Power Method & Triangle Alg.

Yanbo Fang(yf228) & Xuenan Wang(xw336)

Target: Solve $Mx = x$

- Power Method
- Triangle Algorithm
- Jacobi & Gauss_Seidel & SOR

Power Method: Solve $Mx = x$

original Graph

M

$$\begin{pmatrix}1 & 0 & 1 & 1\\ 0 & 1 & 1 & 0\\ 1 & 1 & 1 & 0\\ 1 & 1 & 0 & 0\end{pmatrix} \xrightarrow{\text{process}} \begin{pmatrix}0.32083 & 0.03750 & 0.32083 & 0.46250\\ 0.03750 & 0.46250 & 0.32083 & 0.46250\\ 0.32083 & 0.46250 & 0.32083 & 0.03750\\ 0.32083 & 0.03750 & 0.03750 & 0.03750\end{pmatrix}$$

$$M = d\overline{A} + (1-d)\frac{1}{n}ee^T, \quad d \in (0,1).$$

Usually $d = .85$.

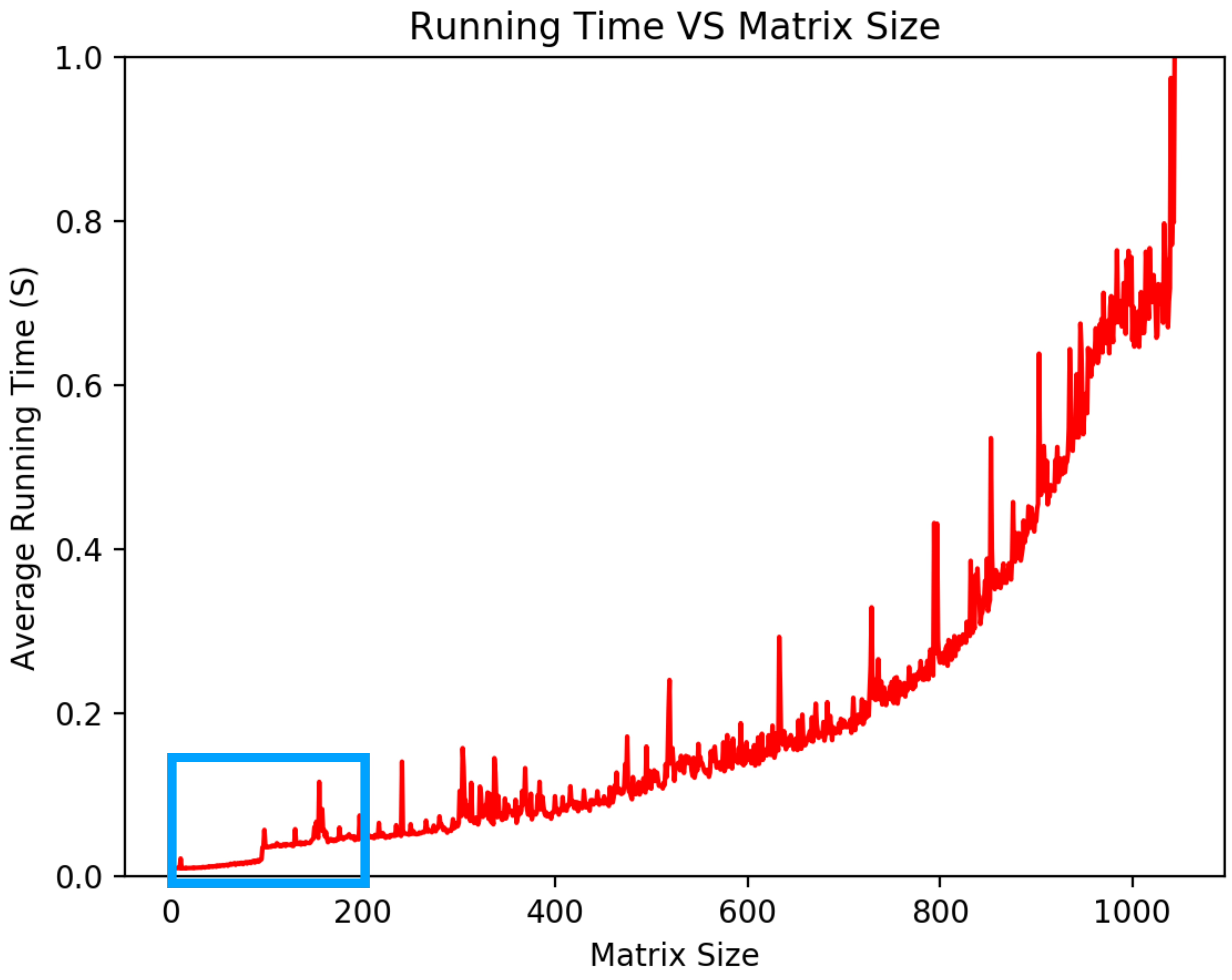
Power Method: Solve $Mx = x$

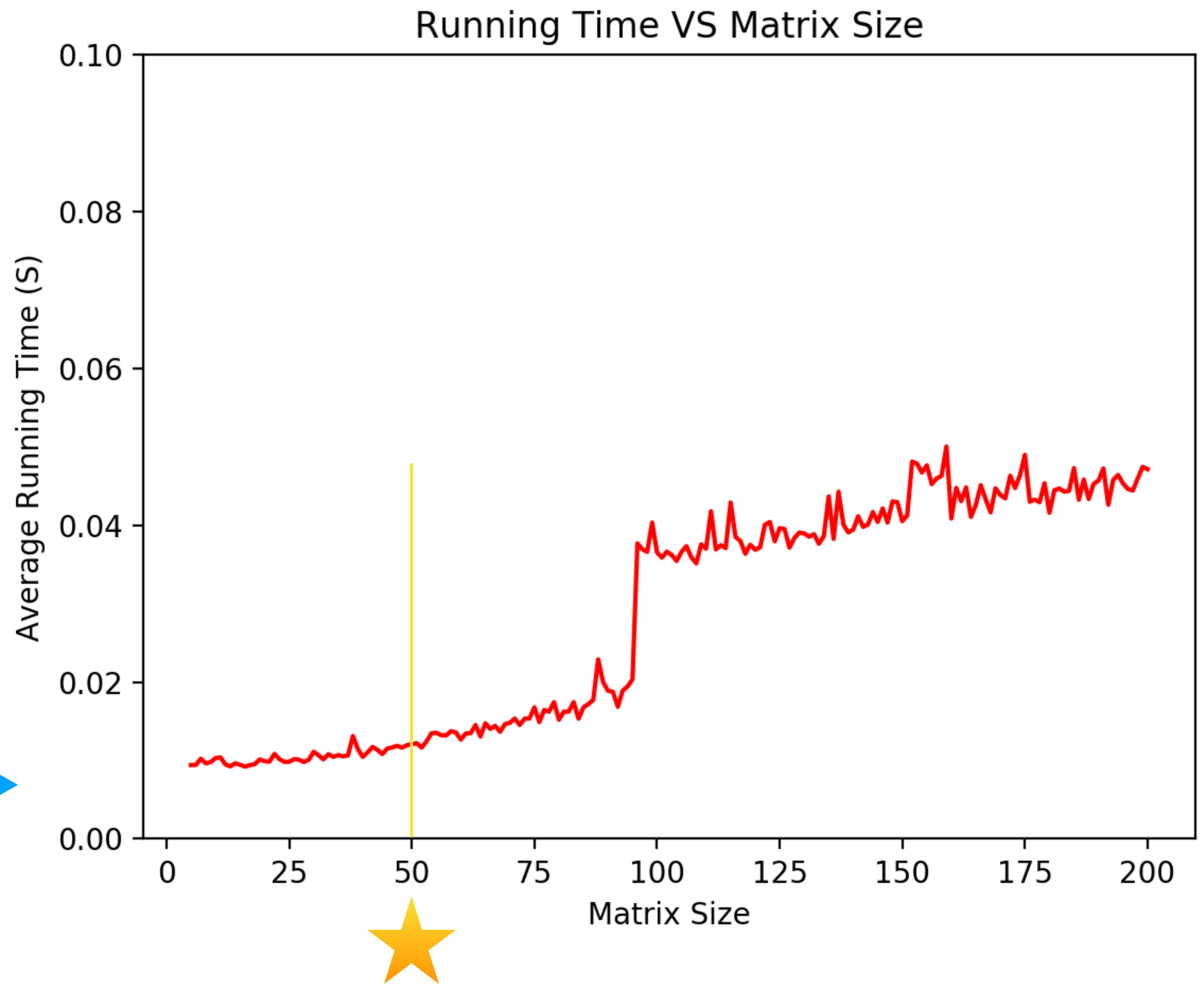
$$M \xrightarrow{\text{Power method}} \overset{\text{Primal x}}{\begin{pmatrix}0.08022\\0.34745\\0.18749\\0.38484\end{pmatrix}} \xrightarrow{\text{200 iteration}} \overset{\text{Result}}{\begin{pmatrix}0.24840\\0.30960\\0.33413\\0.10788\end{pmatrix}}$$

The error is: 0.000008530985

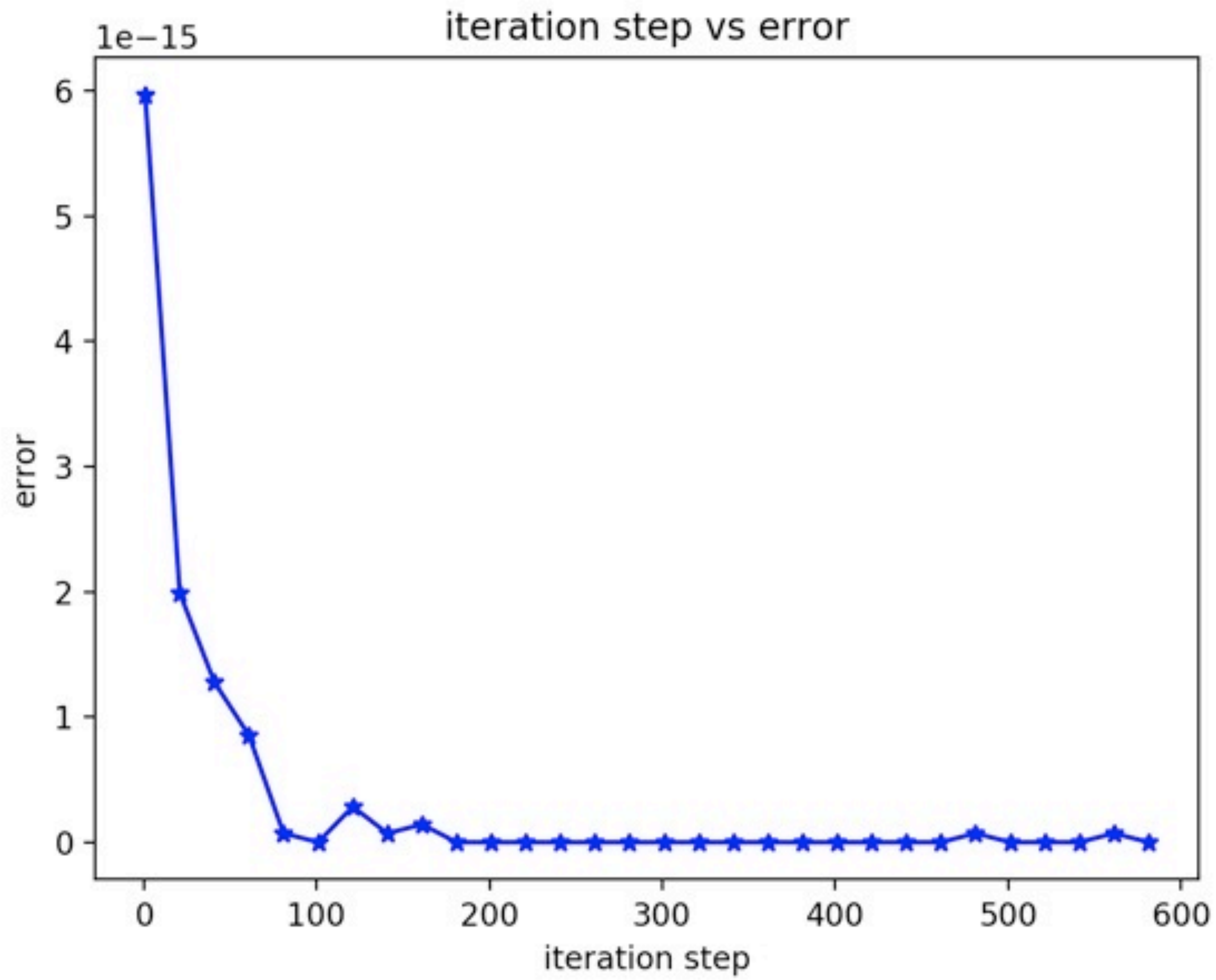
ERROR = abs(Mx-x)

