Diophantine Linear System Solving

Haomin Li *

Abstract

Solving a system of linear Diophantine equations is one of the most fundamental problems in Computer Algebra. In this paper, we will mainly introduce the randomized algorithm in the paper of Diophantine Linear System Solving by Storjohann and Mulders. [MS99].

1 Introduction

Diophantine refers to the mathematician, Diophantus, who was born in 200 AD in Alexandria, Egypt. Diophantus is one of the first mathematician to introduce symbolism into algebra. The method he formulated in his book, *Arithmetica*, for solving later became known as Diophantine analysis.

Diophantine equations is a polynomial equation with 2 or more integers unknowns. A Linear Diophantine equation is a Diophantine equation with each each unknown has degree at most 1. A system of Linear Diophantine equation can be written in matrix form as $Ax = b$, where A is an integer matrix and b contains all the right side of the equations. The goal is to find an integer vector x.

In this paper, we will show the idea and the probabilistic algorithm of Diophantine linear system solving designed by Storjohann and Mulders. The idea of the algorithm can be divided into three parts:

1. Extract a full row rank subsystem which the same solution space
2. Solve the subsystem and get a sequence of rational solutions. Combining the sequence of rational solutions and obtain a Diophantine solution (x', d')
3. Verify if (x', d') is the solution of the original system

1.1 Related Work

1. Efficient Rational Number Reconstruction [CE95]
2. Efficient Parallel Solution of Sparse Systems of Linear Diophantine Equations (the previous fastest Systems of Linear Diophantine Equations randomized algorithm) [Gie97]
3. Hadamard's inequality from Algorithms for computer algebra

*University of Waterloo, email: h439li@uwaterloo.ca.

1.2 Organization

We divide the algorithm into three parts. In section 3, we will show each part of the algorithm and perform correctness analysis.
In section 4, we will show the iteration lower bound to ensure the correctness. Then, we will come all the three parts and show the algorithm.
In section 5, we will conclude the survey and present some further works.

2 Preliminaries

In this section, we establish the notation which will be used throughout the paper and some important background which we shall need to prove our claims in the next sections.

2.1 General Facts and Notations

We will work over the principal ideal domain $\mathbf{R}$ and its quotient field $\mathbf{K}$. In the following, we mainly refer $\mathbf{R}$ to $\mathbb{Z}$.

Definition 2.1 (Rational solution)**.**
Let $x \in \mathbf{K}^m$ such that $Ax = b$. We define (z, w) is a rational solution of where $w \in \mathbf{R}, z \in \mathbf{R}^m$. $x = \frac{1}{w} \cdot z$. That is $Ax = b \iff Az = wb$.

Definition 2.2 (Diophantine solution)**.**
Let (z, w) be a ration solution of $Ax = b$. If w is a unit in $\mathbf{R}$, then we define (z, w) as a Diophantine solution of the system.

Definition 2.3 (Denominator)**.**
For $x \in \mathbf{K}^m$, a generator of the ideal of all $u \in \mathbf{R}$ such that $ux \in \mathbf{R}^m$ is called a denominator of x.

Definition 2.4 (Minimal denominator)**.**
A rational solution (z, w) of $Ax = b$ is called a solution with minimal denominator if w is an associate of $d(A, b)$. i.e. $\frac{w}{d(A,b)}$ is a unit in $\mathbf{R}$

Definition 2.5 (Notation: d(A,b))**.**
$d(A, b)$ is a generator of the ideal $\{w \in \mathbf{R} \mid \exists z \in \mathbf{R}^m : Az = wb\}$.
d(a,b) is the minimal denominator that a rational solution of $Ax = b$ can have in the sense that $d(A, b)$ divides the denominator of any rational solution of $Ax = b$. We can say that $Ax = b$ has a Diophantine solution $\iff$ $d(A, b)$ is a unit in $\mathbf{R}$.

Definition 2.6 (Notation: $ord_p(a)$)**.**
Let $p \in \mathbf{R}$ be prime. For $a \in \mathbf{R}$ we define $ord_p(a)$ as the maximum integer n such that $p^n \mid a$.

