Applied Symbolic Computation
(CS 300)

Dixon’s Algorithm for Solving Rational Linear Systems

Jeremy R. Johnson
Introduction

• Objective: To derive and analyze the computing time of Dixon’s algorithm for solving integer linear systems over the rationals.
  
  – Review complexity of solving linear systems and computing Determinants
  – Hadamard Bound
  – Dixon’s Algorithm
    • mod p inverse
    • P-adic lifting
    • Rational Reconstruction
  – Determinant Algorithm
Determinant Bound

• \[ A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \]

• \[ |\text{Det}(A)| \leq n!d^n \leq (nd)^n, \text{ where } d = \max |a_{ij}| \]

• \[ |\text{Det}(A)| \leq \prod_{i=1}^{n} \|A_i\| \leq n^{n/2}d^n \quad \text{[Hadamard’s Inequality]} \]

• Bound on length of integer determinant

• \[ L(\text{Det}(A)) = O(nL(nd)), \ A \in \mathbb{Z}^{n \times n} \]
Complexity of Gaussian Elimination

• Number of operations performed in the i-th iteration
  – $2(n-i)^2 + (n-i)$
  – Perform $(n-i)$ row operations
  – Each row operation involves 1 division and $(n-i)$ multiplications and subtractions

• Therefore the total number of operations is

  $$\sum_{i=1}^{n-1} 2(n - i)^2 = \frac{2}{3} n^3 - \frac{1}{2} n^2 - \frac{1}{6} n = \Theta(n^3)$$

• This bound does not take into account coefficient growth (bit complexity)
Complexity of Rational GE

• Cost of Rational Arithmetic
  \[ A = \frac{a_1}{a_2}, B = \frac{b_1}{b_2}, L(A) = L(a_1) + L(a_2), L(B) = L(b_1) + L(b_2) \]
  \[ A + B = \frac{a_1b_2 + a_2b_1}{a_2b_2}, AB = \frac{a_1b_1}{a_2b_2} \]
  – Cost \( L(A)L(B) \)

• Cost of GE
  – \( n^3 \) operations
  – Size of fractions in LU bounded by \( nL(nd) \)
  – Bound on rational operations \( n^2L(nd)^2 \)
  – Total time \( O(n^5L(nd)^2) \)
Modular Algorithm

- \( \varphi_p(Det(A)) = Det(\varphi_p(A)) \), where \( \varphi_p(x) = x \mod p \)

- for \( i \) from 1 to \( t \) do
  - Compute \( Det(\varphi_{p_i}(A)) \) using GE

- Use CRT to compute \( D \equiv Det(A) \mod p_i, i = 1, \ldots, t \)

- Need \( t = \Theta(nL(nd)) \)
- Time for modular determinant \( O(n^3) \)
- Time for CRT \( O\left((nL(nd))^2\right) \)
- Total time \( O(n^2L(nd)^2 + n^4L(nd)) \)
Cramer’s Rule

• Assume $A$ is an $n \times n$ non-singular matrix
• Unique solution to $Ax=b$

• Let $A_i(b)$ be the matrix whose $i$th column is replaced by $b$

• $x_i = \frac{\text{Det}(A_i(b))}{\text{Det}(A)}$

• Time to solve when $A$ has integer coefficients
• $O(n(n^2L(nd)^2 + n^4L(nd))) = O(n^3L(nd)^2 + n^5L(nd)))$
Dixon’s Algorithm

- **Solve** $Ax = b, A \in \mathbb{Z}^{n \times n}, b \in \mathbb{Z}^n, x \in \mathbb{Q}^n, d = |A|_{\infty}$

1. **[mod p inverse]**
   1. $C = A^{-1}(mod \ p), p \nmid det(A)$

2. **[p-adic lifting]**
   1. Compute $x_i = Cb_i (mod \ p), \ i = 1, ..., m$
   2. $b_0 = b, b_{i+1} = \frac{b_i - Ax_i}{p}, i = 1, ..., m$ \quad [\text{division is exact}]
   3. $\bar{x} = \sum_{i=0}^{m-1} x_ip^i$ \quad $[A\bar{x} \equiv b \ (mod \ p^m)]$