Power Method: Solve $Mx = x$



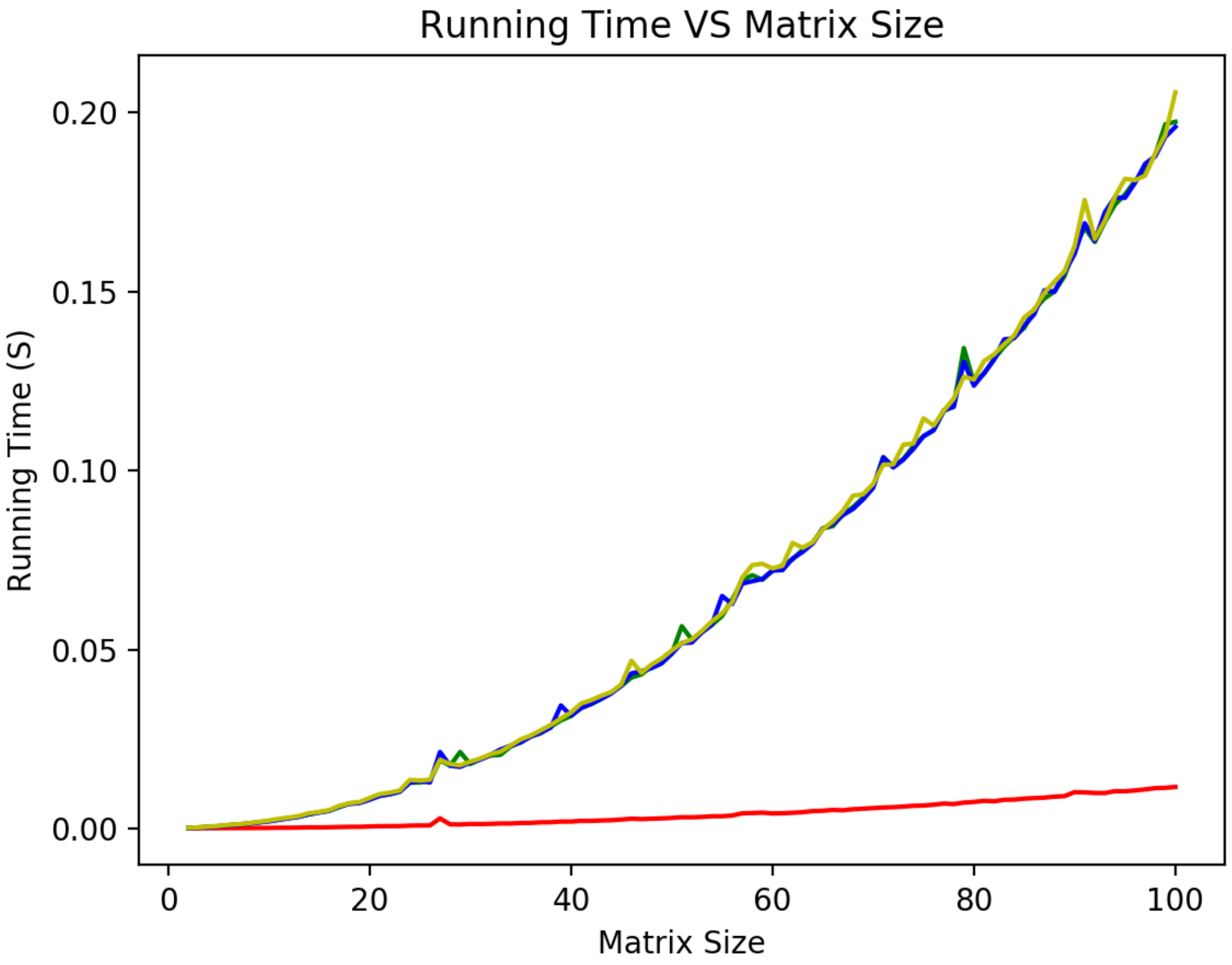


Power Method: Solve $Mx = x$

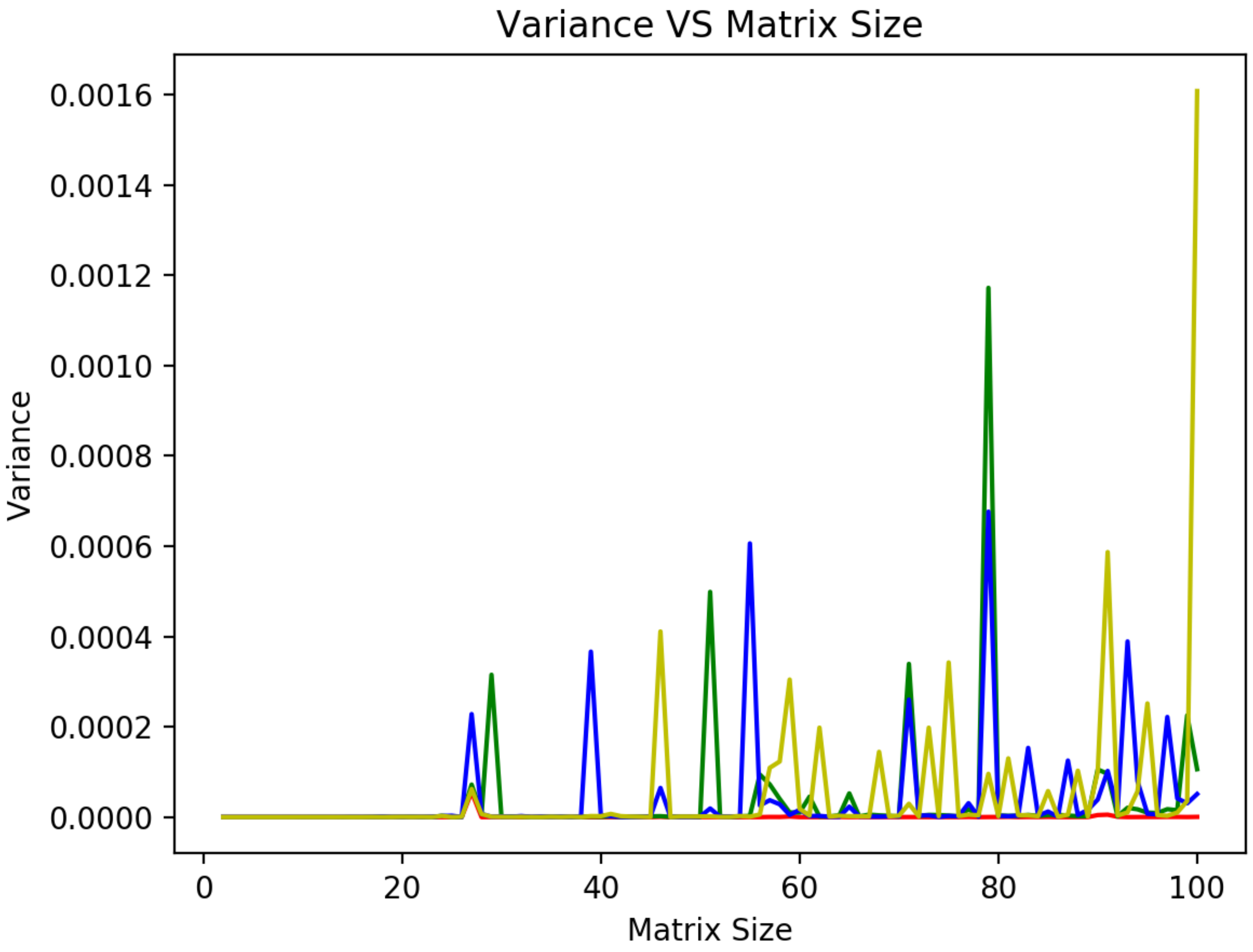


Matrix size: 50

Power Method: Solve $Mx = x$



Running Time VS Matrix Size



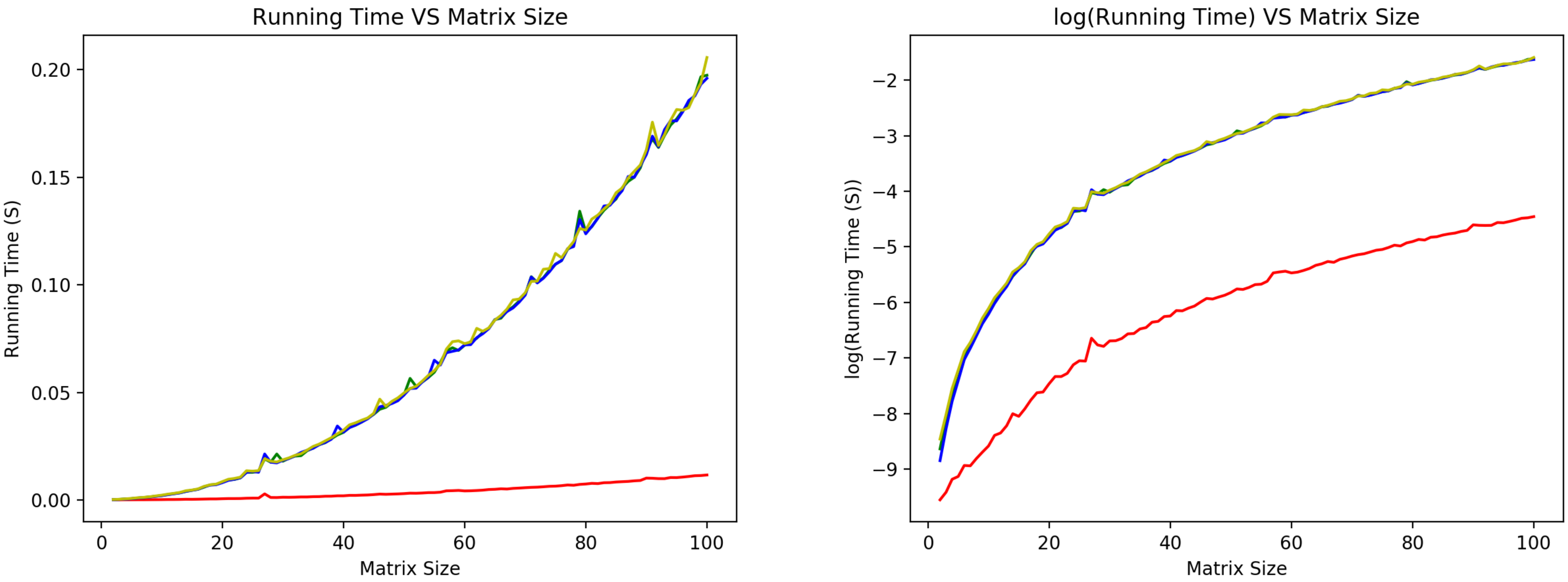
Variance VS Matrix Size

Power Method **Jacobi**

Iteration: 20

