Using a Blackboard Architecture for Assertion Application in Proof Planning

Christoph Benzmüller, Andreas Meier, and Volker Sorge

Universität des Saarlandes, Saarbrücken, Germany

Motivation

Proof Planning in the OMEGA System:

- considers theorem proving as planning process
- employs methods as planning operators
- applies mathematical facts stored in data base: axioms, theorems, lemmas (so-called assertions)

During proof planning:

Methods access data base to look up and apply Assertions:

$$\frac{Prems}{Goal} Assertion$$

Example

Classifying residue class structures wrt.

- algebraic structure they form (semi-group, monoid etc.)
- isomorphic structures

Proof obligations, e.g., $Closed(\mathbb{Z}_5, \lambda x \lambda y (x \bar{*} y) \bar{+} \bar{3}_5)$

- $\mathbb{Z}_5$: set of integer congruence classes modulo 5
- $\lambda x \lambda y (x \bar{*} y) \bar{+} \bar{3}_5$: binary operation on $\mathbb{Z}_5$

One approach: apply known theorems from data base

Example

$ClosedConst : \forall n{:}\mathbb{Z}.\, \forall c{:}\mathbb{Z}_n.\, Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, c)$

$ClosedFV : \forall n{:}\mathbb{Z}.\, Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, x)$

$ClosedSV : \forall n{:}\mathbb{Z}.\, Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, y)$

$$Closed\bar{+} : \forall n{:}\mathbb{Z}.\, \forall op_1.\, \forall op_2.\, (Closed(\mathbb{Z}_n, op_1) \wedge Closed(\mathbb{Z}_n, op_2)) \Rightarrow Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, (x\ op_1\ y)\bar{+}(x\ op_2\ y))$$

$$Closed\bar{-} : \forall n{:}\mathbb{Z}.\, \forall op_1.\, \forall op_2.\, (Closed(\mathbb{Z}_n, op_1) \wedge Closed(\mathbb{Z}_n, op_2)) \Rightarrow Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, (x\ op_1\ y)\bar{-}(x\ op_2\ y))$$

$$Closed\bar{*} : \forall n{:}\mathbb{Z}.\, \forall op_1.\, \forall op_2.\, (Closed(\mathbb{Z}_n, op_1) \wedge Closed(\mathbb{Z}_n, op_2)) \Rightarrow Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, (x\ op_1\ y)\bar{*}(x\ op_2\ y))$$

Motivation

Determining applicable assertions can be difficult

Idea: Separate search for applicable assertions from main proving process

Addressed aspects

- Concurrency: parallelizing applicability check to gain efficency and any-time behavior
- Flexibility: parameterize applicability check to employ, for instance, different matching procedures
- Robustness: become independent from data base details such as theorem/theory names

