Self-contained Robust Cramer Solver

1 Introduction

This document describes a self-contained algorithm for solving linear systems of the form $Ax = b$ using a robust variant of Cramer's rule. The algorithm is designed to handle both overdetermined and underdetermined systems by using the least squares method and minimal norm solution, respectively. Additionally, it includes a stabilization technique that adds a small multiple of the identity matrix ϵI to ensure numerical stability.

2 Cramer's Rule

Cramer's rule provides a solution to a system of linear equations when the number of equations is equal to the number of unknowns. For a 3x3 system:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3 \end{cases}$$

The solution is given by:

$$x_i = \frac{\Delta_i}{\Delta}$$

where Δ is the determinant of matrix A:

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

and Δ_i is the determinant of matrix A with the i-th column replaced by b:

$$\Delta_1 = \begin{vmatrix} b_1 & a_{12} & a_{13} \\ b_2 & a_{22} & a_{23} \\ b_3 & a_{32} & a_{33} \end{vmatrix}$$

$$\Delta_2 = \begin{vmatrix} a_{11} & b_1 & a_{13} \\ a_{21} & b_2 & a_{23} \\ a_{31} & b_3 & a_{33} \end{vmatrix}$$

$$\Delta_3 = \begin{vmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ a_{31} & a_{32} & b_3 \end{vmatrix}$$

3 Least Squares Solution for Overdetermined Systems

For overdetermined systems ($N > 3$), we use the least squares method, which minimizes the mean squared error (MSE):

$$x = (A^T A)^{-1} A^T b$$

To ensure numerical stability, especially when $A^T A$ is nearly singular, we add ϵI to $A^T A$:

$$x = (A^T A + \epsilon I)^{-1} A^T b$$

4 Minimal Norm Solution for Underdetermined Systems

For underdetermined systems ($N < 3$), we augment the matrix A to make it square by adding rows of zeros and augment the vector b by adding zeros. Then, we solve the resulting system using the same method as for the overdetermined case.

5 Algorithm Description

The unified algorithm can be described as follows:

Algorithm 1 Robust Cramer Solver

Require: Matrix $A \in \mathbb{R}^{N\times 3}$, Vector $b \in \mathbb{R}^N$, Scalar ϵ
Ensure: Vector $x \in \mathbb{R}^3$

```
if N < 3 then
    Augment A with (3 − N) rows of zeros
    Augment b with (3 − N) zeros
end if
Compute A^T A
Add εI to A^T A
Compute A^T b
Solve (A^T A + εI)x = A^T b using Cramer's Rule
return x
```

6 Example Implementation

The following is an example implementation of the described algorithm in C++.