Gauss Seidel **SOR**

Power Method: Solve $Mx = x$



Power Method

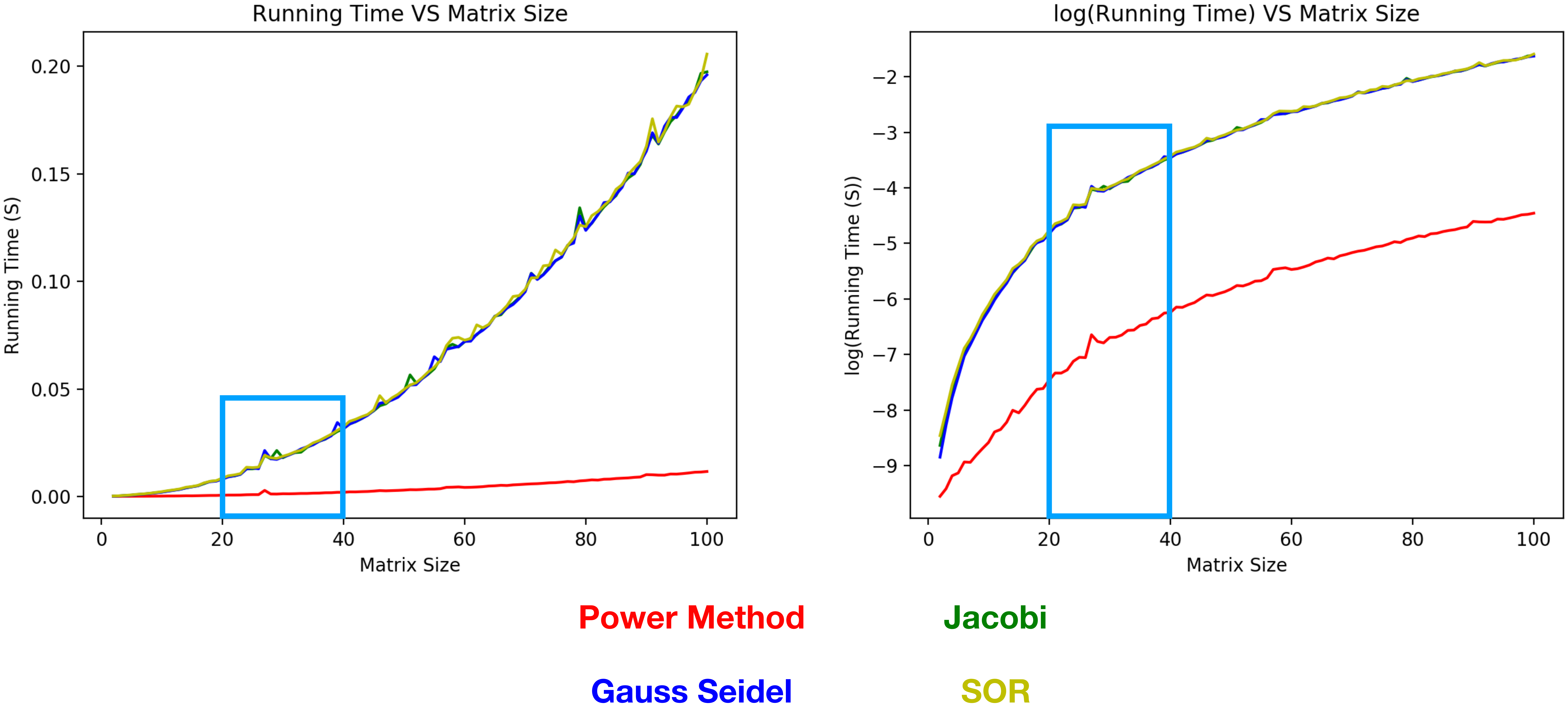
Jacobi

Iteration: 20

Gauss Seidel

SOR

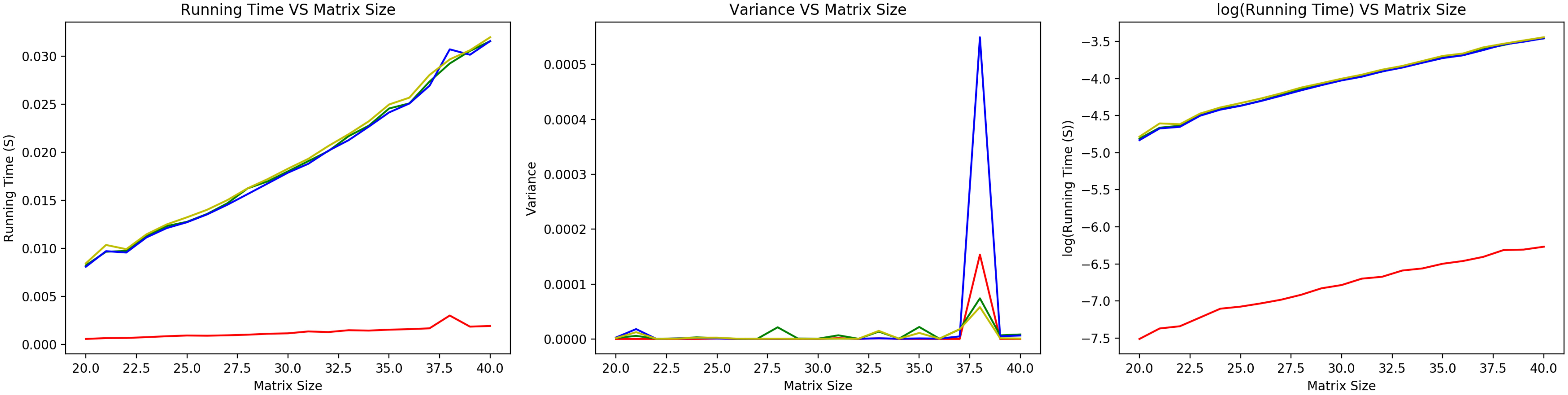
Power Method: Solve $Mx = x$



Running Time VS Matrix Size

log(Running Time) VS Matrix Size

Power Method: Solve $Mx = x$



Iteration: 100

