Math 221 Programming Homework 1

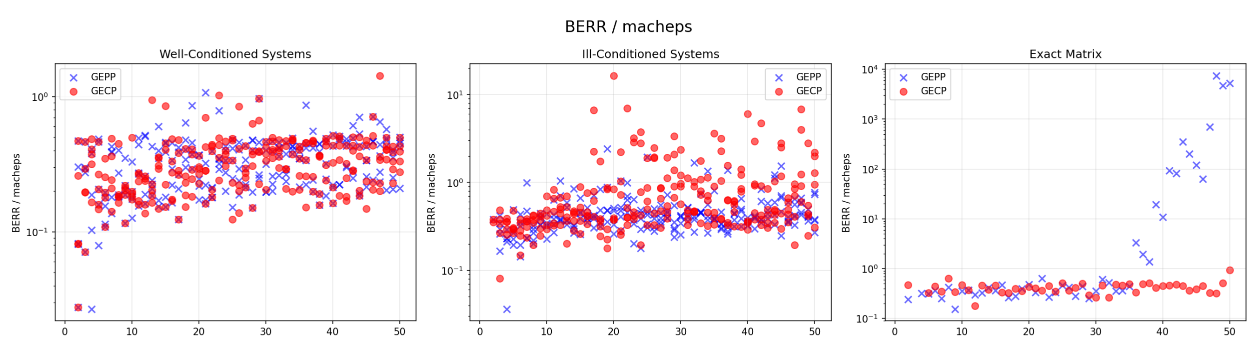
Atharv Sampath

Fall 2025

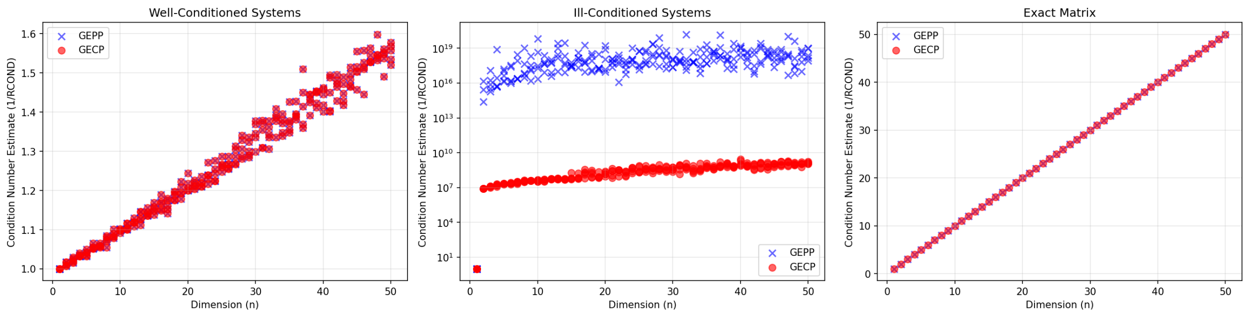
FERR (Forward Error): Estimated forward error bound. It estimates the relative error in the computed solution: $\|X_{\text{computed}} - X_{\text{true}}\|/||X_{\text{true}}||$. This tells you how far your computed solution is from the true solution.

BERR (Backward Error): Componentwise relative backward error. It measures how much you'd need to perturb the original system (A, b) to make your computed solution X exact. Specifically, it's the smallest relative change to A and b such that $(A+\Delta A)X = b+\Delta b$ exactly.