Lemma 2.7. *Let (z, w) and (y, v) be rational solutions of $Ax = b$. $d = gcd(w, v) = sw + tv$, where $d, s, t \in \mathbb{Z}$. Then $(sz + ty, d)$ is a rational solution of $Ax = b$.*

Proof.

$$\begin{aligned} A \cdot (sz + ty) &= sAz + tAy \\ &= swb + tvb \\ &= (sw + tv) \cdot b = db \end{aligned}$$

Hence, $(sz + ty, d)$ is also a rational solution of $Ax = b$.

□

Lemma 2.8. *(Extends from lemma 2.7) Let (z, w), (y, v) and (q, u) be rational solutions of $Ax = b$. Let $t \in \mathbb{Z}$ such that $gcd(v + tu, w) = gcd(v, u, w)$. Then $(y + tq, v + tu)$ is a rational solution of $Ax = b$.*

Proof.

$$\begin{aligned} A \cdot (y + tq) &= Ay + tAq \\ &= vb + tub \\ &= (v + tu) \cdot b \end{aligned}$$

Hence, $(y + tq, v + tu)$ is also a rational solution of $Ax = b$. □

Lemma 2.9. *Let $P \in \mathbb{Z}^{m \times n}$. If (z, w) is a rational solution of $APx = b$, then (Pz, w) is a rational solution of $Ax = b$*

Proof. $A \cdot (Pz) = AP \cdot z = wb$
Hence, (Pz, w) is a rational solution of $Ax = b$. □

Definition 2.10 (Precondition)**.**
Preconditioning the system $Ax = b$ is defined by multiplying the matrix A by a matrix P in order to randomize the primes that divide the denominator of the computed rational solution x of the system $Ax = b$.

Definition 2.11 (Reduced column echelon form)**.**
Let $A \in \mathbf{R}^{n \times m}$ and $rank(A) = s$. B is in reduced column echelon form of $A \iff$ the last $n - s$ columns are zero. It is similar to RREF but with the columns reduced. The key is that $rank(B) = rank(A)$ and $Col(B) = Col(A)$.

Lemma 2.12. *Let $A \in \mathbf{R}^{m \times n}$ and $rank(A) = s$. $SizeOf(\{0, 1\}^m \mid u \in Col(A)\} \leq 2^s$*

Proof. Let B be the reduced column echelon form of A and B has full column rank. Since $Col(A) = \{Ax \mid x \in \mathbf{R}^s\}$. If $Ax \in \{0, 1\}^m$, then $x \in \{0, 1\}^s$. Therefore there are 2^s combinations. □

3 Algorithm Breakdown

The algorithm can be divided into three parts:

1. Extract a full row rank subsystem which the same solution space
2. Solve the subsystem and get a sequence of rational solutions. Combining the sequence of rational solutions and obtain a Diophantine solution (x', d')
3. Verify if (x', d') is the solution of the original system

3.1 Extract the subsystem

The goal is to construct some perturbed systems that have the form $APy = b$, where P is a random dense matrix.

Suppose x' is a rational solution to the perturbed system $APy = b$ where P is a random dense matrix with $\{0, 1\}$. $\implies x = Px'$ is a rational solution to our original system $Ax = b$. Based on a sequence of rational solutions to $Ax = b$, we can possibly compute the Diophantine solution.

For the random dense matrix P, we will randomly pick P from $\{0, 1\}^{n \times n}$.

3.1.1 Theoretical proof

In the following, we will show the requirement $ord_p(w) = ord_p(d(A, b))$ (definition 2.6) of combining a sequence of rational solutions of perturbed systems to get the Diophantine solution of the original system. Besides, we will show the success probability of fulfilling the requirement when P is randomly picked from $\{0, 1\}^{n \times n}$.

Let $A \in \mathbf{R}^{n \times m}$ be of rank n and $b \in \mathbf{R}^n$

Definition 3.1. Let $p \in \mathbf{R}$ be prime. $P \in \mathbf{R}^{m \times n}$ is a good preconditioner with respect to p for the system $Ax = b$ if:

- AP is non singular
- $ord_p(w) = ord_p(d(A, b))$, w is a denominator of $(AP)^{-1}b$

Lemma 3.2. *Let $P \in \mathbf{R}$ be such that AP is non-singular, and let w be a denominator of $(AP)^{-1}b$. Then $(P(w(AP)^{-1}b), w)$ is a rational solution of $Ax = b$*

Proof.

$$\begin{aligned} A \cdot (P(w(AP)^{-1}b) &= w \cdot (AP) \cdot (AP)^{-1} \cdot b \\ &= w \cdot I \cdot b \\ &= wb \end{aligned}$$

□

By the Lemma 3.2, we could have a rational solution of $Ax = b$ without extra prime divisor p in its denominator which is $ord_p(w) = ord_p(d(A, b))$.