Power Method **Jacobi**

Gauss Seidel **SOR**

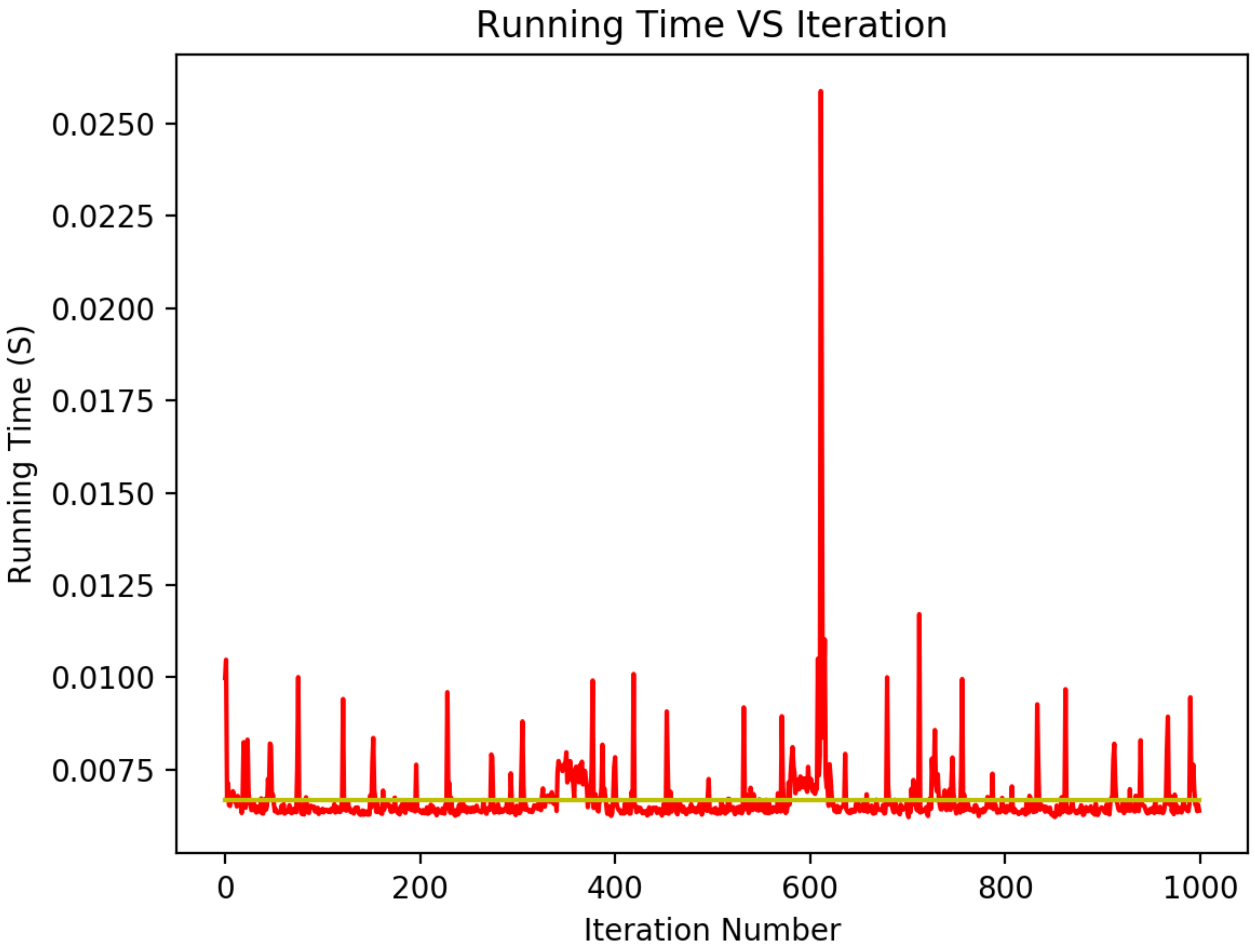
Power Method: Generate Symmetric Matrix

$$\overset{\textbf{Connected graph}}{\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}} \xrightarrow{\textbf{Process}} \overset{\textbf{M}}{\begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}}$$

