

Resolución de Sudoku mediante SAT

PROYECTO 1

MATTEO FERRANDO - 09-10285

ERICK DOS RAMOS - 07-40845

JOSÉ PONTE - 08-10893

El problema

El puzzle consta en una cuadrícula de 9x9 la cual a su vez se divide en 9 cuadrículas pequeñas de 3x3.

El objetivo es rellenar cada una de las casillas con números del 1 al 9 de tal manera que se cumplan cada una de las siguientes condiciones:

Dado un número x en el rango 1..9

1. x no puede aparecer dos veces en una misma fila de 9 casillas.
2. x no puede aparecer dos veces en una misma columna de 9 casillas.
3. x no puede aparecer dos veces en una misma mini cuadrícula de 3x3 de 9 casillas.

Representación

Para hacer uso de un resolucionador SAT se necesita representar el problema en función de variables CNF. Para ello se usa la solución de *Lynce & Ouaknine, 2009*.

En dicha representación, se usan 729 variables, donde cada una representa una casilla con un valor. De tal manera que xyz representa que en la fila x y columna y se encuentra el valor z . De manera análoga, $\neg xyz$ representa que en la fila x y columna y no se encuentra el valor z .

Cláusulas

Según *Lynce & Ouaknine* existen dos representaciones para las cláusulas, de las cuales una es redundante sobre la otra. La versión mínima, contando con 8829 cláusulas, consta de las siguientes reglas:

- Cada casilla tiene por lo menos un número.
- Cada número aparece sólo una vez en cada fila.
- Cada número aparece sólo una vez en cada columna.

- Cada número aparece sólo una vez en cada cuadrilla 3x3.
- Cada número aparece por lo menos una vez en una cuadrilla 3x3.

Mientras que la versión extendida, que aumenta el número de cláusulas a 11988 cuenta con las siguientes reglas redundantes:

- Hay a lo sumo un número en cada casilla.
- Cada número aparece por lo menos una vez en una fila.
- Cada número aparece por lo menos una vez en una columna.

Resolucionador SAT

Para solucionar las instancias de Sudoku, se hace uso del resolucionador de SAT *minisat*, de *Niklas Sorensson, 2007*.

Resultados experimentales

Para los experimentos se usaron 3 conjuntos de instancias las cuales se describen a continuación:

- 95 instancias consideradas difíciles.
- 11 instancias consideradas las más difíciles.
- 10000 instancias aleatorias.

Sobre estos conjuntos de instancias se realizaron corridas del programa usando tanto la codificación mínima como la extendida. A continuación los resultados:

95 instancias consideradas difíciles		
Codificación	Promedio (seg)	Desviación estándar (seg)
Mínima	0.01697420572	0.04042029507
Extendida	0.02214772325	0.0197511312

10000 instancias aleatorias		
Codificación	Promedio (seg)	Desviación estándar (seg)
Minima	0.01664685256	0.01508405552
Extendida	0.02117171717	0.01944773875

11 instancias consideradas las más difíciles		
Codificación	Promedio (seg)	Desviación estándar (seg)
Minima	0.02648839084	0.04042029507
Extendida	0.01683092117	0.004581485279

Conclusiones

Tomando en consideración los resultados obtenidos en los 3 conjuntos de instancias, se puede concluir con evidencia estadística que, para casos sencillos, la codificación mínima resuelve los problemas de manera más rápida. Mientras que para los casos más complicados la codificación extendida provee más información con lo cual puede llegar más rápido a las soluciones. Estos resultados coinciden con los arrojados por *Lynce* & *Ouaknine* respecto a dichas codificaciones.