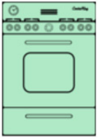


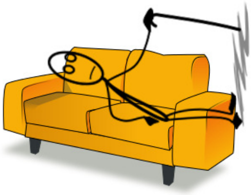
What shall the AI believe?

- The stove is active
- Bob is cooking
- Bob is on the toilet

Assisted Living

An Example of AL in Action



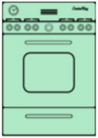


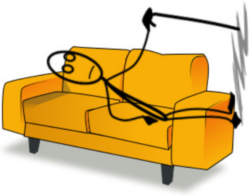
What shall the AI believe?

- The stove is active
- Bob is cooking
- Bob is on the toilet
- Bob is sleeping

Assisted Living

An Example of AL in Action



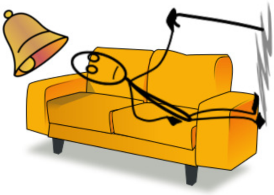


What shall the AI believe?

- The stove is active
- Bob is cooking
- Bob is on the toilet
- Bob is sleeping
- Unwanted situation - what to do now?

Assisted Living

An Example of AL in Action



What shall the AI believe?

- The stove is active
- Bob is cooking
- Bob is on the toilet
- Bob is sleeping
- Unwanted situation - what to do now?
  - Wake up Bob
  - Disable the stove

Realization of this Vision

A first step and some considerations

AL-Environment

- Sensors
- Gadgets to communicate/(re)act
- Reasoning units

Realization of this Vision

A first step and some considerations

AL-Environment

- Sensors
- Gadgets to communicate/(re)act
- Reasoning units

Requirements

- Communication between the components
- Continuous evaluation of the situation
- Intelligent reasoning about intentions and beliefs

Realization of this Vision

A first step and some considerations

AL-Environment

- Sensors
- Gadgets to communicate/(re)act
- Reasoning units

Requirements

- Communication between the components
- Continuous evaluation of the situation
- Intelligent reasoning about intentions and beliefs

Existing Concepts

- (managed) Multi-Context Systems [?]

Realization of this Vision

A first step and some considerations

Requirements

- Communication between the components
- Continuous evaluation of the situation
- Intelligent reasoning about intentions and beliefs

Existing Concepts

- (managed) Multi-Context Systems [?]
- Stream Reasoning concepts
  - oclingo [?]
  - C-SPARQL [?]

(managed) Multi-Context Systems (mMCS)

Definition

A managed Multi-Context System M is a collection $(C_1, \ldots, C_n)$ of managed contexts where, for $1 \leq i \leq n$, each managed context C_i is a quintuple $C_i = (LS_i, kb_i, br_i, OP_i, mng_i)$ such that

- $LS_i = (\mathcal{BS}_{LS_i}, \mathcal{KB}_{LS_i}, \mathcal{ACC}_{LS_i})$ is a logic suite,
- $kb_i \in \mathcal{KB}_{LS_i}$ is a knowledge base,
- OP_i is a management base,
- br_i is a set of bridge rules for C_i, with the form

  $$op_i \leftarrow (c_1 : p_1), \ldots, (c_j : p_j), not(c_{j+1} : p_{j+1}), \ldots, not(c_m : p_m).$$

  such that $op_i \in F^{OP_i}_{LS_i}$ and for all $1 \leq k \leq m$ there exists a context $c_k \in (C_1, \ldots, C_n)$ such that $p_k \in S \in \mathcal{BS}_{LS_{c_k}}$, and
- mng_i is a management function over LS_i and OP_i.