Power Method: Generate Symmetric Matrix

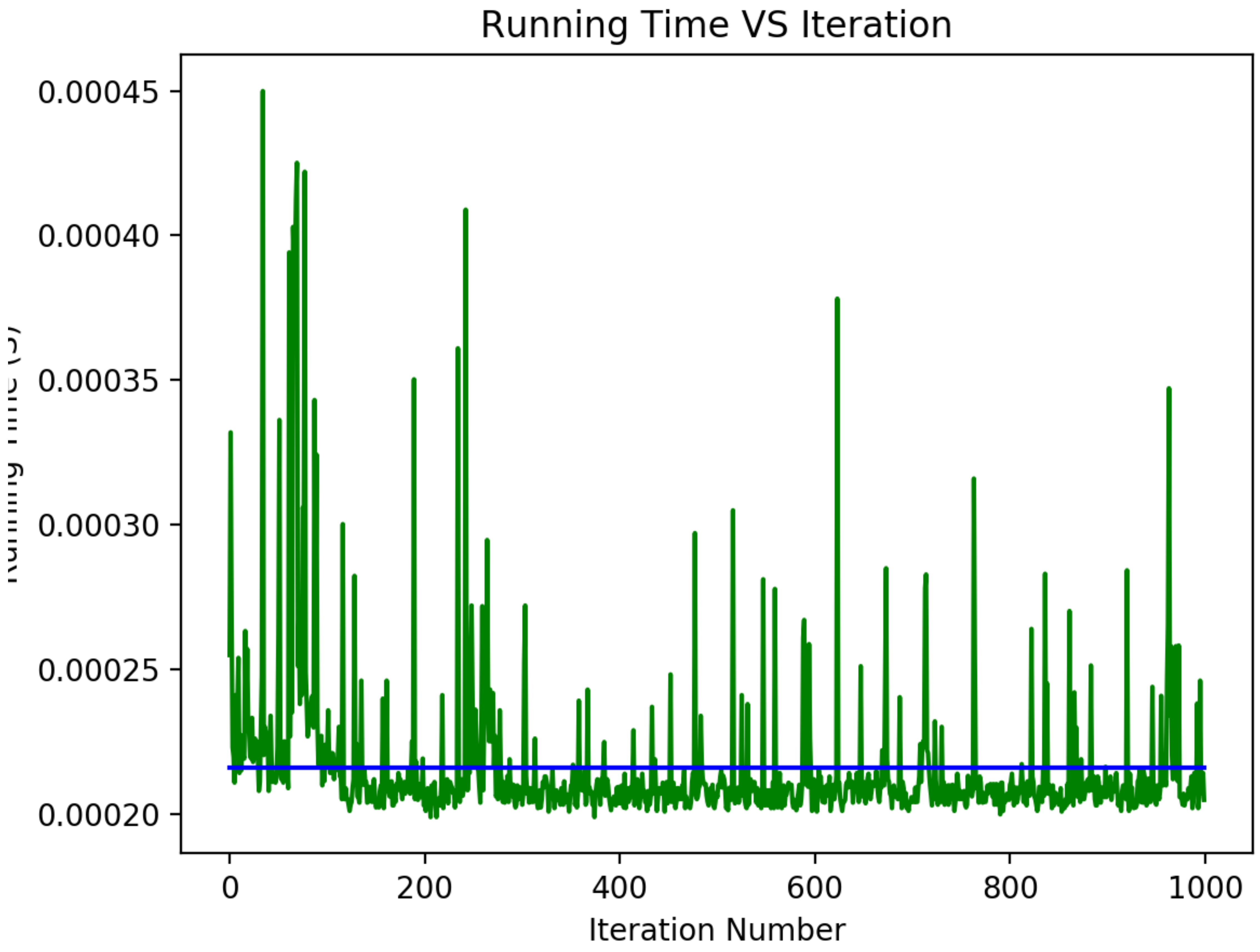
$$\begin{array}{ccccc}
\textbf{M} & & \textbf{Primal X} & & \textbf{Result} \\
\begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}
& \xrightarrow{\textbf{Power method}} &
\begin{pmatrix} 0.71271 & 0.30597 & 0.37185 & 0.30147 \\ 0.30597 & 0.40497 & 0.56893 & 0.78950 \\ 0.37185 & 0.56893 & 0.55788 & 0.06766 \\ 0.30147 & 0.78950 & 0.06766 & 0.71510 \end{pmatrix}
& \xrightarrow{\textbf{200}} &
\begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}
\end{array}$$

Power Method: Apply on Symmetric Matrix



Random Matrix

mean:	0.00668018317222595O
var:	8.55905404366638E-07



Symmetric Matrix

mean:	0.000216048955917358O
var:	7.31300723089134E-10

Random got 100% higher variance in a 100 times test.

Triangle Algorithm: Solve $Mx = x$