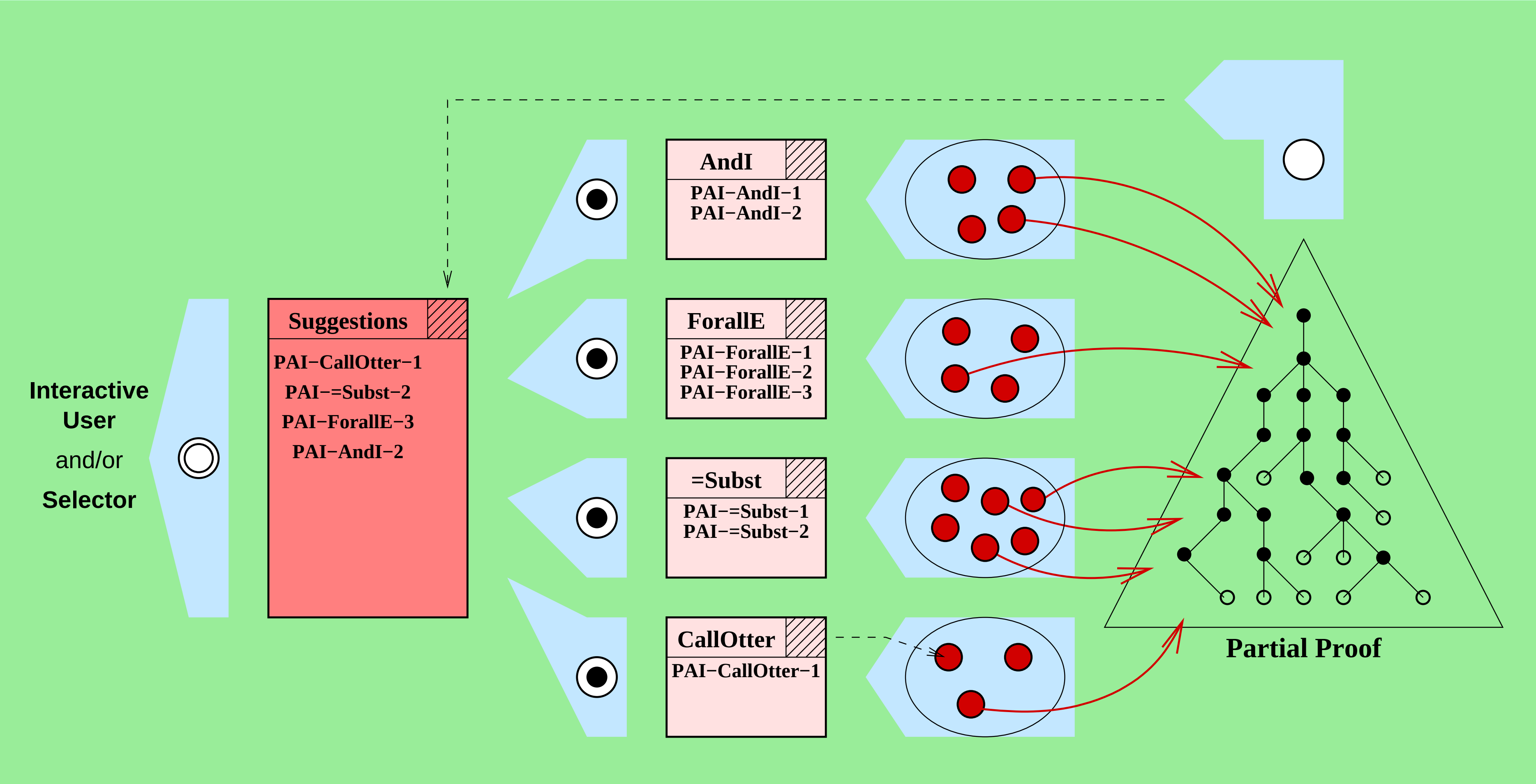
Realization – The Idea

- Form clusters of related theorems
- Use two filters (simple & complex)
- Dynamically extend mechanism for new assertions

Realization – The Idea

- Employ the hierarchical blackboard architecture Ω-ANTS
- In-built concurrency
- Enables cooperation of knowledge sources (so-called agents)

Realization – The Idea



Realization – The Mechanism

Applicability check performed in three stages:

Realization – The Mechanism

Applicability check performed in three stages:

- check for suitable goal (first filter)

Realization – The Mechanism

Ωmega

Applicability check performed in three stages:

- check for suitable goal (first filter)
- try to match theorems
  (with several algorithms in parallel)

Realization – The Mechanism

Applicability check performed in three stages:

- check for suitable goal (first filter)
- try to match theorems (with several algorithms in parallel)
- search for possible premises of an applicable theorem

Realization – The Mechanism

Ωmega

Applicability check performed in three stages:

- Check for suitable goal (first filter)
- Try to match theorems (with several algorithms in parallel)
- Search for possible premises of an applicable theorem

Formation of clusters done automatically

$\Rightarrow$ Special predicate to acquire theorems

Example (continued)

$$\mathfrak{G}^{\{G\}}_{\{\},\{T,P\}} = \{G\text{: } G \text{ contains the } Closed \text{ predicate}\}$$

$$\mathfrak{F}^{\{T\}}_{\{G\},\{P\}} = \{T\text{: Conclusion matches } G \text{ with first order matching}\}$$

$$\{Acquisition\text{: Conclusion contains } Closed \text{ as outermost}\}$$

$$\mathfrak{F}^{\{T\}}_{\{G\},\{P\}} = \{T\text{: Conclusion matches } G \text{ with special algorithm}\}$$

$$\{Acquisition\text{: Conclusion contains } Closed \text{ as outermost}\}$$

$$\mathfrak{S}^{\{P\}}_{\{G,T\},\{\}} = \{P\text{: The nodes matching the premises of } T\}$$

Example (continued)

Goal: $Closed(\mathbb{Z}_5, \lambda x\textbf{.}\, \lambda y\textbf{.}\, (x \bar{*} y) \bar{+} \bar{3}_5)$

Closed

Goal contains $Closed$ predicate?