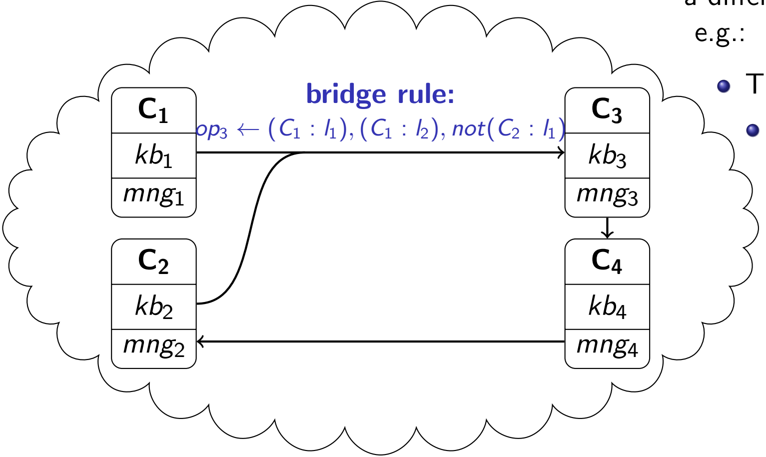
mMCS
Semantics

Definition

Let $M = (C_1, \dots, C_n)$ be an mMCS. A belief state $S = (S_1, \dots, S_n)$ is an equilibrium of M iff for every $1 \leq i \leq n$ there exists some $(kb'_i, ACC_{LS_i}) \in mng_i(app_i(S), kb_i)$ such that $S_i \in ACC_{LS_i}(kb'_i)$.

mMCS

Intuitive Concept



Each context may use a different formalism e.g.:

- Theorem Prover
- Datalog
- ASP
- SQL
- DL
- ...

Preference-based Iterative Managed Multi-Context Systems

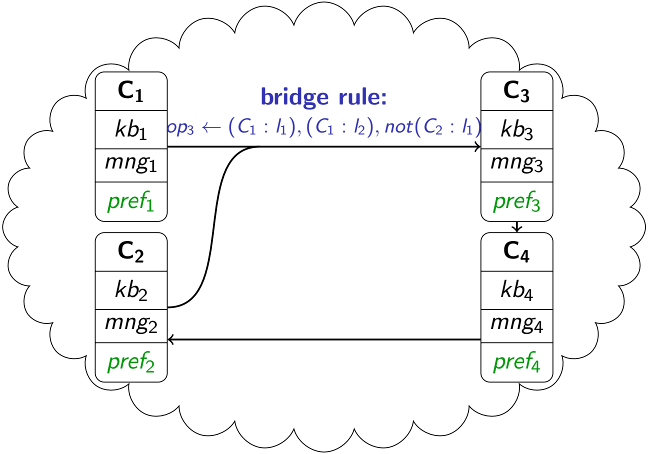
Basic Concepts

- Utilize iterative and stream reasoning approach from potassco [**?**, **?**]
- Specialized contexts for different tasks

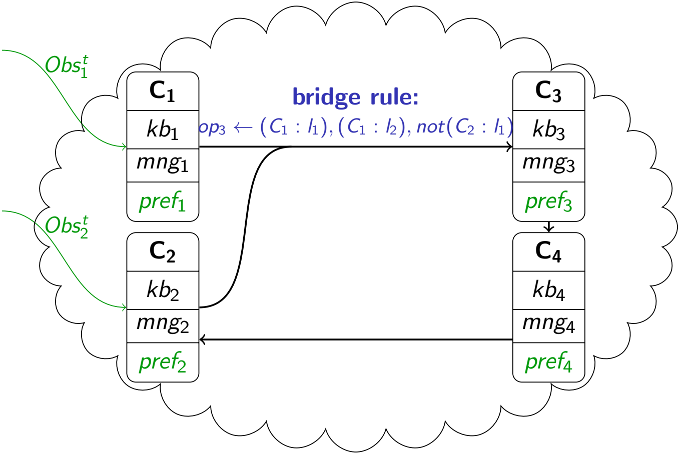
Context types

- Observing Contexts
- Reasoning Contexts
- Control Contexts
  - sliding windows
  - inconsistency handling policies
  - semantics and reasoning modes
  - determine actions
  - decide meta-reasoning aspects

