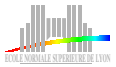
LinBox : une bibliothèque générique pour l'algèbre linéaire exacte

Pascal Giorgi

Laboratoire LIP - École Normale Supérieure de Lyon

LIP  - ENS Lyon  - CNRS  - INRIA  - UCBL 

Projet LinBox

- Projet international USA-Canada-France, fincancement NSF/CNRS
  31 chercheurs
  $\Longrightarrow$ algébre linéaire exacte

- Site web: *www.linalg.org*

- Bibliothèque générique C++, licence GPL, (80000 lignes de code)

Principaux développements:

- algorithmes (rang, systémes linéaire,...)
- matrices (boîtes noires, conteneurs)
- domaines de calcul (corps finis, entiers, rationnels)
- généricité (plug&play)

Multiplication de matrices sur un corps finis

- Opération hybride (exacte-numérique):
  Approche **FFLAS** package [Dumas-Gautier-Pernet 2002]
  - conversions éléments $\Longleftrightarrow$ flottants
  - opération numérique (BLAS)
  - conversions flottants $\Longleftrightarrow$ éléments

  Interêt:
  - Minimise le nombre de réductions modulaires
  - Avantage des routines numériques BLAS (bloc, optimisation de cache)

  $\Rightarrow$ 67.58s pour une multiplication d'ordre 5000 sur GF(101)

- LinBox: domaine générique au travers d'une interface matrice BLAS

Algorithmes sur les corps finis

Algèbre linéaire sur un corps

- Depuis 1969, multiplication matrices d'ordre n en moins que $O(n^3)$
  [Strassen 1969]: $O(n^{2.81})$
  ...

- Meilleurs algorithmes $\Longrightarrow$ multiplication matrices (complexité $O(n^\omega)$)
  [Strassen 1969]: inversion, systèmes linéaires, determinant
  [Bunch-Hopcroft 1974]: extension aux matrices non génériques
  [Ibarra-Moran-Hui 1982]: cas singulier: rang, noyau

- Cas creux
  [Wiedemann 1986], systémes linéaires en $O(n^2)$, probabiliste

Méthode à base d'élimination

Elimination de Gauss $\Rightarrow$ simplification de la matrice

- Gauss $\Leftrightarrow$ Décomposition LUP
- Algorithme de décomposition LSP [Ibarra-Moran-Hui 1982]
- Algorithme par bloc en $O(n^\omega)$ $\Longrightarrow$ { multiplication matrices ; résolution systèmes triangulaires }
  - pivot = matrice triangulaire
  - elimination = multiplication et addition matrices

Implantation via les bibliothèques numériques BLAS

- Multiplication matrice:
  → FFLAS package

- Systèmes linéaires triangulaires: algorithme bloc récursif

  Résolution hybride (exacte-numérique):
  recursif → solution représentable exactement sur les flottants
  résolution numérique via BLAS et conversion

FFPACK package [Dumas-Giorgi-Pernet 2004]
→ implantation efficace (en-place) LSP/LQUP
→ Intégration à LinBox (interface avec MAPLE)

Performance FFPACK package

- Décomposition LQUP sur GF(101)

n	400	700	1000	2000	3000	5000
LQUP	0.05s	0.18s	0.46s	2.80s	7.79s	32.9s
Fgemm	0.04s	0.23s	0.62s	4.28s	14.72s	67.58s
Ratio	1.25	0.78	0.74	0.65	0.53	0.48

Complexité arithmétique pour $\omega = 3$: LQUP = 1/3 * Multiplication matrices

- Inversion sur GF(101)

n	400	700	1000	2000	3000	5000
Inv	0.18s	0.70s	1.79s	10.84s	32.33s	139.5s
Fgemm	0.04s	0.23s	0.62s	4.28s	14.72s	67.58s
Ratio	4.50	3.04	2.89	2.53	2.20	2.07

Complexité arithmétique pour $\omega = 3$: Inverse = 4/3 * Multiplication matrices

Systèmes linéaires sur $\mathbb{Z}$

- Algorithme p-adique[Moenck et Carter 1979, Dixon 1982]:

- soient $A \in \mathbb{Z}^{n\times n}$, $b \in \mathbb{Z}^n$ trouver $x \in \mathbb{Q}^n / Ax = b$

  idée: calculer $Ay \equiv b \bmod p^{k+1}$, p premier

$$y = x_{[0]} + x_{[1]}p + x_{[2]}p^2 + \ldots + x_{[k]}p^k$$

- Reconstruction de $x \in \mathbb{Q}^n$: fractions continus y/p^{k+1}

- calcul itératif des chiffres p-adiques :
  $\rightarrow$ systèmes linéaire sur $\mathbb{Z}_p$

$$b_{[0]} \leftarrow b$$
pour i=0,1,2,...k
$$Ax_{[i]} \equiv b_{[i]} \bmod p$$
$$b_{[i+1]} \leftarrow \frac{b_{[i]} - A.x_{[i]}}{p}$$

- k choisit assez grand (Cramer, Hadamard)

implantation dans LinBox

- définition d'interface pour l'approximation (conteneur/itérateur)
- Résolution des systèmes sur $\mathbb{Z}_p$ via :
  méthode d'élimination (Decomposition LSP)
  méthode de Krylov (Wiedemann)
- optimisations :
  utilisation maximale: FFLAS et FFPACK
  construction approximation: pas de bébé / pas de géant
  reconstruction rationnelle: seulement les facteurs inconnus
- Comparaison avec la bibliothèque NTL

Performances: méthode d'elimination

- Systèmes avec coefficients 32bits

n	50	150	350	500	650	1000
LinBox	0.08s	0.96s	7.47s	18.57s	36.65s	115.43s
NTL	0.06s	1.92s	24.07s	72.49s	159.88s	688.57s

- Systèmes avec coefficients 128bits

n	50	150	350	500	650	1000
LinBox	1.67s	18.18s	126.78s	311.56s	602.5s	1870.85s
NTL	0.39s	11.69s	141.86s	448.3s	922.59s	4124.55s

conclusion

LinBox:

- Utilisation des routines numérique est efficace
- Plug-ins (BLAS, Maple)
- Implantations efficaces (corps finis, entiers)

Perspectives

- Développer: outils pour les entiers (interface d'anneaux, CRT)
- Implanter: systèmes linéaires diophantiens (idée [Giesbrecht 1997])
- Généraliser: interaction entre logiciels de calcul formel (ROXANE,Maple)

Questions:

⋆ Algorithme pour matrices creuses sur $\mathbb{Z}$ en $\tilde{O}(n^2)$
⋆ Système linéaire singulier: sans calcul de rang ?