$$\underset{}{\overset{\textbf{M}}{\begin{pmatrix} 0.32083 & 0.03750 & 0.32083 & 0.46250 \\ 0.03750 & 0.46250 & 0.32083 & 0.46250 \\ 0.32083 & 0.46250 & 0.32083 & 0.03750 \\ 0.32083 & 0.03750 & 0.03750 & 0.03750 \end{pmatrix}}} \xrightarrow{\textbf{(M-I)x = 0}} \overset{\textbf{M-I}}{\begin{pmatrix} -0.67916 & 0.03750 & 0.32083 & 0.46250 \\ 0.03750 & -0.53750 & 0.32083 & 0.46250 \\ 0.32083 & 0.46250 & -0.67916 & 0.03750 \\ 0.32083 & 0.03750 & 0.03750 & -0.96250 \end{pmatrix}}$$

Triangle Algorithm: Solve $Mx = x$

Primal x **Result**

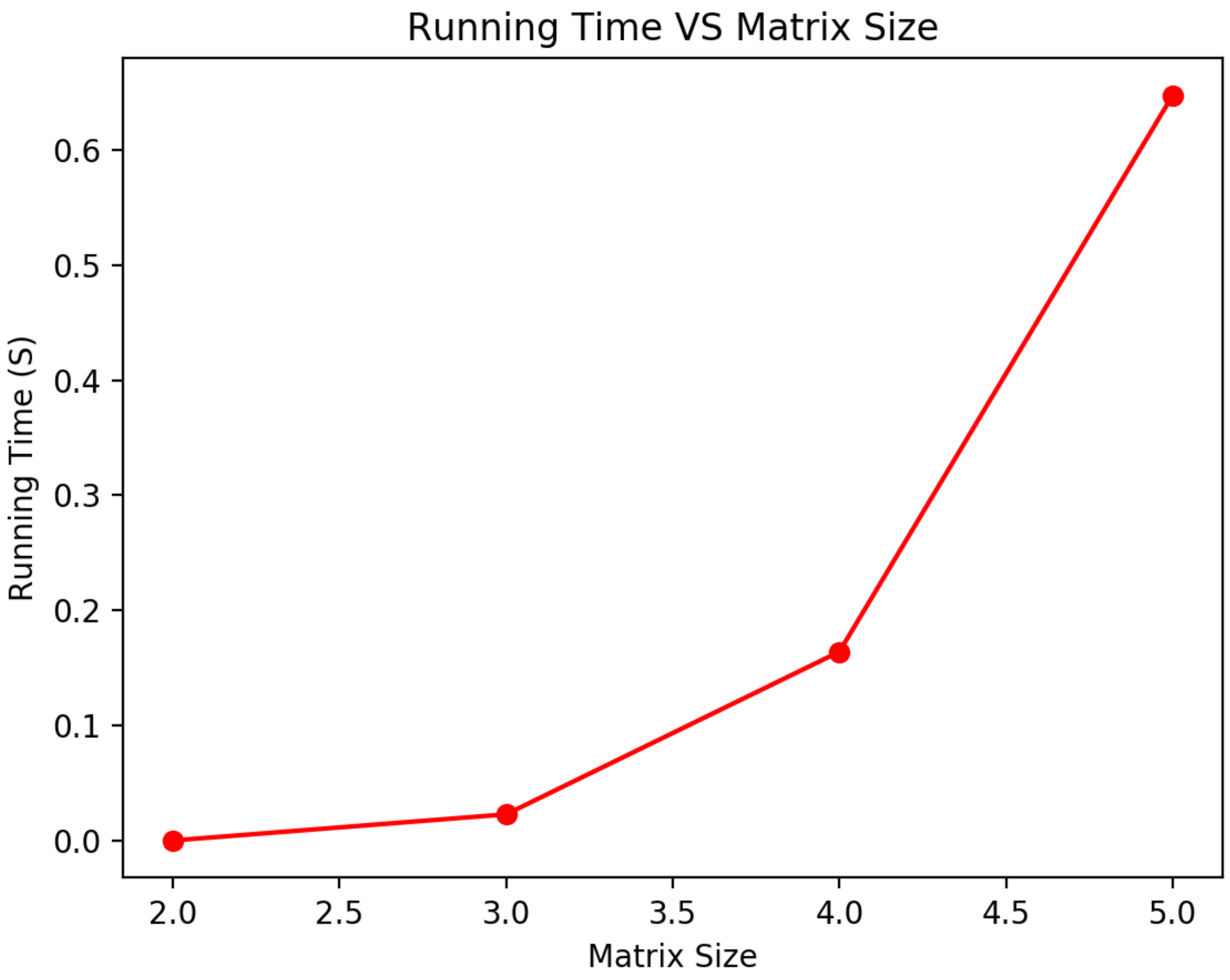
$$\text{M-I} \xrightarrow{\text{Triangle algorithm}} \begin{pmatrix} 0.29427 \\ 0.08407 \\ 0.10436 \\ 0.51730 \end{pmatrix} \xrightarrow{\text{179 Iteration}} \begin{pmatrix} 0.24840 \\ 0.30960 \\ 0.33412 \\ 0.10789 \end{pmatrix}$$