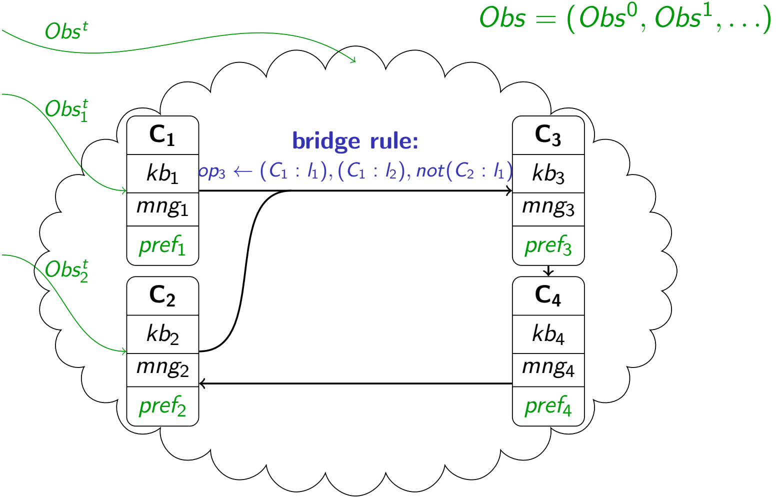
Preference-based Iterative Managed Multi-Context Systems (pimMCS)



Preference-based Iterative Managed Multi-Context Systems (pimMCS)

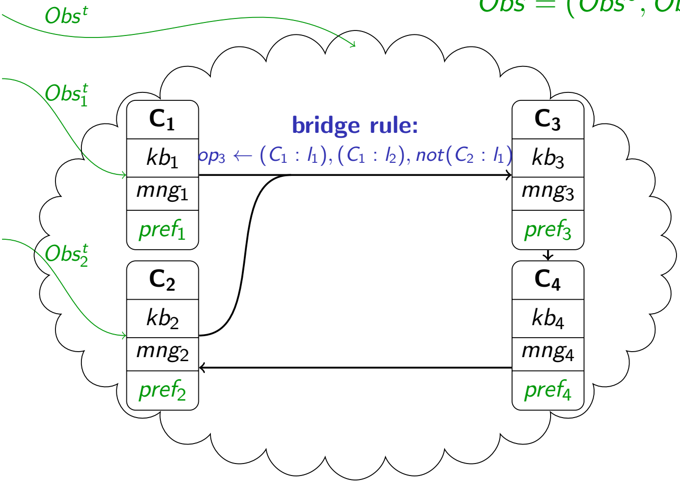


Preference-based Iterative Managed Multi-Context Systems (pimMCS)



Preference-based Iterative Managed Multi-Context Systems (pimMCS)

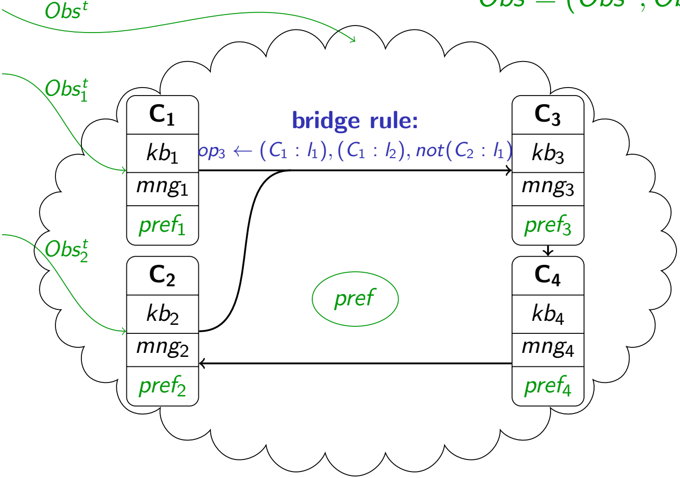
$Obs = (Obs^0, Obs^1, \ldots)$



$R(Obs) = Kb^0, Eq^0, Kb^1, Eq^1, \ldots$

Preference-based Iterative Managed Multi-Context Systems (pimMCS)

$Obs = (Obs^0, Obs^1, \ldots)$



$R(Obs) = Kb^0, Eq^0, Kb^1, Eq^1, \ldots$

- If there is no global equilibrium, no actions between contexts
- Computation of one global equilibrium is expensive [**?**]

- If there is no global equilibrium, no actions between contexts
- Computation of one global equilibrium is expensive [**?**]

$\Rightarrow$ fast reaction to events is highly unlikely

Reactive Bridge Rules

Concept Idea

- Use bridge rules on local belief sets instead of global equilibria
- All contexts have input streams
- Manipulate the input stream of other contexts

Comparison to pimMCS

- Contexts do not have to wait for the global equilibria
- No agreement neccessary
- Communication in case of emergencies is more immediate
- Inconsistency handling needs to be done via stream handling

Reactive Bridge Rules

Definition

A Reactive Bridge Rule (RBR) r for a context C_i of a collection of n contexts is a rule of the form

$$t, j : h \leftarrow b_1, \ldots, b_k, not\ b_{k+1}, \ldots, not\ b_m$$

where

- $t \in \{b, c\}$ specifies whether the literals need to be evaluated bravely or cautiously,
- $j \leq n$ specifies which context will be provided with additional information,
- h is information which may be added to the input stream of C_j, and
- for $l \leq m$, b_l is a literal.

