Verificare Formala

Haiosta Michelle Cerean Bogdan-Ioan Popovici Adrian-Robert
Bosna Marinel

West University of Timișoara
Faculty of Mathematics and Informatics
Bachelor Study Program: Inginerie Software in Romana

Scientific Coordinator: : Conf. Dr. Madalina Erascu

Monday 27th January, 2025

Overview

Studiu SAT Solver Minisat

În această lucrare analizăm și explicăm funcționarea programului MiniSat, un SAT Solver folosit aât în cercetare cât și în industrie. Vom analiza modul în care algoritmul MiniSAT operează și vom discuta modul în care ar putea fi înbunătățit.

Related Work

SAT Solverele sunt programe care rezolvă problemele de satisfiabilitate logică. Acestea determină dacă o formulă logică, scrisă în Forma Normală Conjunctivă (CNF), poate fi satisfăcută. Cu alte cuvinte, ele caută o combinație de valori pentru variabilele formulei care fac ca întreaga expresie să fie adevărată.

Benchmarks

MiniSat este adesea evaluat folosind benchmark-uri standardizate, care măsoară performanța sa pe probleme complexe. Pentru această analiză, am utilizat un benchmark bazat pe "Simon Cipher", o schemă criptografică simplificată. Problema a fost redusă la o formulă SAT folosind mai puține runde și stări prestabilite, ceea ce facilitează analiza performanței.

Benchmarkul "Simon Cipher" a fost rulat folosind comanda:

```
minisat -var-decay=0.95 -cla-decay=0.93 -rfirst=200 "$benchmark" "$output"
```

Unde:

1. -var-decay: factorul de decădere al variabilelor
2. -cla-decay: factorul de decădere al clauzelor
3. -rfirst: intervalul de restart inițial