Lemma 3.3. *Let $A \in \mathbf{R}^{n \times m}$ of rank n and $b \in \mathbf{R}^n$. There exists a $W \in \mathbf{R}^{n \times m}$ such that for every prime $p \in \mathbf{R}$:*

1. *$rank(W) = n \mod p$*
2. *For $P \in \mathbf{R}^{m \times n}$, p does not divides $\det(WP) \implies P$ is a good preconditioner for $Ax = b$ with respect to p.*

Proof.

1. By Fact 3.7, we can construct the matrices $V \in \mathbf{R}^{m \times m}$ and H. Let W be the first n rows of V^{-1} and $p \in \mathbf{R}$ be prime. Since V^{-1} is also over $\mathbf{R}$ and unimodular, $rank(W) = n \mod p$

2. Let $P \in \mathbf{R}^{m\times n}$ and assume p does not divide $\det(WP)$

   - By Fact 3.7, we have $A = HV^{-1}$, $\begin{bmatrix} H_1 & 0 \end{bmatrix}$, and $H_1 \in \mathbf{R}^{n\times n}$. Since W be the first n rows of V^{-1}, we have $A = H_1W$.
     $\Longrightarrow AP = H_1WP$. Since H_1 is non-singular and WP is non-singular modulo p, we have $AP = H_1WP$ be non-singular.
   - Let w be a denominator of $(AP)^{-1}b$.
     - Prove $ord_p(w) \leq ord_p(d)$:
       Let (y, d) be a rational solution of $Ax = b$ with minimal denominator, where $Ay = db$. With $A = H_1W$ calculated in the previous step, we have $Ay = db \Longleftrightarrow H_1Wy = db \Longleftrightarrow H_1^{-1}b = \frac{1}{d}Wy$

$$\begin{aligned}(AP)^{-1}b &= (H_1WP)^{-1}b \\ &= (WP)^{-1}H_1^{-1}b \\ &= (WP)^{-1}(H_1^{-1}b) \\ &= (WP)^{-1}(\frac{1}{d}Wy) \\ &= \frac{1}{d\cdot\det(WP)}\det(WP)\cdot(WP)^{-1}Wy\end{aligned}$$

       Since $(AP)^{-1}b = \frac{1}{d\cdot\det(WP)}\det(WP)\cdot(WP)^{-1}Wy$ and $\det(WP)\cdot(WP)^{-1}Wy$ is over $\mathbf{R}$, we have $w \mid d\cdot\det(WP)$. Thus we have $ord_p(w) \leq ord_p(d)$.
     - Prove $ord_p(d) \leq ord_p(w)$:
       By Lemma 3.2, we have $(P(w(AP)^{-1}b), w)$ as a rational solution of $Ax = b$. Therefore, $ord_p(d) \leq ord_p(w)$.

     We show that $ord_p(d) = ord_p(w)$.

   We showed that AP is nonsingular and $ord_p(w) = ord_p(d(A, b))$ where w is a denominator of $(AP)^{-1}b$. By Definition 3.1, we show that P is a good preconditioner.

□

By Lemma 3.3, we can find appropriate perturbed matrix P:
Let $p \in \mathbf{R}$ be prime and let $(\hat{z}, \hat{w})$ be a rational solution of $Ax = b$ resulting from combining several rational solutions obtained by using several preconditioners P. In order that there be no extra factor p in $\hat{w} \Longleftrightarrow ord_p(\hat{w}) = ord_p(d(A, b))$. It is required that at least one of these matrices P is a good preconditioner of $Ax = b$ with respect to p. That is $p \nmid \det(WP)$

Next, we will show that the probability that a randomly chosen $P \in \{0, 1\}^{m\times n}$ is a good preconditioner is at least $\frac{1}{4}$.

