Encapsulation for Practical Simplification Procedures

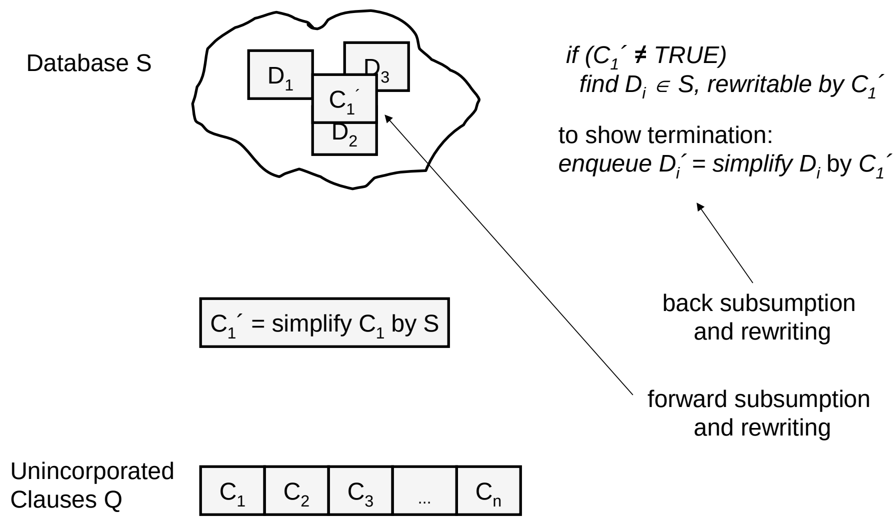
Olga Shumsky Matlin & William McCune

Mathematics and Computer Science Division
Argonne National Laboratory
{matlin,mccune}@mcs.anl.gov

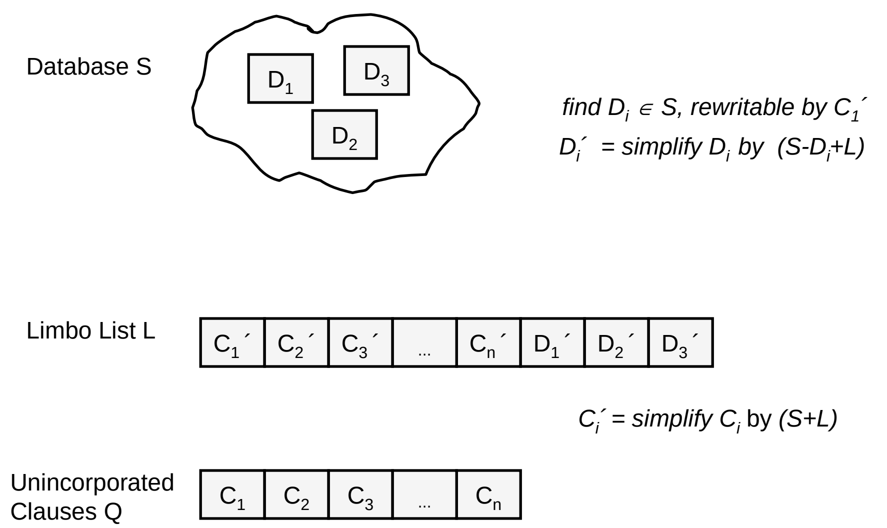
Problem Origin

- First-order resolution and paramodulation theorem prover OTTER
- Interdependent data structures and algorithms, performance concerns
- Sometimes impossible to use the simplest algorithm to solve a particular problem
- Procedures for incorporating newly derived clauses into the main database
- Term rewriting and demodulation are at the core of the incorporation procedures

Simple Solution: Direct Incorporation



Limbo Incorporation



Verification Goals

- Termination of both procedures
  – in practice, implementation of the simplification function (term rewriting) contains an artificial stopping condition
  – in practice, termination of the simplification procedure is assumed
- Database is irreducible
  – no element is rewritable by any other element
- Procedures produce equivalent databases
  – order of rewriting is different, does not produce canonical forms
  – no guarantee that database will contain the same elements
  – show equivalence with respect to evaluation, sufficient to show that each procedure preserves evaluation of the conjunction of clauses in the original database and queue

Key Observations

- Simplification is via term rewriting
  – Rewriting function terminates, rewrites as much as possible, simplifies, is sound, other details unimportant
- Details of the evaluation function unimportant
- Encapsulate simplification and evaluation functions
- Termination of direct incorporation depends on slight modification of the procedure
- Measure function based on a special count function:

  (cons (+ 1 (count q) (count s))
  (+ 1 (count q)))
- Property for irreducibility proof for limbo incorporation

  $\forall$ A,B $\in$ L, pos(A) < pos(B) $\rightarrow$ A does not rewrite B

Solution Statistics

- 4 constrained functions
  - simplify, ceval, scount, true-symbolp
- 8 properties of constrained functions
- 20 functions to model the procedures and correctness properties, including auxiliary functions
- 89 theorems proved, 28 hints required
  - 2 main irreducibility, 2 main soundness theorems

Related Work

- IVY project (ACL2 Case Studies)
  - Verification of the same software
  - IVY checked soundness of OTTER proofs
  - Errors in incorporation procedures could lead OTTER to miss some or all proofs
  - Difficulties in formalization of the evaluation function encouraged the use of encapsulation in this project
- J. L. Ruiz Reina, J. A. Alonso, M. J. Hidalgo, and F. J. Martín. Formal proofs about rewriting using ACL2. *Annals of Mathematics and Artificial Intelligence*, 36(3):239--262, 2002.
  - Formalization of basic reduction and simplification procedures and their properties
  - Our project takes both for granted