3. **Rational Reconstruction**
   1. Find $\bar{x}_k \equiv \frac{a_k}{b_k} \ (mod \ p^m), \ |a_i|, |b_i| < (nd)^n$
P-adic Lifting

1. Compute \( x_i = Cb_i \pmod{p} \), \( i = 1, \ldots, m \)

2. \( b_0 = b, b_{i+1} = \frac{b_i - Ax_i}{p}, i = 1, \ldots, m \)  
   [division is exact ]

3. \( \bar{x} = \sum_{i=0}^{m-1} x_i p^i \)
   \( [A\bar{x} \equiv b \pmod{p^m}] \)

\[
b_i - Ax_i \equiv b_i - ACb_i \equiv b_i - b_i \equiv 0 \pmod{p}
\]

\[
A\bar{x} = \sum_{i=0}^{m-1} p^i Ax_i = \sum_{i=0}^{m-1} p^i (b_i - pb_{i+1}) = \\
b_0 - pb_1 + pb_1 - p^2 b_2 + \cdots + p^{m-1} b_{m-1} - p^m b_m = \\
b - p^m b_m \equiv b \pmod{p^m}
\]

- **Complexity** = \( O(mn^2 \log(nd)) \)
Rational Reconstruction

- Compute $\frac{a}{b} \equiv c \pmod{m}$, $|a|, |b| \leq \sqrt{m/2}$

$u := (1,0,m)$
$v := (0,1,c)$

while $\sqrt{m/2} \leq v_3$

  $q := \begin{bmatrix} u_3 \\ v_3 \end{bmatrix}$; $r := u - qv$; $u := v$; $v := r$;

if $|v_2| \geq \sqrt{m/2}$ then error() else return $[v_3, v_2]$

- $v_1m + v_2c = v_3 \rightarrow v_2c \equiv v_3 \pmod{m}$
- Complexity $O(\log(m)^2)$
Complexity of Dixon’s Algorithm

- Solve $Ax = b$, $A \in \mathbb{Z}^{n \times n}$, $b \in \mathbb{Z}^n$, $x \in \mathbb{Q}^n$, $d = |A|_{\infty}$
- $\sqrt{p^m/2} \geq (nd)^n \rightarrow m = \Theta(n \log(nd))$

1. [mod p inverse]
   - $\Theta(n^3)$

2. [p-adic lifting]
   - $\Theta(mn^2 \log(nd)) = \Theta(n^3 \log(nd^2))$

3. Rational Reconstruction
   - $\Theta(nm^2) = \Theta(n^3 \log(nd^2))$

- Total time $\Theta(n^3 \log(nd^2))$
Determinants via Dixon’s Algorithm

- **Solve** $Ax = b, A \in \mathbb{Z}^{n \times n}, b \in \mathbb{Z}^n, x \in \mathbb{Q}^n, d = |A|_\infty$
  - By Cramer’s rule, $x_i = \text{Det}(A_i(b))/\text{Det}(A)$
  - Denominator of $x_i$ divides Det(A)
  - With high probability Det(A) = lcm($x_i$) [need to investigate]

- **Total time** $\Theta(n^3 \log(nd)^2)$?
Summary

• Solving Integer Linear Systems
  – Gaussian Elimination \( \mathcal{O}(n^5L(nd)^2) \)
  – Cramer’s Rule \( \mathcal{O}(n^3L(nd)^2 + n^5L(nd)) \)
  – Dixon’s Algorithm \( \mathcal{O}(n^3\log(nL)^2) \)

• Computing Determinants of Integer Matrices
  – Gaussian Elimination \( \mathcal{O}(n^5L(nd)^2) \)
  – Modular Algorithm \( \mathcal{O}(n^2L(nd)^2 + n^4L(nd)) \)
  – Dixon’s Algorithm \( \mathcal{O}(n^3\log(nd)^2) \)

• Bounds can be improved using Block methods and fast matrix multiplication