Theorem 3.4. *Let $A \in \mathbf{R}^{n\times m}$ of rank n, $b \in \mathbf{R}^n$, and $p \in \mathbf{R}$ be prime. the probability that a randomly chosen $P \in \{0, 1\}^{m\times n}$ is a good preconditioner is at least $\frac{1}{4}$.*

Proof.

Lemma 3.5. *Let $W \in \mathbf{R}^{n\times m}$ and $rank(W) = n$. For $0 \leq t \leq n$, we have*

$$SizeOf(\{P \in \{0, 1\}^{m\times t} \mid rank(WP) = t\}) \geq 2^{mt}(1 - (\frac{1}{2})^{n-t+1})\ldots(1 - (\frac{1}{2})^n)$$

Proof. Prove by induction

- Base case: when t = 0. we have $P = 0$ such that $(\{P \in \{0,1\}^{m\times t} \mid rank(WP) = t\})$. Therefore, the base case holds.
- Inductive hypothesis: The statement holds for $t \leq k < n$
- Inductive conclusion: Let K be the right kernel of W. $P = [Q\ u] \in \{0,1\}^{m\times(t+1)}$ where $Q \in \{0,1\}^{m\times t}$ and $u \in \{0,1\}^m$. We have $rank(WP) = t+1 \iff rank(WQ) = t$ and $u \notin Col([K\ Q])$
  When $rank(WP) = k+1$, then $rank([K\ Q]) = m-n+k$ and $u \notin Col([K\ Q]) \longrightarrow rank([K\ Q\ u])] = m-n+k+1$.
  We have that for each $Q \in \{0,1\}^{m\times(k+1)}$ and $rank(WQ) = k+1$, there are at least $2^m - 2^{m-n+(k+1)} = 2^m(1-(\frac{1}{2})^{n-(k+1)})$ vectors $u \in \{0,1\}^m$ where $u \notin Col([K\ Q])$.. By the inductive hypothesis, we have $t = k$, $SizeOf(\{P \in \{0,1\}^{m\times k} \mid rank(WP) = k\}) \geq 2^{mk}(1-(\frac{1}{2})^{n-k+1})\ldots(1-(\frac{1}{2})^n)$. Then, for $t = k+1$, $SizeOf(\{P \in \{0,1\}^{m\times(k+1)} \mid rank(WP) = (k+1)\}) \leq SizeOf(u)\cdot SizeOf(\{P \in \{0,1\}^{m\times k} \mid rank(WP) = k\}) \leq 2^m(1-(\frac{1}{2})^{n-(k+1)})\cdot 2^{mk}(1-(\frac{1}{2})^{n-k+1})\ldots(1-(\frac{1}{2})^n) = 2^{m(k+1)}(1-(\frac{1}{2})^{n-k+2})\ldots(1-(\frac{1}{2})^n)$

The statement is proved by the induction. □

We have $W \in \mathbf{R}^{m\times n}$ and $rank(W) = n$. We choose a random $P \in \{0,1\}^{m\times n}$, the probability that WP is non-singular is $\geq (1-\frac{1}{2})\ldots(1-(\frac{1}{2}(^n) \geq \frac{1}{4}$. (Take $t = n$). Therefore the probability that P is a good preconditioner for $Ax = b$ with respect to prime p when $p \nmid \det(WP) \geq \frac{1}{4}$ □

3.2 Solve the subsystem

We solve the subsystem by using p-adic lifting and rational reconstruction. The rational reconstruction algorithm is proposed by Collins and Encarnacion [CE95]. The algorithm constructs a rational number from its residue modulo a given integer based on Extended Euclidean algorithm.

3.2.1 Idea

Suppose we have a $p \in \mathbb{N}, p > 1, p - \det(A)$. If $p - \det(A) \neq 0$, then p is relatively prime to $\det(A)$. Then x has a unique p-adic expansion $x = z_0 + z_1p + z_2p^2 + \ldots$, where $z_i \in \mathbb{Z}^n$ and $z_i = z_i \mod p$. We can use linear p-adic lifting to compute $z_0, z_1, ..., z_{k-1}$ for large enough k. We have $z = z_0 + z_1p + \cdots + z_{k-1}p^{k-1}$. By the definition of p-adic lifting, $z = x \mod p^k$. Then we can use rational reconstruction with residue z and modulus p^k to compute the subsystem solution x.

