Calcul du rang de grandes matrices creuses modulo p par des méthodes d'élimination.

Charles Bouillaguet* **Claire Delaplace***†

* Université de Lille, CRIStAL
† Université de Rennes 1, Irisa

Soit p un nombre premier et $M \in \mathbb{F}_p^{n \times m}$ une matrice creuse dont on cherche à calculer le rang r. Il existe essentiellement deux grandes familles d'algorithmes pour calculer les opérations usuelles (rang, déterminant, solution d'un système linéaire) sur M : les méthodes itératives et les méthodes directes.

Les méthodes itératives, telles que l'algorithme de Wiedemann [6], fonctionnent en calculant une succession de produits matrice-vecteur, où la matrice de départ M n'est jamais modifiée. Retrouver le rang r de M nécessite généralement $\mathcal{O}(r)$ produits matrice-vecteur, dont la complexité est proportionnelle au nombre d'entrées dans la matrice. Le temps d'exécution de ces méthodes est donc prévisible, et leur avantage principal est que leur complexité en mémoire est faible.

Les méthodes directes, telles que la méthode du pivot de Gauss, sont très répandues dans le monde du calcul numérique. Leur principe est d'éliminer certaines entrées de la matrice de départ pour la mettre sous forme triangulaire. Le problème est que ce processus d'élimination produit souvent du « remplissage » ; c'est à dire que des entrées non-nulles apparaissent là où il n'y en avait initialement pas. Cela peut ralentir considérablement le processus et, dans certains cas, cela peut même conduire à un crash de mémoire. La complexité de ces algorithmes est donc assez difficile à prédire et la mémoire est souvent le facteur limitant.

En combinant des heuristiques de sélection de pivots utilisées pour le traitement de matrices issues de calcul de bases de Gröbner [4] et l'algorithme d'élimination GPLU [5] développé par Gilbert et Peierls, nous avons mis au point un nouvel algorithme [1] qui effectue une variante de la méthode du pivot de Gauss en réduisant le remplissage. Nous avons implémenté cet algorithme et nous l'avons comparé à l'algorithme d'élimination présent dans Linbox [3], ainsi qu'à une implémentation de GPLU adaptée aux corps finis. Les résultats obtenus montrent que notre nouvel algorithme est plus rapide que ces algorithmes sur les matrices de la collection SIMC de Jean-Guillaume Dumas [2], et dans certains cas, il est également plus rapide que la méthode itérative de Wiedemann. Les récentes modifications que nous avons apportées - notamment une heuristique de sélection de pivots plus élaborée - nous ont permis d'obtenir des résultats encore meilleurs.

Dans cet exposé, je présenterai cet algorithme et les améliorations qui y ont été apporté. J'illustrerai son fonctionnement sur quelques exemples.

Références

[1] C. Bouillaguet, C. Delaplace : Sparse Gaussian Elimination modulo p : An update. Proceeding of the 18th International Workshop of Computer Algebra in Scientific Computing, Bucharest, Romania, pp. 101-116, (2016).

[2] J.-G. Dumas : Sparse Integer Matrix Collection, `http://hpac.imag.fr`

[3] J.-G. Dumas, G. Villard : Computing the Rank of Sparse Matrices over Finite Fields. Proceeding of the 5th International Workshop of Computer Algebra in Scientific Computing, Yalta, Ukraine, pp. 47-62, (2002).

[4] J.-C. Faugère, S. Lachartre : Parallel Gaussian Elimination for Gröbner Bases Computations in Finite Fields. In PASCO. pp. 89-97, ACM, (2010).

[5] J. R. Gilbert, T. Peierls : Sparse Partial Pivoting in Time Proportional to Arithmetic Operations. SIAM Journal on Scientific and Statistical Computing 9 No. 5, 862-874, (1988).

[6] D.H. Wiedemann : Solving sparse linear equations over finite fields. IEE Trans. Information Theory 32 No. 1, 54-62, (1986).