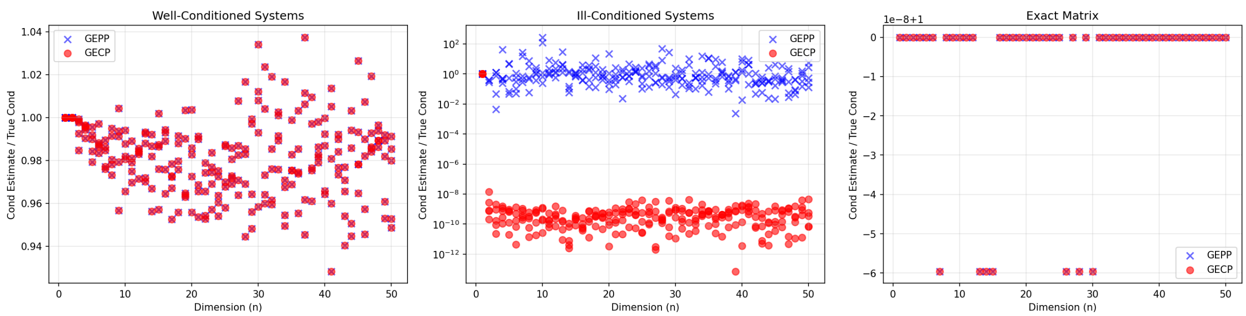
The following is the data returned by my code. I didn't number the test matrices, but the well-conditioned matrices are randomly generated by perturbing a random permutation matrix, the ill-conditioned matrices are randomly generated by choosing $A = L * U$ where U and L are triangular matrices with small diagonal and medium size super/sub diagonal (respectively). Finally, the exact matrices are the family of worst-case pivot growth matrices mentioned in the problem. All other information about the charts is self explanatory.



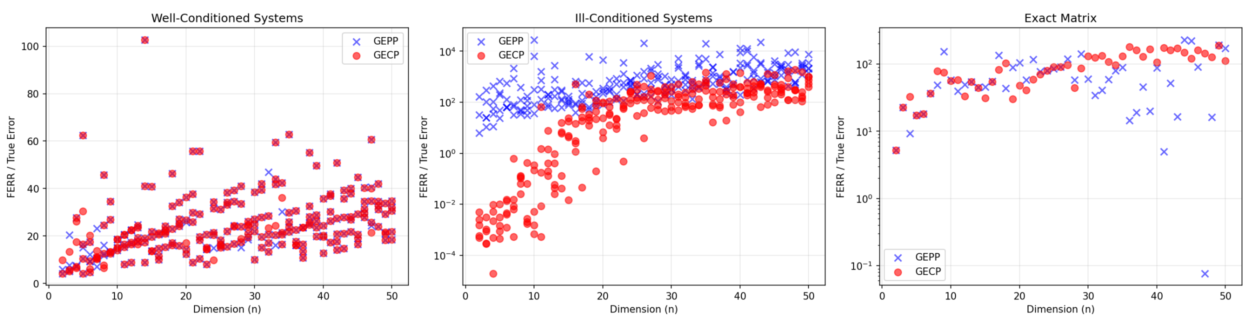
Condition Number Estimate (1/RCOND)