We can bound the size of rational solution x to $Ax = b, A \in \mathbf{R}^{n\times n}, \det(A) \neq 0, b \in \mathbf{R}^n$

Theorem 3.6. $|\det(A)| \leq n^{\frac{n}{2}}||A||^n$. $\det(A)x$ *is over* $\mathbb{Z}$ *and satisfies* $||\det(A)x|| \leq n^{\frac{n}{2}}||A||^{n-1}\cdot||b||$

Proof. By Hadamard's inequlity, we have $|det(A)| \leq n^{\frac{n}{2}}||A||^n$

With Cramer's rule

$$\begin{aligned} ||\det(A)x|| &= |\det(A)| \cdot ||x|| \\ &\leq |\det(A)| \cdot ||A^{-1}b|| \\ &\leq n^{\frac{n}{2}} ||A||^n \cdot ||A^{-1}b|| \\ &\leq n^{\frac{n}{2}} ||A||^{n-1} ||b|| \end{aligned}$$

□

Therefore, the size of rational solution to the system can be bounded by $n^{\frac{n}{2}} ||A||^{n-1} ||b||$.

By Theorem 3.6, we have

$$\text{DenominatorBound(A)} \longrightarrow \lceil n^{\frac{n}{2}} ||A||^n \rceil$$

$$\text{NumeratorBound(A, b)} \longrightarrow \lceil n^{\frac{n}{2}} ||A||^{n-1} ||b|| \rceil$$

Collins and Encarnacion [CE95] proved that

Fact 3.7. *we have $z \in \mathbb{Z}$ and $N, D, M \in \mathbb{N}$ be given. If $M > 2ND \Longrightarrow$ there exists at most one $x \in \mathbb{Q}$ with $z = x \mod M$ and with numerator and denominator bounded in magnitude by N and D respectively. If$|z| \leq M$ then such an x can be recovered in $O((logM)^2)$ bit operations.*

Therefore we have

$$\text{LiftingBound(N, D)} \longrightarrow 2N \cdot D$$

Algorithm 1: RationalSolver(A, b, p)

input : $A \in \mathbb{Z}^{n \times n}, p \in \mathbb{Z}$
output: Either **Nil** or $x \in \mathbb{K}^n$ where $Ax = b$

Initialization;
$N :=$ NumberatorBound(A, b);
$D :=$ DenominatorBound(A);
$L :=$ LiftingBound(N,D);
if $p = det(A)$ **then return Nil** ;
$B := A^{-1} \mod p$;
Lift;
$z := 0^{n \times 1}$;
$c := b$;
$M := 1$;
while $M \leq L$ **do**
  $\bar{c} := c \mod p$;
  $\bar{z} := b \cdot \bar{c} \mod p$;
  $c := \frac{c - A\bar{z}}{p}$;
  $z := z + M \cdot \bar{z}$;
  $M := M \cdot p$;
Reconstruct $x :=$ RationalReconstruction(z, M, N, D);
return x

3.3 Verify the solution

After solving a sequence of subsystems, we get a sequence of rational solutions. Then we combine the sequence of rational solution and possibly obtain the Diophantine solution denoted as (x', d'). We test it by substituting (x', d) to the original system $Ax = b$. If $Ax' = d'b$, then (x', d') is the real solution. Otherwise, it is not and the function returns **Nil**.

4 Algorithm

Denote the the error tolerance in the algorithm as ϵ. Since the algorithm – DiophantineSolver is a randomized iterative algorithm, we would like to calculate the minimum number of iteration that returning a rational solution $(\hat{x}, \hat{u})$ of $Ax = b$ with minimal denominator has success probability $\geq 1 - \epsilon$