Benchmark Results

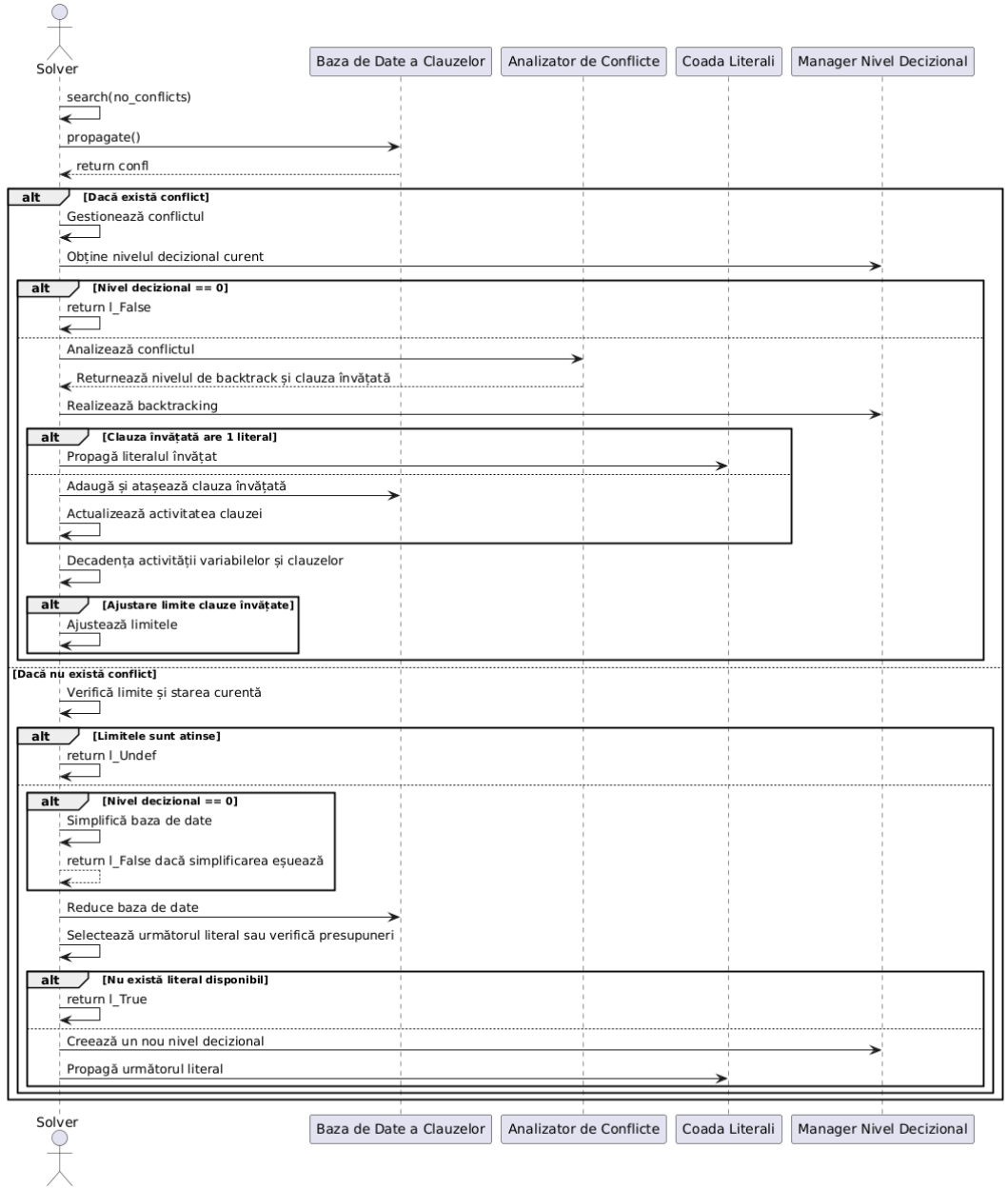
```
Running MiniSat on 2b043efb4bde6d83f7c95a8e8e2d7bf8-simon-r21-0.sanitized.cnf with output to output_file1
WARNING: for repeatability, setting FPU to use double precision
============================[ Problem Statistics ]=============================
|                                                                             |
|  Number of variables:          3488                                         |
|  Number of clauses:           11488                                         |
|  Parse time:                   0.00 s                                       |
|  Eliminated clauses:           0.10 Mb                                      |
|  Simplification time:          0.01 s                                       |
|                                                                             |
============================[ Search Statistics ]==============================
| Conflicts |          ORIGINAL         |          LEARNT          | Progress |
|           |    Vars  Clauses Literals |    Limit  Clauses Lit/Cl |          |
===============================================================================
|       100 |    1280     8192    29440 |     3003      100     83 | 10.092 % |
|       250 |    1280     8192    29440 |     3304      250     60 | 10.092 % |
|       475 |    1280     8192    29440 |     3634      475     55 | 10.092 % |
|       812 |    1280     8192    29440 |     3997      812     53 | 10.092 % |
|      1318 |    1280     8192    29440 |     4397     1318     51 | 10.092 % |
|      2077 |    1280     8192    29440 |     4837     2077     51 | 10.092 % |
|      3216 |    1280     8192    29440 |     5321     3216     50 | 10.092 % |
|      4924 |    1280     8192    29440 |     5853     4924     48 | 10.092 % |
|      7486 |    1280     8192    29440 |     6438     4301     43 | 10.092 % |
|     11330 |    1280     8192    29440 |     7082     4723     44 | 10.092 % |
|  11362814 |    1280     8192    29440 |    35799    21852     43 | 10.092 % |
|  17044326 |    1280     8192    29440 |    39378    24328     48 | 10.092 % |
|  25566595 |    1280     8192    29440 |    43316    19357     37 | 10.092 % |
|  38349998 |    1280     8192    29440 |    47648    22717     47 | 10.092 % |
|  57525103 |    1280     8192    29440 |    52413    24211     44 | 10.092 % |
===============================================================================
restarts              : 98302
conflicts             : 76070974       (7972 /sec)
decisions             : 83292761       (0.00 % random) (8729 /sec)
propagations          : 40060287405    (4198260 /sec)
conflict literals     : 4081479941     (60.49 % deleted)
Memory used           : 110.00 MB
CPU time              : 9542.12 s

INDETERMINATE
-----------------------------------
```