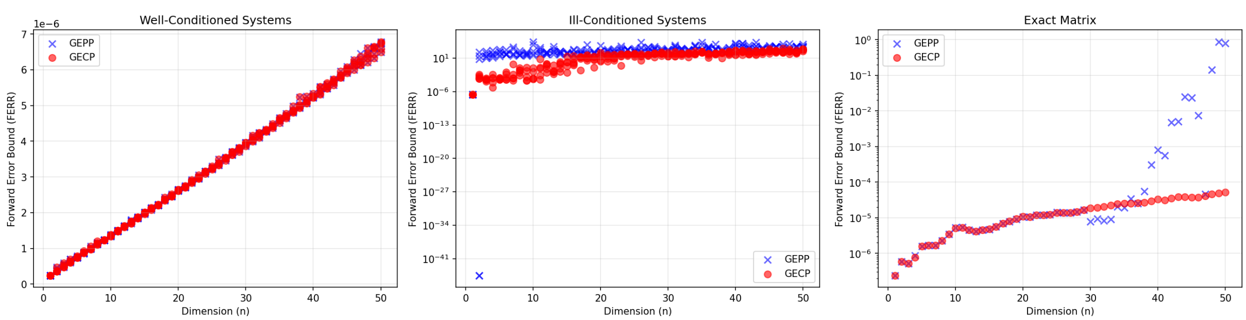
Cond Estimate / True Cond

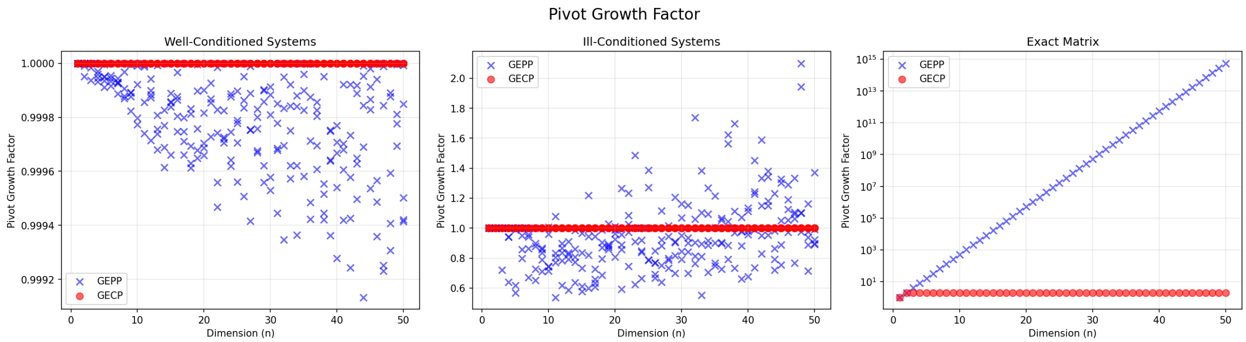


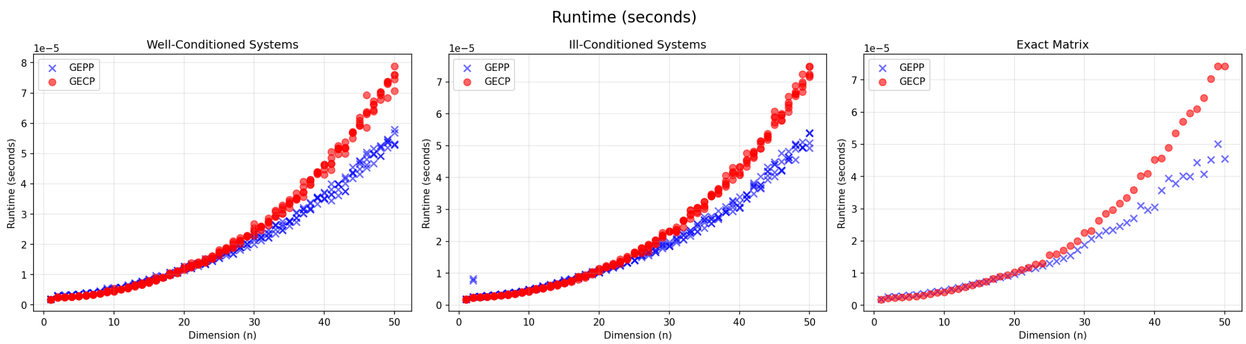
FERR / True Error

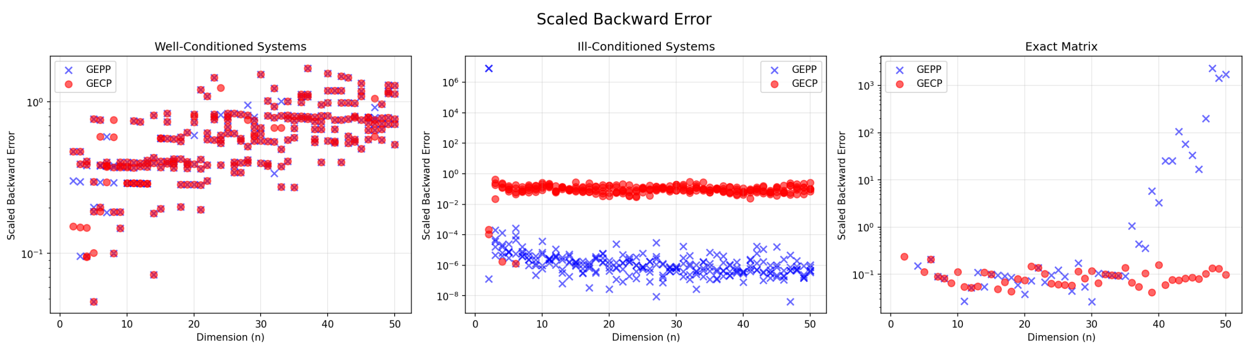


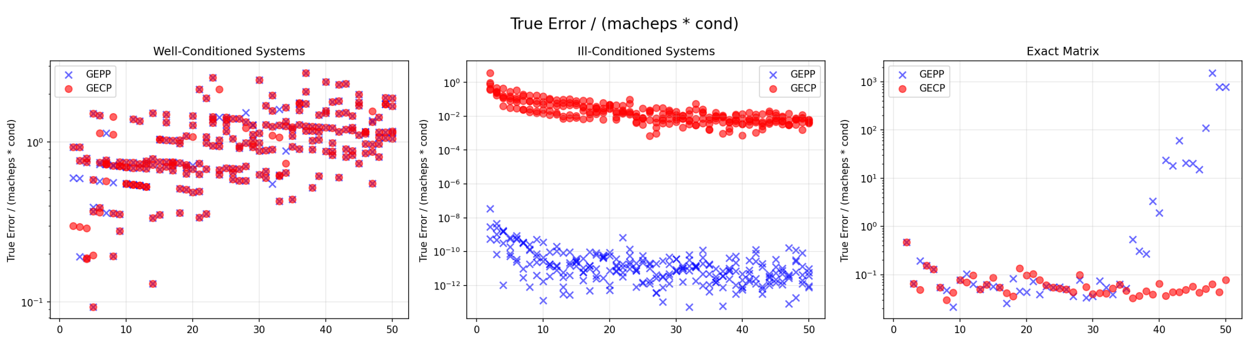
Forward Error Bound (FERR)




Pivot Growth Factor
Well-Conditioned Systems
Ill-Conditioned Systems
Exact Matrix
Pivot Growth Factor
Dimension (n)
GEPP
GECP


Runtime (seconds)
Well-Conditioned Systems
Ill-Conditioned Systems
Exact Matrix
Runtime (seconds)
Dimension (n)
1e−5
GEPP
GECP


Scaled Backward Error
Well-Conditioned Systems
Ill-Conditioned Systems
Exact Matrix
Scaled Backward Error
Dimension (n)
GEPP
GECP


True Error / (macheps * cond)
Well-Conditioned Systems
Ill-Conditioned Systems
Exact Matrix
True Error / (macheps * cond)
Dimension (n)
GEPP
GECP

1. **Do the error bounds really bound the errors?**

   - **FERR (Forward Error Bound):**
     - *Well-conditioned & Exact matrices:* Yes! FERR/True Error is consistently ≥ 1 (mostly 5–100), meaning FERR successfully bounds the true error.
     - *Ill-conditioned systems:* No, not always. The plots show FERR/True Error can be < 1 (some points at 10^{-4} to 10^{-2}), meaning the actual error exceeds the bound. This is a known limitation—FERR estimates can fail for very ill-conditioned problems.
   - **BERR (Backward Error):**