Reactive Bridge Rules

Definition

Let r be an RBR of a context C_i, $ACC_{LS_i} \in \mathcal{ACC}_{LS_i}$ be a selected semantics, and $S = \{S_1, \ldots, S_j\}$ be the belief sets of C_i at step t, such that $S = ACC_{LS_i}(kb_i^t)$, where kb_i^t is the knowledge base of context C_i at step t.

- If r is a cautious RBR, it is satisfied if $\forall_{B \in S}(b^+(r) \subseteq B \wedge b^-(r) \cap B = \emptyset)$.
- If r is a brave RBR, it is satisfied if $\exists_{B \in S}(b^+(r) \subseteq B \wedge b^-(r) \cap B = \emptyset)$.

If a rule r is satisfied, then h will be added to the input stream of the context C_j at step $t + 1$.

Conclusion & Future Work

Conclusion

We have introduced

- pimMCS to compute equilibria on stream based MCS
- RBRs to modify input streams of other contexts

Conclusion & Future Work

Conclusion

We have introduced

- pimMCS to compute equilibria on stream based MCS
- RBRs to modify input streams of other contexts

In addition there is

- a combination of pimMCS and RBRs
  - reactive managed Multi-Context Systems (rmMCS)
  - computes runs with equilibria like pimMCS
  - free capacities used for additional belief sets
  - RBRs may fire during the computation of the equilibria

Conclusion & Future Work

Conclusion

We have introduced

- pimMCS to compute equilibria on stream based MCS
- RBRs to modify input streams of other contexts

In addition there is

- a combination of pimMCS and RBRs
  - reactive managed Multi-Context Systems (rmMCS)
  - computes runs with equilibria like pimMCS
  - free capacities used for additional belief sets
  - RBRs may fire during the computation of the equilibria

Further Work

- Restrictions to Contexts
- Side effects of rmMCS
- Instantiation
- Implementation
- Reactive “extensions” for one-shot formalisms

Thank you!

The pictures used in this talk are taken from [**?**, **?**]

References I

Appendix

Preference-based Iterative Managed Multi-Context Systems

Definition

Let M be a managed MCS with contexts $C = (C_1, \ldots, C_n)$ ($C_1, \ldots, C_k$ are observer contexts), where $C_i \in C$ is a quintuple $C_i = (LS_i, kb_i, br_i, OP_i, mng_i, pref_i)$. Let $Obs = (Obs^0, Obs^1, \ldots)$ be a sequence of observations, that is, for $j \geq 0$, $Obs^j = (Obs_i^j)_{i \leq k}$, where Obs_i^j is the new (sensor) information for context i at step j, which is formalized as sets of formulas.

A run R of M induced by Obs is a sequence

$$R = Kb^0, Eq^0, Kb^1, Eq^1, \ldots$$

where

Definition

A run R of M induced by Obs is a sequence

$$R = Kb^0, Eq^0, Kb^1, Eq^1, \ldots$$

where

- $Kb^0 = (Kb_i^0)_{i \leq n}$ is the collection of initial knowledge bases, Eq^0 an equilibrium of Kb^0,
- for $j \geq 1$ and $i \leq n$, Kb_i^j is the knowledge base of context C_i produced by the context's management function for the computation of Eq^{j-1}, and $Kb^j = (Kb_i^j)_{i \leq n}$,
- for $j \geq 1$, Eq^j is an equilibrium for the knowledge bases

$$(Kb_0^j \cup Obs_0^j, \ldots, Kb_k^j \cup Obs_k^j, Kb_{k+1}^j, \ldots, Kb_n^j).$$

$(C, Obs, pref)$ is called a preference-based iterative managed Multi-Context System.