

1 Introduction et méthode

Dans ce rapport, nous allons rapporter les performances de différentes versions des solveurs *SAT4J*, *MiniSAT* et *CryptoMiniSAT*. Pour ce faire, nous utiliserons la base de donnée de formules booléennes trouvable à cet url : <https://github.com/diverse-project/samplingfm/tree/master/Benchmarks>. Plus précisément, nous utiliserons uniquement les fichiers .cnf qui sont dans le répertoire "Benchmarks" de cette base de données.

Les solveurs testés sont :

- SAT4J - version 2.3.1 - JAR exécutable - paramètres par défaut
- _____
- MiniSAT - version 1.14.0 - binaire - paramètres par défaut
- MiniSAT - version 2.2.0 - binaire - -rnd-freq=0 (Défaut)
- MiniSAT - version 2.2.0 - binaire - -rnd-freq=0.5
- MiniSAT - version 2.2.0 - binaire - -rnd-freq=0.9
- _____
- CryptoMiniSAT - version 2.4.0 - binaire - paramètres par défaut
- CryptoMiniSAT - version 3.1.0 - binaire - paramètres par défaut
- CryptoMiniSAT - version 4.5.3 - binaire - paramètres par défaut
- CryptoMiniSAT - version 5.6.8 - binaire - -freq=0 (Défaut)
- CryptoMiniSAT - version 5.6.8 - binaire - -freq=0.5
- CryptoMiniSAT - version 5.6.8 - binaire - -freq=0.9

Afin de comparer ces solveurs, nous récupérerons leur résultat et leur temps d'exécution sur toutes les formules du dossier "Benchmarks", avec un timeout à 10 minutes. Afin d'obtenir des résultats les moins bruités possible, nous effectuons ces calculs plusieurs fois et utilisons la moyenne.

Les tests sont lancés depuis un projet XText. Pour reproduire les données présentées dans ce rapport, il suffit d'exécuter le fichier "org.xtext.example.msat.tests/src/org/xtext/example/msat/theos/Mein.xtext" avec JUnit. Les résultats seront stockés dans les fichiers "result_*.csv".

En tout les benchmarks ont durés 20 heures soit six itérations par tests.

2 Résultats

Les données brutes peuvent être trouvées dans le dossier "results", avec "results_*.csv" les données de chaque exécution de la base de données.

2.1 Aspect fonctionnel

Tous les solveurs ont renvoyé la même réponse pour toutes les formules présentes dans "Benchmarks".

A noter que dans les cas où les solveurs ont timeout il se pourrait qu'il existe en réalité un bug mais que les conditions de tests ne nous aient pas permis de le découvrir. Par exemple, pour la formule *listReverse.sk_11_43*, le solveur Sat4J a timeout systématiquement.

Solveur	Version	Paramètres	Premier	Temps moyen	Timeout
SAT4J-Jar	2.3.1		0	35.71311904761907	8
MiniSAT	1.14.0		0	30.90854285714287	8
MiniSAT	2.2.0	-rdn-freq=0	5	28.379771428571427	8
MiniSAT	2.2.0	-rdn-freq=0.5	8	21.300999999999998	4
MiniSAT	2.2.0	-rdn-freq=0.9	1	28.466257142857135	8
CryptoMiniSAT	2.4.0		0	24.241333333333337	5
CryptoMiniSAT	4.5.3		3	4.498499999999998	0
CryptoMiniSAT	5.6.8	-freq=0	3	10.498333333333331	0
CryptoMiniSAT	5.6.8	-freq=0.5	1	20.116500000000006	4
CryptoMiniSAT	5.6.8	-freq=0.9	1	20.638809523809538	4

Table 1: Resultats sur les 70 formules de "Benchmarks"

2.2 Efficacité

Afin de comparer l'efficacité, nous ne parlerons ici que du temps d'exécution, sans compter d'autres facteurs tels que l'espace utilisé.

La table 1 récapitule les résultats que nous avons trouvé pertinent de mettre en avant. La colonne "Premier" indique combien de fois le solveur a fini avant les autres. Le total ne s'additionne pas à 70 car nous n'avons pas pris en compte les formules que tous les solveurs triaient en moins de 0.01s, car le temps est trop court pour être pertinent. La colonne "Timeout" indique le nombre de timeout effectué par chacun de ces solveurs, sur les 70 * 6 formules traitées.

3 Questions

3.1 Le "meilleur" solveur ?

Le temps moyen de résolution pour toutes les formules est trouvable dans la table 1.

On remarque que CryptoMiniSAT 4 n'a été battu que sur ??? formules. Et seulement de ???.

Les SAT-solveurs modernes utilisent des heuristiques qui peuvent légèrement varier et, pour des instances particulières, mener à des écarts de performances plus ou moins importants. De fait, la question du "meilleurs" solveur n'a pas tant de sens sans ses guillemets. Cependant dans notre cas, au vu des informations à notre disposition, la réponse n'appelle pas particulièrement à la nuance :

Sur cet ensemble de tests, CryptoMiniSAT 4 domine largement le reste des solveurs. Il est d'ailleurs étonnant de constater que la version 4 a de bien meilleurs résultats que la version 5. Une explication possible est que la version 5 a été optimisée pour de très grandes instances et que notre timeout de 10 minutes (ainsi que notre ensemble de tests limité) ne permet pas de mettre en valeur ces cas de figures.

3.2 L'existence de bugs

Are there functional bugs in some solvers? An immediate approach is to control whether the solvers return the same result for the same formula (we hope so!). You can use the benchmark or generate random formulae Are there performance deviations of some solvers? Significant deviations may suggest performance-related bugs in some solvers. Is it the case?

Pour toutes les exécutions n'ayant pas timeout, les résultats des solveurs étaient cohérents entre eux.

Il est cependant probable que les erreurs auraient des tendances à apparaître pour les formules "difficiles". Il serait donc intéressant de refaire les tests avec un timeout plus grand.

3.3 La difficulté des formules

Are there harder SAT formulae?

3.4 De meilleurs benchmark ?

Can we reduce a benchmark and only consider a subset? (open question) Numerous formulae are used, but only a few lead to performance variations. Can't we only use a subse

3.5 Lol lol lol

Write a report that addresses the questions above: your answers should be supported by data (visualizations, statistics, tables, etc.). You should also include a technical description of your experiments (what solvers you use, what formulae, the measurements conditions, etc.), explain how to reproduce your results, and point out the data you have relied on. Push the report and all material in a subfolder (with a unique name) in the “reports” folder: repo.