

Un retour d'expérience sur la reproductibilité et la comparaison de résultats expérimentaux

Boris Detienne¹, François Clautiaux¹, Laurent Facq¹, Gaël Guillot¹

¹ Université de Bordeaux, 351 Cours de la Libération, 33400 Talence

boris.detienne@u-bordeaux.fr

francois.clautiaux@u-bordeaux.fr

laurent.facq@math.u-bordeaux.fr

gael.guillot@u-bordeaux.fr

Applications de la RO (santé, bio-informatique, économie, énergie, développement durable, cloud computing, gestion du risque, réseaux sociaux...), transfert vers l'industrie et beaux logiciels (APP)

Sur les meilleures pratiques de programmation en RO

Mots-clés : *Reproductibilité, analyse de performance.*

La reproductibilité des résultats est un élément crucial des sciences expérimentales, notamment des mathématiques appliquées. En recherche opérationnelle, les comparaisons numériques sont un élément majeur de la démarche scientifique. Les chercheurs sont amenés à effectuer des comparaisons entre leurs résultats (numériques ou temps de calcul) et ceux de la littérature sur divers problèmes. Cette comparaison peut s'avérer complexe si les machines, langages, compilateurs, solveurs... sont différents. Par exemple, il peut paraître simple de comparer deux processeurs ayant un unique cœur pour un programme séquentiel, mais si le programme est parallélisé, et/ou le nombre de cœurs est différent et/ou si un seul thread est utilisé sur une machine multi-cœurs, le problème devient plus complexe.

Le but de cette présentation est d'apporter un retour d'expérience et de montrer les difficultés que nous avons rencontrées sur deux applications distinctes.

La première situation est un cas où le code auquel on souhaite se comparer n'est pas disponible. Les expérimentations ne peuvent donc pas être faites sur la même machine. Nous montrerons une comparaison de deux machines différents sur le problème de sac-à-dos temporel. Ce problème généralise le problème standard de sac à dos en considérant des objets n'occupant la capacité que durant certains intervalles de temps fixés. Notre problématique de comparer les temps de calcul de nos méthodes à celles d'une équipe dont les résultats sont disponibles, mais pas les codes. Pour ce faire, nous nous appuierons sur des comparaisons de type benchmarks (SPEC...) et sur des comparaisons de temps observés sur ces machines.

La deuxième situation est un cas où le code auquel on souhaite se comparer est disponible mais pose plusieurs problèmes : langage différent, méthode plus ou moins générique, difficultés de reproductibilité du code... Pour illustrer ce cas, nous comparons des résultats

obtenus sur un problème classique d'ordonnancement à une machine sans préemption où l'objectif est de minimiser la somme des retards pondéré par le poids des tâches.