     - BERR/`macheps` $\approx$ 0.1–10 for most cases, indicating BERR $\approx \mathcal{O}(\texttt{macheps})$, which is expected.
     - Notably, for the exact matrix (special structure), GECP achieves much smaller backward error than GEPP (BERR/`macheps` around 0.3 vs. 100–1000+). GECP's complete pivoting handles this pathological case better.

2. **Speed Comparison: GEPP vs GECP**

   - GEPP takes longer and longer as the dimension of the system grows.
   - From the runtime plot:
     - Both scale as $O(n^3)$ as expected.
     - GECP's extra cost comes from searching the entire remaining submatrix for the pivot ($O(n^2)$ per step) vs. just one column for GEPP ($O(n)$ per step).
     - Despite better stability (smaller pivot growth, better backward error on pathological matrices), GECP's speed penalty usually isn't worth it for typical problems.

GEPP is faster and has adequate error bounds for well-conditioned systems, but FERR can underestimate errors for ill-conditioned problems.

One thing to notice is the pivot growth for GEPP on the exact matrices (which is the family of matrices mentioned in the problem). It is clear that if the dimension of the matrix is n, the pivot growth for GEPP is 2^{n-1} which is the worst case pivot growth for GEPP as mentioned in the text.

The source code for all experiments is on Github.