Welcome

Efficiency of Different Algorithms over Sparse Matrices

Muhammad Mohebujjaman
Graduate Student
Clemson University

Outline

- Motivation
- Objectives
- Sparse Matrix
- Representation of Sparse Matrix
- Performance of Different Algorithms

Motivation

- The large matrices that arise in real-world problems are often sparse.
- Several algorithms for solving system of equations.
- Different algorithms have different efficiencies.

Objectives

- To count the number of iteration needs to converge
- To compare the efficiency of different algorithms

Sparse Matrices

- Sparse . . . . . . .many elements are zero.
- Dense . . . . . . .few elements are zero.

Structured Sparse Matrices

- Diagonal
- Tri-diagonal
- Lower triangular

- May be mapped into a 1D array
- Mapping function can be used to locate an element

Unstructured Sparse Matrices

Airline flight matrix.

- Airports are numbered 1 through n
- flight(i,j) = list of nonstop flights from airport i to airport j.

Web page matrix.

- Web pages are numbered 1 through n
- web(i,j) = number of links from page i to page j

Representation of Unstructured Sparse Matrices

- MATLAB provides a sparse function of the form

matrix = sparse (i, j, s, m, n, nz_max)

- i = the row indices of the nonzero elements;
- j = the column indices of the nonzero elements;
- s = the values of the nonzero elements;
- m = the number of rows in the matrix;
- n = the number of columns in the matrix;
- nz_max is the maximum number of nonzero elements in the matrix

Representation of Unstructured Sparse Matrices

$$A = \begin{pmatrix} 11 & 12 & 0 & 0 & 15 \\ 0 & 22 & 23 & 0 & 0 \\ 31 & 0 & 33 & 34 & 35 \end{pmatrix} \quad m = 3,\ n = 5,\ nz\_max = 9;$$

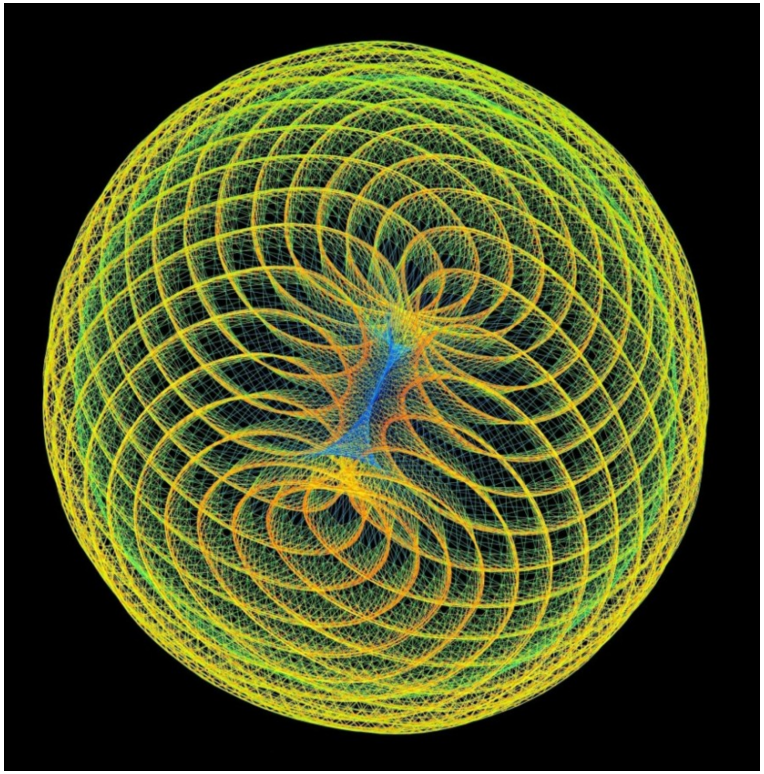
$$i = [1 \quad 1 \quad 1 \quad 2 \quad 2 \quad 3 \quad 3 \quad 3 \quad 3]$$

