

Informe de Rendimiento

Camilo José Cruz Rivera (1428907), cruz.camilo@correounivalle.edu.co

- **Descripción del solver SAT que se utilizó**

El solver SAT elegido es “PalnleSS: A Framework for Parallel SAT Solvers”, el cual fue desarrollado por Ludovic LE FRIUX, mediante programación paralela utilizando el lenguaje de programación C++.

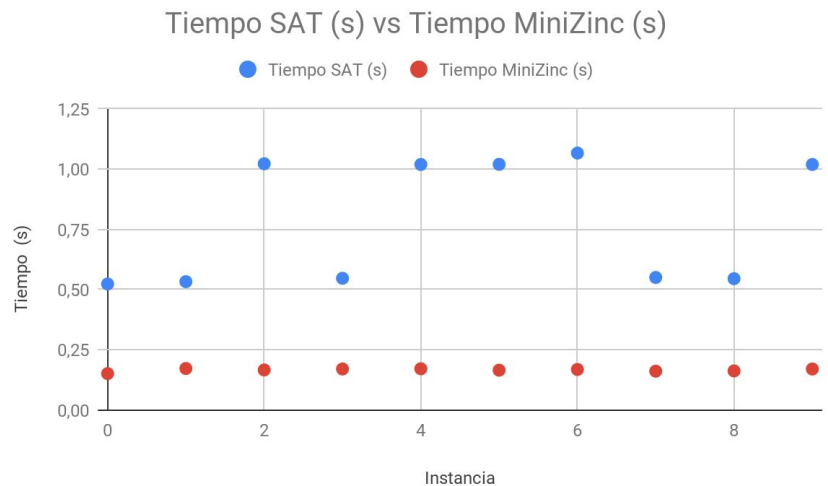
Este framework simplifica la implementación y evaluación de nuevos solucionadores de SAT paralelos para entornos de muchos núcleos, permitiendo instanciar independientemente cada componente para producir un nuevo solucionador completo. Este solver obtuvo el tercer lugar en la competencia SAT 2017, y el primer lugar en la competencia SAT 2018.

- **Descripción del solver en MiniZinc que se utilizó**

El solver elegido para en MiniZinc es “Gecode”. Este es un conjunto de herramientas desarrolladas en el lenguaje de programación C++, de código abierto para desarrollar sistemas y aplicaciones basados en restricciones. Gecode proporciona un solucionador de restricciones con un rendimiento de vanguardia a la vez que es modular y extensible.

- **Gráfico comparativo de los tiempos de ejecución entre el solver SAT y el solver escogido para MiniZinc.**

Instancia	Tiempo SAT (s)	Tiempo MiniZinc (s)
0	0,5235	0,152
1	0,5331	0,173
2	1,0219	0,167
3	0,5471	0,171
4	1,0189	0,172
5	1,0196	0,166
6	1,0663	0,169
7	0,5505	0,162
8	0,5455	0,163
9	1,0190	0,171



- **Análisis de los resultados observados**

Se observa que el tiempo de ejecución del solver MiniZinc es estable, sin importar la instancia a resolver, a diferencia del solver SAT que el tiempo varía entre instancias. Además se observa una diferencia de tiempo considerable entre el solver MiniZinc y el solver SAT, gastando este último, el doble de tiempo que el primero y en ocasiones aún más.