

Razonamiento automatizado

Tarea 3

Profesora: Pilar Selene Linares Arévalo
Ayudante: Diego Carrillo Verduzco

Entrega: 20 de marzo de 2020

(4.5 pts) Algoritmo de saturación

Modifica la implementación del algoritmo de saturación con simplificación que creaste en la tarea 1, agregando las siguientes opciones (que pueden ser elegidas por el usuario) de ejecución:

1. Ejecución de algoritmo de saturación con simplificación original.
2. Ejecución de algoritmo de saturación con simplificación utilizando resolución positiva.
3. Ejecución de algoritmo de saturación con simplificación utilizando resolución unitaria.

(4.5 pts) DPLL

Crear un programa que decide si un conjunto de cláusulas es o no satisfacible utilizando DPLL. El programa debe encontrar todos los modelos existentes cuando el conjunto de entrada es satisfacible. Para esto hay dos alternativas:

- Implementar DPLL mediante el sistema de transiciones visto en clase.
- Implementar el algoritmo de DPLL tradicional visto en clase junto con el mecanismo para obtener todos los posibles modelos cuando el conjunto es satisfacible.

En ambos casos, implementar también una heurística para la elección de literales al aplicar RD ó `split`.

(3 pts) Refutaciones

Considera la siguiente afirmación:

Tree-like resolution exactly corresponds to the DPLL algorithm by the following well-known correspondence: the search tree produced by the run of a DPLL algorithm on an unsatisfiable formula F forms a tree-like resolution refutation of F , and from a given tree-like regular resolution refutation of F one can construct a run of a DPLL algorithm showing the unsatisfiability of F that produces essentially the given search tree.

Diseña una adecuación al algoritmo DPLL que satisfaga lo siguiente:

Dado un conjunto de cláusulas insatisfacibles el algoritmo genera una refutación en forma de árbol.

- Describe la adecuación al algoritmo y una justificación de la adecuación.
- Muestra la aplicación del algoritmo al siguiente conjunto:

$$\{p \rightarrow q, r \rightarrow s, p \vee r, \neg(q \wedge s), \neg(q \rightarrow p) \wedge (s \rightarrow r)\}$$