- Number of iterations:
  Suppose $Ax = b$ given has a rational solution and $rank(A) = t$. Then A should contain a non-singular submatrix $\hat{A} \in \mathbf{R}^{t \times t}$.
  By Hadamard's bound, $\det(\hat{A})$ has at most $\frac{s}{2}$ different prime divisors. When $\hat{p} \nmid \det(\hat{A})$, the $\hat{r}$ computed for $\hat{p}$ should equal to t. Therefore, $\Pr(\hat{r} < t) \leq \frac{1}{2}$ where $\hat{p}$ is randomly chosen from the prime set S. Denote the new subsystem we extract in step 1 as $Bx = c$ and the rank of the subsystem as r. By $\Pr(\hat{r} < t) \leq \frac{1}{2}$, we can derive that $\Pr(r = t) \geq 1 - (\frac{1}{2})^M \geq 1 - \frac{\epsilon}{2}$
  By using the RationalSolver, we calculate the rational solution (x, u) to the subsystem $Bx = c$
  In section 3.1, we showed that a random P from $\{0, 1\}^{m \times n}$ is a good preconditioner for a system with respect to prime $p \in \mathbb{Z}$ has probability $\geq \frac{1}{4}$. Since BP is also nonsingular, by Hadamard's bound, we also have $\det(\hat{BP})$ has at most $\frac{s}{2}$ different prime divisors. For a random $\hat{p}$ from S, $\Pr(p \mid det(BP)) \leq \frac{1}{2}$. From the above analysis, we have that for each iteration, the probability that a rational solution (z, w) of $Bx = c$ with $ord_{\hat{p}}(w) = ord_{\hat{p}}(d(B, c))$ is $\geq \frac{1}{8}$. Let the number of iterations be N. After N iterations of the refining step, the probability that we have $ord_{\hat{p}}(w) > ord_{\hat{p}}(d(B, c))$ instead of $ord_{\hat{p}}(w) = ord_{\hat{p}}(d(B, c))$ is $\leq (1 - \frac{1}{8})^N = (\frac{7}{8})^N$.
  Let u be the least common multiple of denominators in the rational solution of our system. u has at most $log_2(u)$ prime divisors. After solving the subsystem, the probability that we still don't have $ord_{\hat{p}}(w) = ord_{\hat{p}}(d(B, c))$ for some prime divisor $\hat{p}$ of u is $\geq log_2(u)(\frac{7}{8})^N$.
  Since $\Pr(r = t) \geq 1 - (\frac{1}{2})^M \geq 1 - \frac{\epsilon}{2}$ which is what we want. We have:

$$\begin{aligned} log_2(u)(\frac{7}{8})^N &\leq 1 - \Pr(r = t) \\ &\leq 1 - (1 - \frac{\epsilon}{2}) \\ &\leq \frac{\epsilon}{2} \end{aligned}$$

  $log_2(u)(\frac{7}{8})^N \leq \frac{\epsilon}{2} \iff \frac{2 \cdot log_2(u)}{\epsilon} \leq (\frac{8}{7})^N \iff log_{\frac{8}{7}}(\frac{2 \cdot log_2(u)}{\epsilon}) \leq N.$
  Hence, we need at least $log_{\frac{8}{7}}(\frac{2 \cdot log_2(u)}{\epsilon})$ iterations.
  Given the least common multiple of denominators in the rational solution of our system u and the error rate ϵ:

$$\text{NumberOfIterations}(u, \epsilon) \longrightarrow \lceil log_{\frac{8}{7}}(\frac{2 \cdot log_2(u)}{\epsilon}) \rceil$$

- Then we would like to verify the success rate is $\geq 1-\epsilon$:
  We have the probability that $ord_{\hat{p}} = ord_{\hat{p}}(d(B,c))$ for all pirmes $\hat{p}$ is $\geq 1-\frac{\epsilon}{2}$. When the $rank(A) = t \Longleftrightarrow r = t$, then a solution with minimal denominator of the susystem $Bx = c$ is equivalent to the solution with minimal denominator of the original system $Ax = b$. The success probability would $(1-\frac{\epsilon}{2})^2 = 1-\epsilon+\frac{\epsilon^2}{4} > 1-\epsilon$ which concludes our proof.

Besides the NumberOfIterations parameter, we also need one another parameter and two helper functions.

