Utilisation de compression Block Low-Rank multiprécision pour résoudre des systèmes linéaires

Patrick Amestoy[1], Olivier Boiteau[2], Alfredo Buttari[3], Matthieu Gerest*[4,5], Fabienne Jézéquel[6,7], Jean-Yves L'excellent[1], and Theo Mary[8]

[1]Mumps Technologies – Mumps Technologies – France
[2]EDF Lab Paris-Saclay – EDF Recherche et Développement – France
[3]Institut de recherche en informatique de Toulouse (IRIT) – CNRS : UMR5505 – 118 Route de Narbonne, F-31062 Toulouse Cedex 9, France
[4]LIP6 – Sorbonne Universite, Centre National de la Recherche Scientifique : UMR7606 – France
[5]EDF Lab Paris-Saclay – EDF Recherche et Développement – France
[6]Université Panthéon-Assas – université Paris 2, Panthéon-Assas – France
[7]LIP6 – Sorbonne Universite, Centre National de la Recherche Scientifique : UMR7606 – France
[8]LIP6 – Sorbonne Université, CNRS, LIP6 – France

Résumé

Dans de nombreuses applications apparaissent des matrices dites Block Low-Rank (BLR), pour lesquelles les blocs éloignés de la diagonale ont un rang numérique faible. On peut donc stocker ces derniers sous la forme d'une approximation de rang faible telle qu'une SVD tronquée. Cette compression permet de réduire à la fois le coût de stockage la matrice ainsi que le temps de calcul de sa factorisation LU, tout en contrôlant l'erreur introduite à l'aide d'un paramètre ϵ. Dans cet exposé, une nouvelle variante de cette compression BLR est présentée, utilisant à bon escient plusieurs formats de précision pour représenter les coefficients. La plupart des entrées peuvent en effet être converties en précision faible, telles que les simple et demie précisions, au lieu de la double précision plus couramment utilisée. En effet, une analyse théorique ainsi que des expériences numériques permettent de justifier que l'erreur de compression reste ainsi du même ordre de grandeur que l'erreur de compression initiale. Nous illustrons l'intérêt de cette approche en précision mixte sur plusieurs matrices provenant d'applications réalistes, et montrons que l'utilisation combinée de trois arithmétiques binary64, binary32 et bfloat16 permet d'obtenir une réduction des coûts en stockage et en nombre d'opérations allant jusqu'à un facteur trois, le tout sans perte de précision significative.

*Intervenant