Example (continued)

Goal: $Closed(\mathbb{Z}_5, \lambda x\ldotp \lambda y\ldotp (x \bar{*} y) \bar{+} \bar{3}_5)$

Closed

(Goal:Closed(...))

$ClosedConst : Closed(\mathbb{Z}_n, \lambda x\ldotp \lambda y\ldotp c)$

$ClosedFV : \ldots \quad \lambda x\ldotp \lambda y\ldotp x$

$ClosedSV : \ldots \quad \lambda x\ldotp \lambda y\ldotp y$

with FO matching?

$Closed\bar{+} : \ldots \quad \lambda x\ldotp \lambda y\ldotp (\ldots \bar{+} \ldots)$

$Closed\bar{-} : \ldots \quad \lambda x\ldotp \lambda y\ldotp (\ldots \bar{-} \ldots)$

$Closed\bar{*} : \ldots \quad \lambda x\ldotp \lambda y\ldotp (\ldots \bar{*} \ldots)$

with special algorithm?

Example (continued)

Goal: $Closed(\mathbb{Z}_5, \lambda x.\, \lambda y.\, (x \bar{*} y) \bar{+} \bar{3}_5)$

Closed
(Goal:Closed(. . .))
(Goal: Closed(. . .), Thm:$Closed\bar{+}$)

Nodes matching the premises of Thm?

Example (continued)

Goal: $Closed(\mathbb{Z}_5, \lambda x. \lambda y. \bar{3}_5)$

Closed

Goal contains $Closed$ predicate?

Example (continued)

Goal: $Closed(\mathbb{Z}_5, \lambda x.\, \lambda y.\, \bar{3}_5)$

Closed

(Goal:Closed(. . .))

$ClosedConst : Closed(\mathbb{Z}_n, \lambda x.\, \lambda y.\, c)$

$ClosedFV : \quad \ldots \quad \lambda x.\, \lambda y.\, x$

$ClosedSV : \quad \ldots \quad \lambda x.\, \lambda y.\, y$

$Closed\bar{+} : \quad \ldots \quad \lambda x.\, \lambda y.\, (\ldots \bar{+} \ldots)$

$Closed\bar{-} : \quad \ldots \quad \lambda x.\, \lambda y.\, (\ldots \bar{-} \ldots)$

$Closed\bar{*} : \quad \ldots \quad \lambda x.\, \lambda y.\, (\ldots \bar{*} \ldots)$

with FO matching?

with special algorithm?

Example (continued)

Goal: $Closed(\mathbb{Z}_5, \lambda x. \lambda y. \bar{3}_5)$

Closed
(Goal:Closed(. . .))
(Goal: Closed(. . .), Thm:$ClosedConst$)

Nodes matching the premises of Thm?

General Aspects

1. Interactive theorem proving
   – approach also supports interactive theorem proving
   – Ω-ANTS:
   ranking and suggestion of applicable theorems to user

2. Retrieval from other data bases
   – approach not restricted to particular data base
   – also possible (not implemented yet):
   retrieval from distributed data base via www

Discussion

Alternative use of Ω-ANTS:

- one (or several) agent for each single theorem
- unique agent to search for matching supports

Discussion

Use of alternative techniques

- for knowledge base retrieval:
  hashing or term indexing techniques

Discussion

Use of alternative techniques

- for knowledge base retrieval:
  hashing or term indexing techniques

- for assertion matching:
  higher-order pattern matching
  higher-order (pre-)unification
  theorem proving

Future Work

Evaluation:

- compare our approach with some of the alternative techniques
- conduct a large case study

Future Work

Evaluation:

- compare our approach with some of the alternative techniques
- conduct a large case study

Integration:

- use our approach in combination with / as part of a more advanced knowledge base