- NumberOfPrimes(A) $\longrightarrow 2 \times \lfloor min(n,m) \cdot \log_w(min(n,m)^{\frac{1}{2}} \cdot m \cdot ||A||) \rfloor$
- ExtendedGCD(u,v) $\longrightarrow (d,s,t)$ where $d = gcd(u,v) = su + tv$
- The Split function:
  Given $v, u \in \mathbf{R}$ The Split function aims to compute $t \in \mathbf{R}$ such that t divides u. Besides, for all primes p we have:
  - If p divides t, then p does not divide v.
  - If p divides $\frac{u}{t}$, then p divides v.

Algorithm 2: Split(v,u)

```
x := v;
t := u;
while x ≠ 1 do
    x := gcd(x, t);
    t := t/x;
return t
```

Combining all the parts, we have the algorithm as below:

Algorithm 3: Diophantine Solver(A, b, ϵ)

input : $A \in \mathbb{Z}^{n\times m}$, $b \in \mathbb{Z}^n$, $\epsilon > 0$
output: Either **NIL** or a rational solution of $Ax = b$

/* Initialization */
s := NumberOfPrimes(A);
S := set of s primes;
/* Find rank and determine subsystem */
$M := \lceil \log_2(\frac{2}{\epsilon}) \rceil$;
$r := -1$;
for i **to** M *do* **do**
  $\hat{p}$:=a random element from S;
  $\hat{r}$:=the rank of A modulo $\hat{p}$;
  if $\hat{r} > r$ **then**
    $p := \hat{p}$;
    $r := \hat{r}$;
  end
end
Q, P := submatrices of permutation matrices such that QAP is an $r \times r$ submatrix of A which is nonsingular modulo p;
$B := QA$;
$c := Qb$;
/* Get initial solution */
q := RationalSolver(BP, c, p);
$y := 0$;
$v := 0$;
/* Refine solution */
N := NumberOfIterations(u, ϵ);
$y := 0$;
$v := 0$;
for i **to** N **do**
  if $gcd(u, v) \neq 1$ **then**
    break;
  end
  P := a random matrix from $\{0, 1\}^{m\times r}$;
  p := a random element from S;
  q := RationalSolver(BP, c, p);
  if $q \neq$ ***NIL*** **then**
    w := Least common multiple of denominators in q;
    if $gcd(v, w, u) \neq gcd(v, u)$ **then**
      t :=Split(v, u);
      $z := P(wq)$;
      $y := y + tz$;
      $v := v + tw$;
    end
  end
end
(d, s, t) := ExtendedGCD(u, v);
$x := sx + ty$;
/* Check solution */
if $Ax = db$ **then return** (x, d) ;
else return Nil;

5 Conclusion and Open Problems

In the paper, we review the problem of Diophantine linear equations system problem and introduce the randomized iterative algorithm purposed by Storjohann and Mulders [MS99]. The randomness occurs when we construct a subsystem which has the same Diophantine solution as the original system.
We also showed the requirement and success probability of the random construction of a subsystem. Besides, we show the bound of number of iterations we need to achieve the success probability we want. The algorithm is vastly based on the concepts of linear algebra.
The runtime dominating part is using the p-adic lift to solve the rational solution. If any faster rational system solving method compared to this method is invented. The total runtime of the algorithm can be reduced by that.
In terms of finding rational solutions to sparse linear system, we could use sparse or structured block projections to accelerate the performance.[EGG$^+$06]

References

[CE95] George E. Collins and Mark J. Encarnación. Efficient rational number reconstruction. *Journal of Symbolic Computation*, 20(3):287–297, 1995.

[EGG$^+$06] Wayne Eberly, Mark Giesbrecht, Pascal Giorgi, Arne Storjohann, and Gilles Villard. Solving sparse rational linear systems. In *Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation*, ISSAC '06, page 63–70, New York, NY, USA, 2006. Association for Computing Machinery.

[Gie97] Mark Giesbrecht. Efficient parallel solution of sparse systems of linear diophantine equations. In *Proceedings of the Second International Symposium on Parallel Symbolic Computation*, PASCO '97, page 1–10, New York, NY, USA, 1997. Association for Computing Machinery.

[MS99] Thom Mulders and Arne Storjohann. Diophantine linear system solving. In *Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation*, ISSAC '99, page 181–188, New York, NY, USA, 1999. Association for Computing Machinery.