



Faire face aux erreurs dans les super-ordinateurs

Aurélien Cavelan¹

Aurélien Cavelan a soutenu sa thèse² en juillet 2017 à l'université de Lyon, thèse opérée par l'École normale supérieure de Lyon, et préparée au Laboratoire de l'informatique du parallélisme (LIP) et au centre Inria Grenoble Rhône-Alpes (équipe ROMA), sous la direction de Anne Benoit et Yves Robert. Il effectue actuellement un stage postdoctoral à l'université de Bâle, en Suisse.



La liste du TOP500³, actualisée tous les six mois, recense les ordinateurs les plus puissants dans le monde : on parle alors de « super-ordinateurs ». Ces monstres de calcul comptent plusieurs dizaines de milliers de processeurs et sont indispensables aujourd'hui dans beaucoup de domaines notamment scientifiques : simuler la naissance de l'univers, chercher de nouveaux vaccins ou encore prévoir la météo.

La puissance de calcul de ces machines a été multipliée par 10 000 en vingt ans, et on approche aujourd'hui la barre symbolique de l'exaflop, soit un milliard de milliards d'opérations par seconde (voir Figure 1).

D'un point de vue purement logistique, il suffirait de multiplier le nombre de processeurs par 100 pour atteindre cette performance. Après tout, c'est une stratégie qui

1. http://hpc.dmi.unibas.ch/HPC/Aurelien_Cavelan.html

2. « Algorithmes d'ordonnancement et schémas de résilience pour les pannes et les erreurs silencieuses », <http://www.theses.fr/2017LYSEN031>.

3. <https://www.top500.org/lists/>

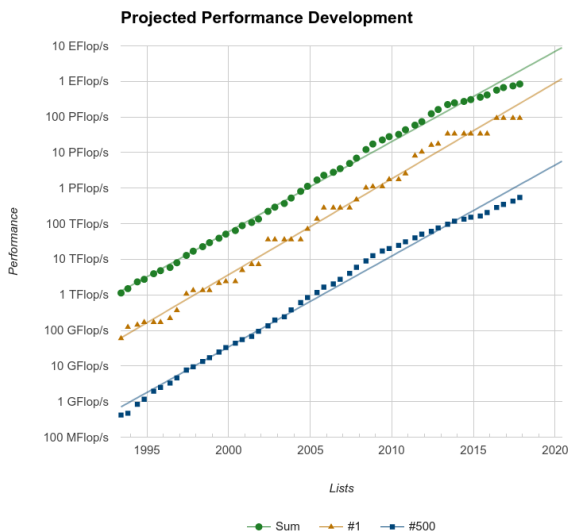


FIGURE 1. Performances des super-ordinateurs du TOP500 de 1992 à 2018.

a bien fonctionné ces dix dernières années. Mais le problème est ailleurs. En fait, la fiabilité d'un système est directement proportionnelle au nombre de ses composants. Plus il y a de processeurs, plus la probabilité que le système subisse une panne pendant l'exécution d'une application est élevée. Aujourd'hui, ce n'est pas moins d'une à deux pannes par jour qu'on observe sur les plus imposants, comme Titan2 et ses 36000 processeurs.

Si une panne par jour semble déjà élevé, on estime qu'il pourrait y en avoir 10 à 100 fois plus dans la prochaine génération de super-ordinateurs. En février 2014, la *résilience*, c'est-à-dire la capacité à récupérer rapidement d'erreurs ou de pannes, a été identifiée comme l'un des principaux défis à relever pour le passage à l'Exascale [8], et c'est dans ce contexte que s'inscrit ma thèse [4].

L'approche classique pour faire face aux pannes est l'utilisation de points de sauvegarde [5, 7]. L'application sauvegarde périodiquement le travail sur un support stable. Ainsi, en cas de panne, il est possible de redémarrer l'application à partir du dernier point de sauvegarde, sans avoir à tout recalculer depuis le début. Il y a un compromis évident entre le temps passé à sauvegarder, et le temps perdu en cas de ré-exécutions à cause des pannes. Comme le temps de calcul est précieux, on cherche



FIGURE 2. Titan est un super-ordinateur installé par Cray au laboratoire de Oak Ridge (USA) et est utilisé pour différents projets scientifiques.

à maximiser l'utilisation du système et à minimiser le temps d'exécution pour une application.

Dans ma thèse, je propose des modèles qui me permettent d'estimer le temps d'exécution d'une application en fonction de plusieurs paramètres comme le temps de sauvegarde ou le nombre de pannes moyen pour un système. Grâce à ces modèles on peut déterminer, par exemple, la période optimale de sauvegarde. Un résultat bien connu qui va dans ce sens est la formule dérivée par Young et Daly qui nous donne la période optimale de sauvegarde suivante [9, 6] :

$$(1) \quad T = \sqrt{2\mu S}$$

en fonction du temps de sauvegarde S et du temps moyen entre deux pannes μ , pour une application et un système donné. Cette formule très simple peut être utilisée pour un grand nombre d'applications. Dans ma thèse, je propose plusieurs extensions de ce résultat, par exemple lorsqu'il existe non pas un seul, mais plusieurs types de sauvegarde possibles [2].