Problem Specification

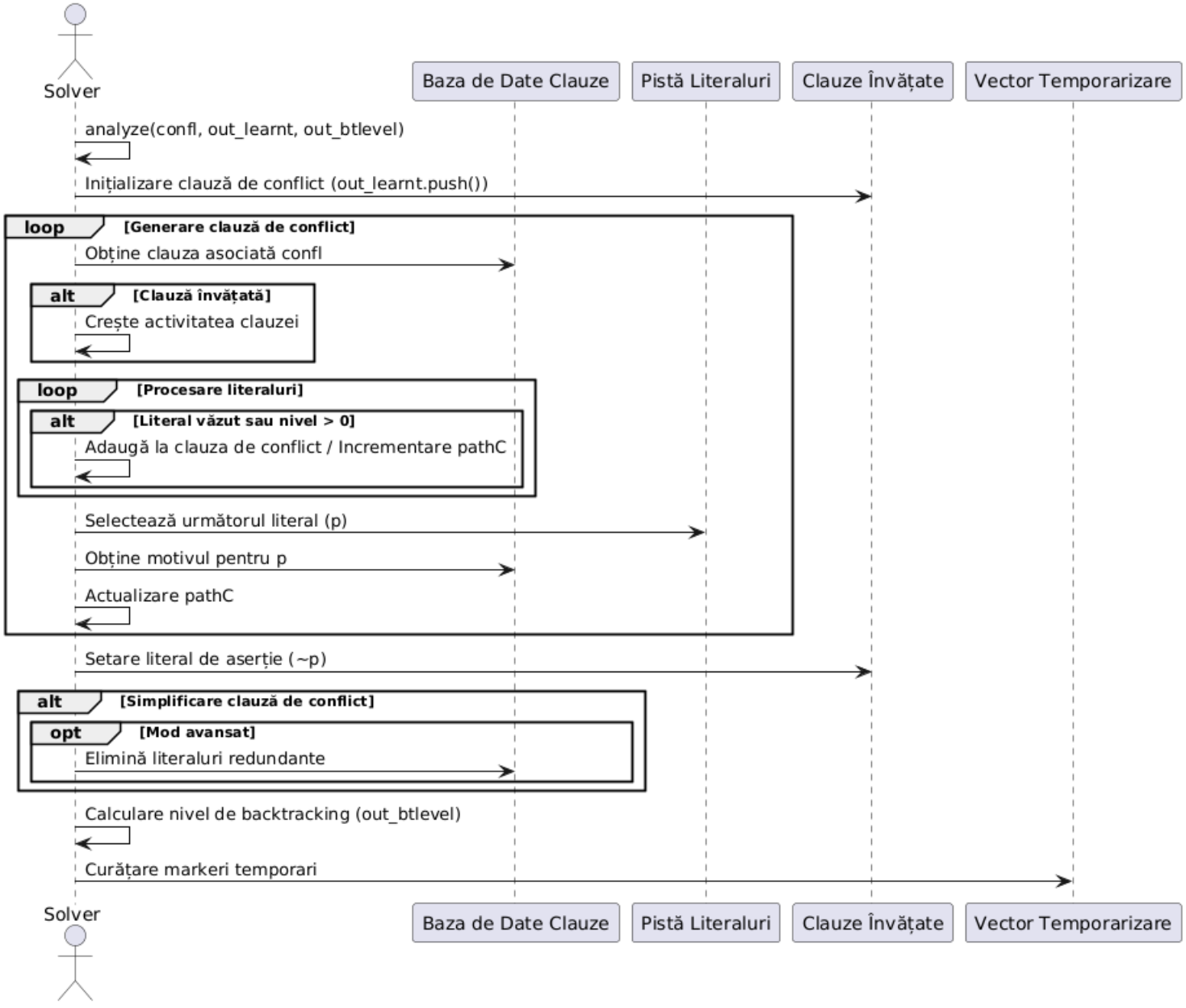
O problemă de tip SAT necesită folosirea unui algoritm combinat DPLL și CDCL. Algoritmul foloseste patru funcții: Search(), Propagate(), ReduceDB() și Analyze() pentru a verifica dacă problema este satisfiabilă.

Următoarele diagrame UML prezintă modul de funcționare a funcțiilor.

