

Sujet de thèse :

CloudSat : Vers un solveur SAT distribué efficace

Directeur Souheib Baarir (LIP6 - équipe Move)

1 Contexte : problèmes SAT et cloud computing

Avec l'intégration toujours plus grande, des systèmes informatiques dans des applications critiques (métro, voitures, centrales nucléaires, blocs opératoires ...), assurer la correction des programmes devient une nécessité. Si l'utilisation systématique de tests permet d'améliorer grandement la qualité des codes, elle ne peut réellement garantir leur correction.

À l'inverse, les techniques dites de *model checking*, qui vérifient la validité du workflow applicatif (modélisé par un graphe d'états) par rapport à une propriété donnée, permettent de garantir la validité des programmes. Ces techniques reposent, essentiellement, sur deux théories : celle basée sur les automates et celle fondée sur la satisfiabilité booléenne (SAT). Dans les deux théories, l'ennemi commun est le problème bien connu de l'explosion combinatoire. En utilisant les automates, le problème se manifeste par une consommation mémoire démesurée, alors qu'en utilisant le SAT, le problème se manifeste sous la forme d'un temps de calcul qui peut être extrêmement long.

Or, le cloud computing rend aujourd'hui accessible une puissance de calcul autrefois réservé aux supercalculateurs. Si les machines de ces systèmes sont toutes dotées de multicœurs et d'une grande capacité de mémoire, la scalabilité passe aussi par l'utilisation en parallèle d'un grand nombre de machines. L'utilisation d'une telle architecture nécessite donc l'utilisation d'un solveur SAT multithreadé, mais aussi distribué.

2 Objectif : distribution de solveurs SAT

La distribution de solveurs SAT est donc aujourd'hui un enjeu majeur, comme en témoigne la création pour la première fois d'une catégorie dédiée à la compétition internationale des solveurs SAT, avec le soutien d'Amazon EC2.

La conception de tels solveurs pourra s'inspirer des nombreux travaux menés par la communauté ces dernières années pour paralléliser la résolution des problèmes SAT afin d'exploiter au mieux les architectures multicœurs. Cependant elle pose de nouveaux problèmes en effet le coût de la communication sera bien plus important qu'en mémoire partagée. Les algorithmes devront donc être repensés pour optimiser les informations échangées entre les différentes instances. Et dans ce domaine tout reste à faire, comme le montrent les résultats de la compétition où, malgré l'utilisation de 100 machines, les meilleurs participants n'ont été qu'à peine plus performants que les solveurs parallèles multicœurs.

L'objectif de cette thèse est donc de concevoir et implémenter un solveur SAT distribué réellement efficace. Pour y arriver, le doctorant devra non seulement repenser les algorithmes du solveur, mais aussi attacher une très grande importance à l'efficacité de la distribution, tout en gardant à l'esprit que chaque nœud exécute un solveur multithreadé. Les problèmes de synchronisation, tant locales que distants, seront donc au cœur de cette thèse.

3 Plateforme : intégration dans *Painless*

Cette thèse s’inscrit dans la continuité directe de travaux menés au LIP6 depuis 5 ans sur la parallélisation de solveurs SAT [4, 3, 2, 1]. Il pourra donc s’appuyer sur Painless (<http://painless.lrde.epita.fr/>) : un solveur SAT parallèle à l’architecture générique, modulaire et facilement instanciable, pour les systèmes multicœurs. Entièrement développé en interne, Painless est depuis 2018 le meilleur solveur SAT parallèle et est aujourd’hui utilisé comme base par de nombreuses équipes de recherche à travers le monde, comme le montrent les résultats de la compétition où 7 des 10 meilleurs solveurs utilisaient Painless (<https://satcompetition.github.io/2020/results.html>).

L’objectif final de la thèse est donc d’offrir à la communauté une nouvelle version de Painless intégrant la distribution. Partant de la version parallèle, le doctorant pourra tester sa solution sur l’un des clusters du laboratoire et sur les machines de la plateforme Grid5000. En cas de succès, il sera amené à participer à la prochaine compétition internationale dans la catégorie des solveurs distribués.

Références

- [1] L. L. Frioux, S. Baarir, J. Sopena, and F. Kordon. Painless : A framework for parallel SAT solving. In S. Gaspers and T. Walsh, editors, *Theory and Applications of Satisfiability Testing - SAT 2017 - 20th International Conference, Melbourne, VIC, Australia, August 28 - September 1, 2017, Proceedings*, volume 10491 of *Lecture Notes in Computer Science*, pages 233–250. Springer, 2017.
- [2] L. L. Frioux, S. Baarir, J. Sopena, and F. Kordon. Modular and efficient divide-and-conquer SAT solver on top of the painless framework. In T. Vojnar and L. Zhang, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 25th International Conference, TACAS 2019, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2019, Prague, Czech Republic, April 6-11, 2019, Proceedings, Part I*, volume 11427 of *Lecture Notes in Computer Science*, pages 135–151. Springer, 2019.
- [3] V. Vallade, L. L. Frioux, S. Baarir, J. Sopena, V. Ganesh, and F. Kordon. Community and lbd-based clause sharing policy for parallel SAT solving. In L. Pulina and M. Seidl, editors, *Theory and Applications of Satisfiability Testing - SAT 2020 - 23rd International Conference, Alghero, Italy, July 3-10, 2020, Proceedings*, volume 12178 of *Lecture Notes in Computer Science*, pages 11–27. Springer, 2020.
- [4] V. Vallade, L. L. Frioux, S. Baarir, J. Sopena, and F. Kordon. On the usefulness of clause strengthening in parallel SAT solving. In R. Lee, S. Jha, and A. Mavridou, editors, *NASA Formal Methods - 12th International Symposium, NFM 2020, Moffett Field, CA, USA, May 11-15, 2020, Proceedings*, volume 12229 of *Lecture Notes in Computer Science*, pages 222–229. Springer, 2020.