En plus des pannes, les utilisateurs de ces systèmes doivent maintenant faire face à une nouvelle menace : les erreurs silencieuses. La cause de ces erreurs peut paraître surprenante : ce sont en fait des rayons cosmiques ou plus précisément des particules chargées en énergie qui traversent notre atmosphère et qui, de temps en temps, interagissent avec la mémoire des composants électroniques. Il arrive alors qu'un « bit » passe d'un « 1 » à un « 0 » (ou vice versa), modifiant ainsi le contenu de la mémoire à cet endroit. En fait, ces super-ordinateurs, à cause de leur taille, agissent comme

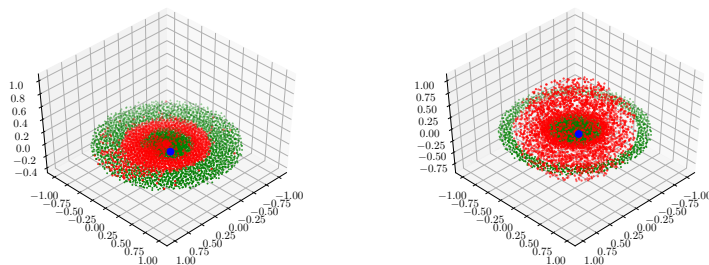


FIGURE 3. Propagation d’une erreur silencieuse dans une simulation de particules représentant l’effondrement d’un nuage de gaz sur lui-même (coupe transversale). À chaque fois, une seule particule est affectée (point bleu), mais l’erreur se propage très rapidement aux particules voisines, menant à des résultats bien différents de ceux attendus (en rouge).

de véritables détecteurs de particules. En 2010, Jaguar, alors le plus puissant super-ordinateur (qui deviendra plus tard Titan), enregistrait pas moins de 350 erreurs de ce type par minute⁴ !

Même si des mécanismes existent pour détecter ces erreurs au niveau du matériel, toutes les erreurs ne peuvent pas être détectées. Contrairement aux pannes dont la détection est immédiate, une erreur silencieuse n’est identifiée que lorsque les données corrompues conduisent l’application à se comporter étrangement. Une telle latence de détection soulève un nouveau défi : comment les détecter et surtout comment garantir que les résultats obtenus sont corrects ?

Malheureusement, de manière générale, il n’existe aucune solution à la fois efficace et peu coûteuse pour détecter ces erreurs. Dans ma thèse, je propose plusieurs modèles pour comparer différentes méthodes de détection [1], et les combiner à des mécanismes de sauvegarde [3]. Plus récemment, je me suis aussi intéressé à la propagation des erreurs silencieuses dans les simulations de particules (voir Figure 3) dans le but de développer de nouvelles méthodes de détection.

Références

- [1] L. Bautista-Gomez, A. Benoit, A. Cavelan, S. K. Raina, Y. Robert, and H. Sun. Coping with recall and precision of soft error detectors. *Journal of Parallel and Distributed Computing*, 98 :8–24, 2016.

4. <https://spectrum.ieee.org/computing/hardware/how-to-kill-a-supercomputer-dirty-power-cosmic-rays-and-bad-solder>

- [2] A. Benoit, A. Cavelan, V. Le Fèvre, Y. Robert, and H. Sun. Towards optimal multi-level checkpointing. *IEEE Transactions on Computers*, 2016.
- [3] A. Benoit, A. Cavelan, Y. Robert, and H. Sun. Optimal resilience patterns to cope with fail-stop and silent errors. In *International Parallel and Distributed Processing Symposium (IPDPS)*, pages 202–211. IEEE, 2016.
- [4] A. Cavelan. *Scheduling algorithms and resilience patterns for fail-stop and silent errors. (Algorithmes d'ordonnancement et schémas de résilience pour les pannes et les erreurs silencieuses)*. PhD thesis, École Normale Supérieure de Lyon, France, 2017.
- [5] K. M. Chandy and L. Lamport. Distributed snapshots : Determining global states of distributed systems. *ACM Transactions on Computer Systems*, 3(1) :63–75, 1985.
- [6] J. T. Daly. A higher order estimate of the optimum checkpoint interval for restart dumps. *Future Generation Comp. Syst.*, 22(3) :303–312, 2006.
- [7] E. N. Elnozahy, L. Alvisi, Y.-M. Wang, and D. B. Johnson. A survey of rollback-recovery protocols in message-passing systems. *ACM Computing Survey*, 34 :375–408, 2002.
- [8] R. Lucas, J. Ang, K. Bergman, S. Borkar, W. Carlson, L. Carrington, G. Chiu, R. Colwell, W. Dally, J. Dongarra, et al. Top ten exascale research challenges. *DOE ASCAC subcommittee report*, pages 1–86, 2014.
- [9] J. W. Young. A first order approximation to the optimum checkpoint interval. *Comm. of the ACM*, 17(9) :530–531, 1974.