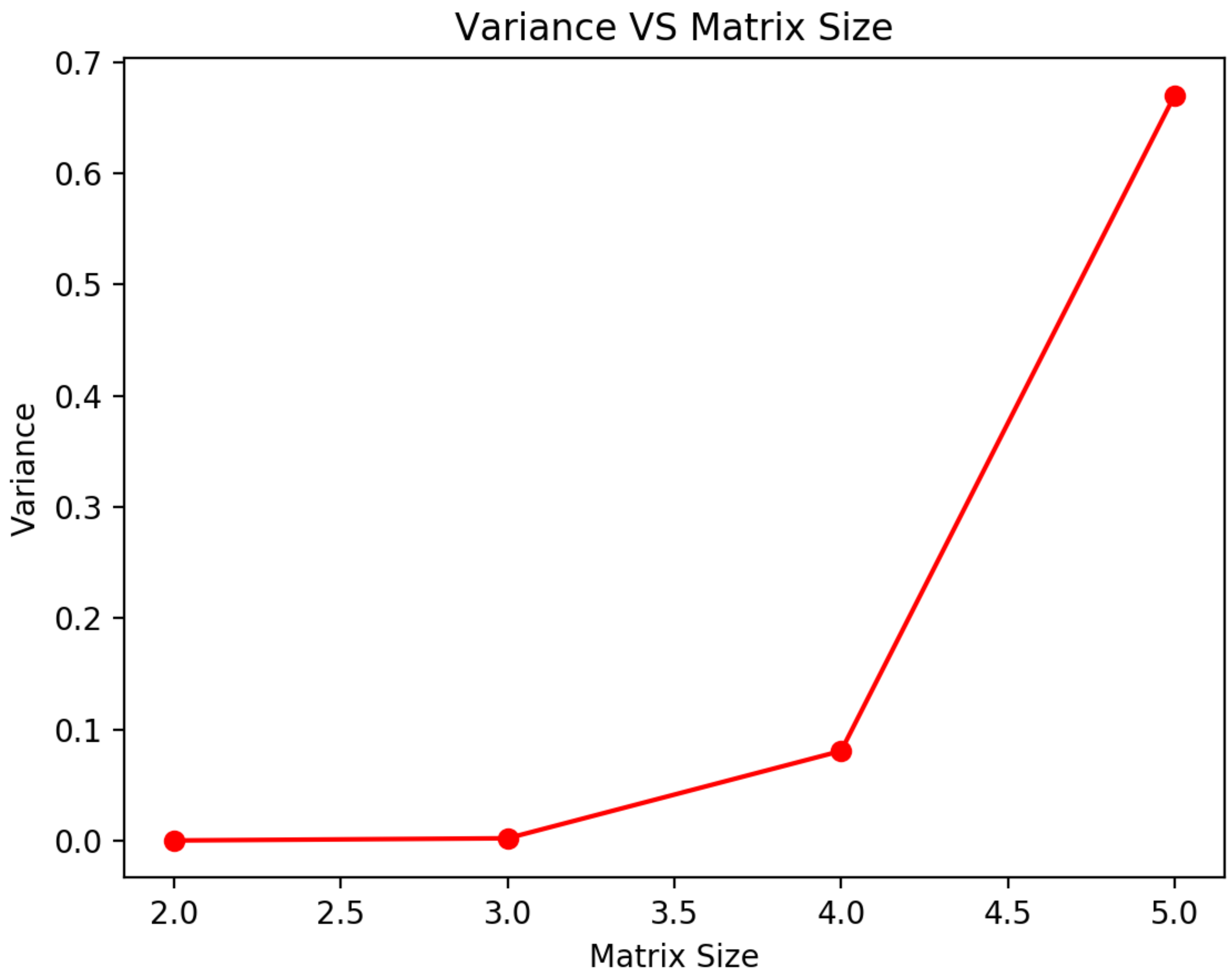
Result of Power Method, error: 0.000008530985

The error is: 0.00001210521

$$\begin{pmatrix} 0.24840 \\ 0.30960 \\ 0.33413 \\ 0.10788 \end{pmatrix}$$

Triangle Algorithm: Solve $Mx = x$





Iteration: 100

THANK YOU