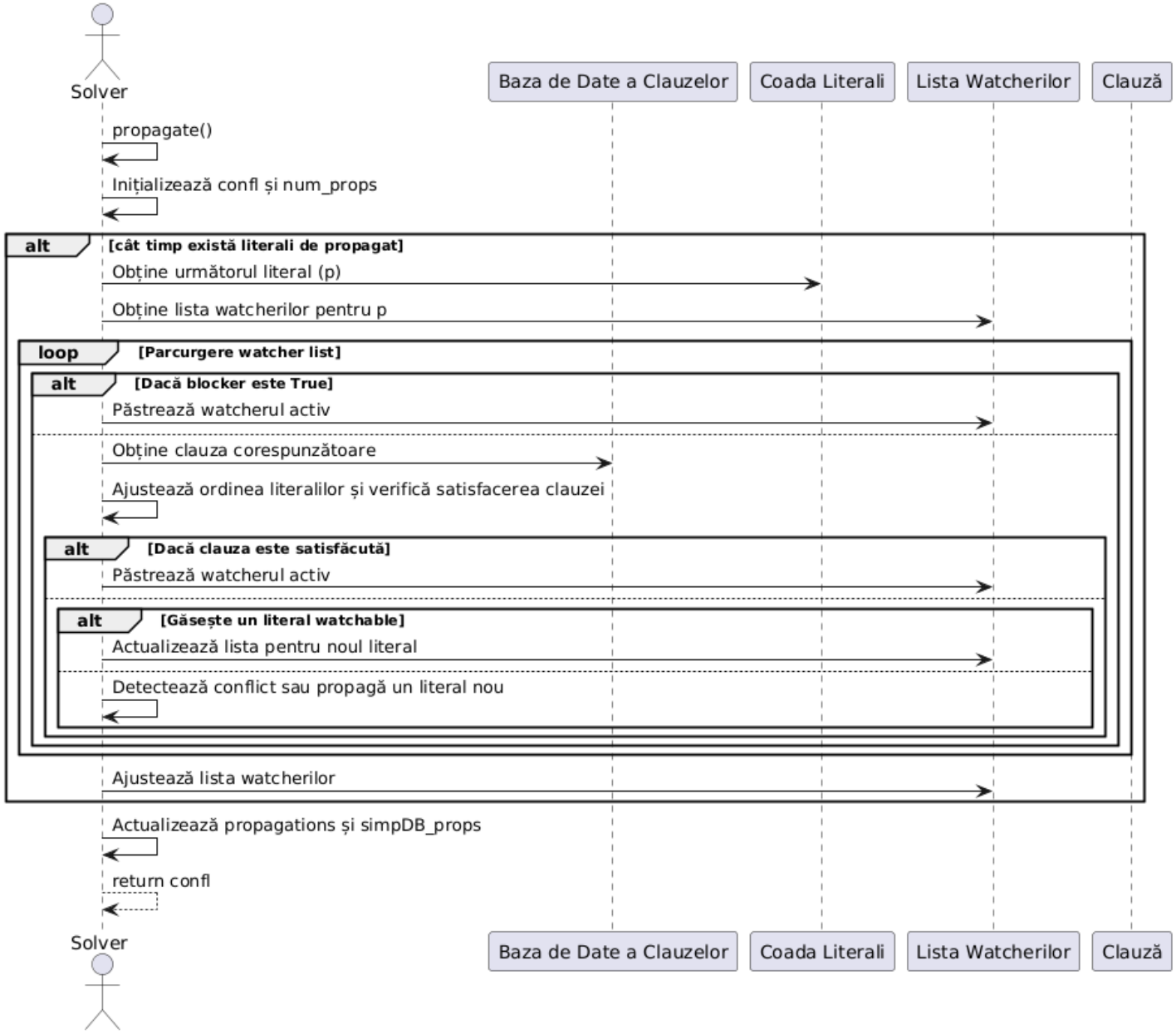
Search Diagram



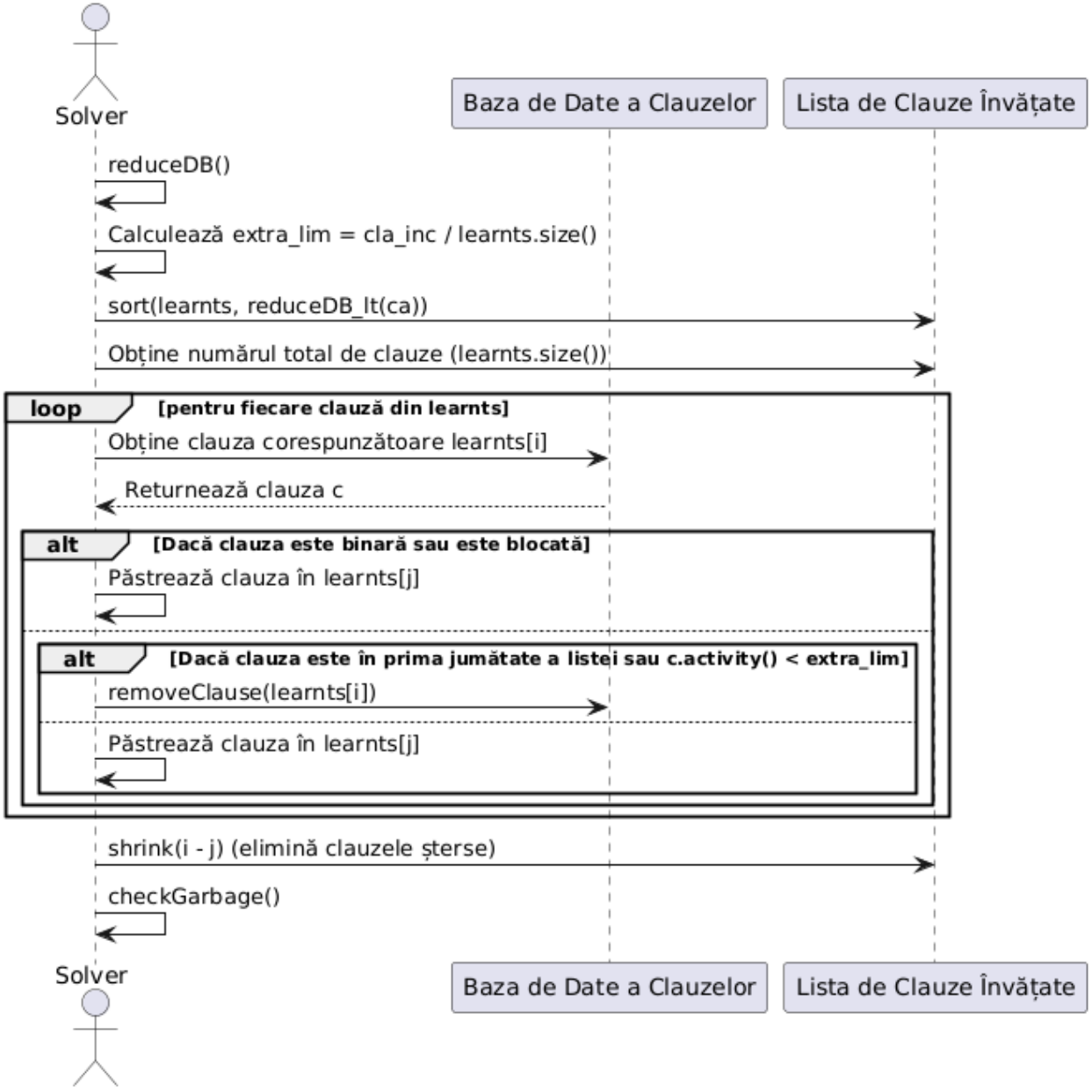
Analyze Diagram



Propagate Diagram



ReduceDB Diagram



Doxygen

Doxygen este un program folosit în generarea documentației pentru programe scrise în mai multe limbaje de programare, incluzând c și c++. Acest program extrage comentariile din cod si le salvează într-un document care are un anumit format. În cazul nostru, am folosit HTML și Latex

Solution

Moduri în care am putea îmbunătăți Minisatul includ:

1. Implementarea unei structuri de date mai eficiente, cum ar fi un hash map, pentru listele watch". Aceasta ar reduce timpul de căutare a literaliilor activi.
2. Curățarea bazei de date: Introducerea unor reguli dinamice pentru eliminarea clauzelor. De exemplu, clauzele care nu au fost utilizate recent ar putea fi prioritizate pentru ștergere
3. Învățarea clauzelor conflictuale: Reducerea dimensiunii clauzelor conflictuale pentru a economisi memorie și a accelera procesul de propagare.

Conclusion and Future Work

MiniSat este un instrument valoros pentru rezolvarea problemelor SAT. Analiza sa tehnică ne arată că, deși este foarte eficient, există loc de îmbunătățire, mai ales pentru problemele mari și complexe. Propunerile discutate astăzi pot contribui la optimizarea performanței acestuia.