$$j = [1 \quad 2 \quad 5 \quad 2 \quad 3 \quad 1 \quad 3 \quad 4 \quad 5]$$

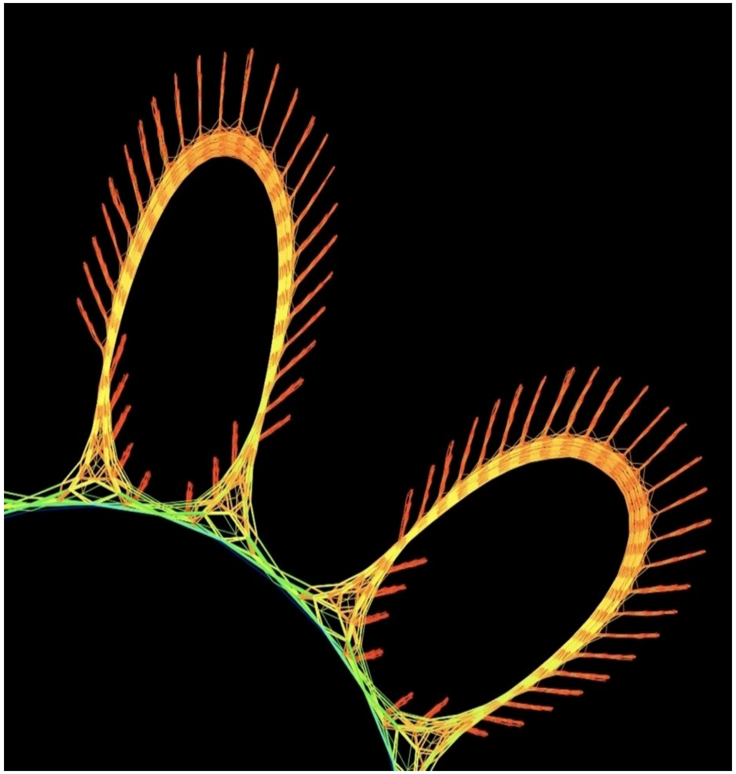
$$s = [11 \quad 12 \quad 15 \quad 22 \quad 23 \quad 31 \quad 33 \quad 34 \quad 35],$$

$$a = \text{sparse} \quad (i,\ j,\ s,\ m,\ n,\ nz\_max)$$

Examples of Sparse Matrices

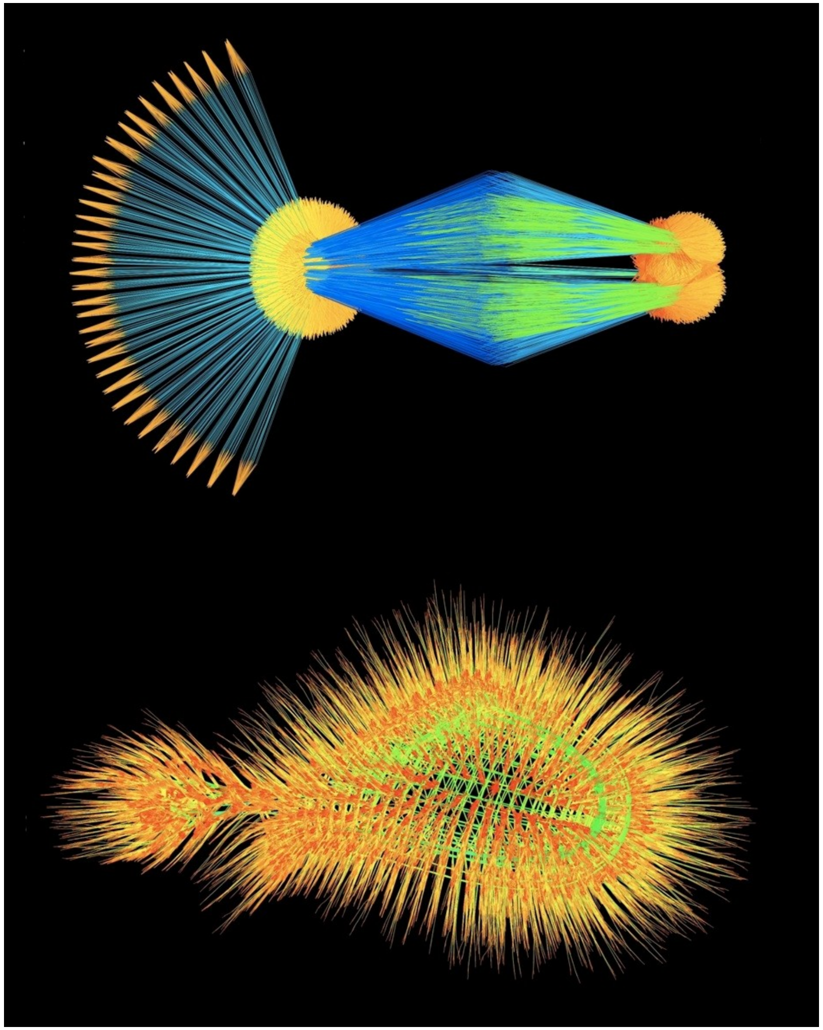


A Barrier Hessian Matrix



A Matrix From Financial Portfolio Optimization

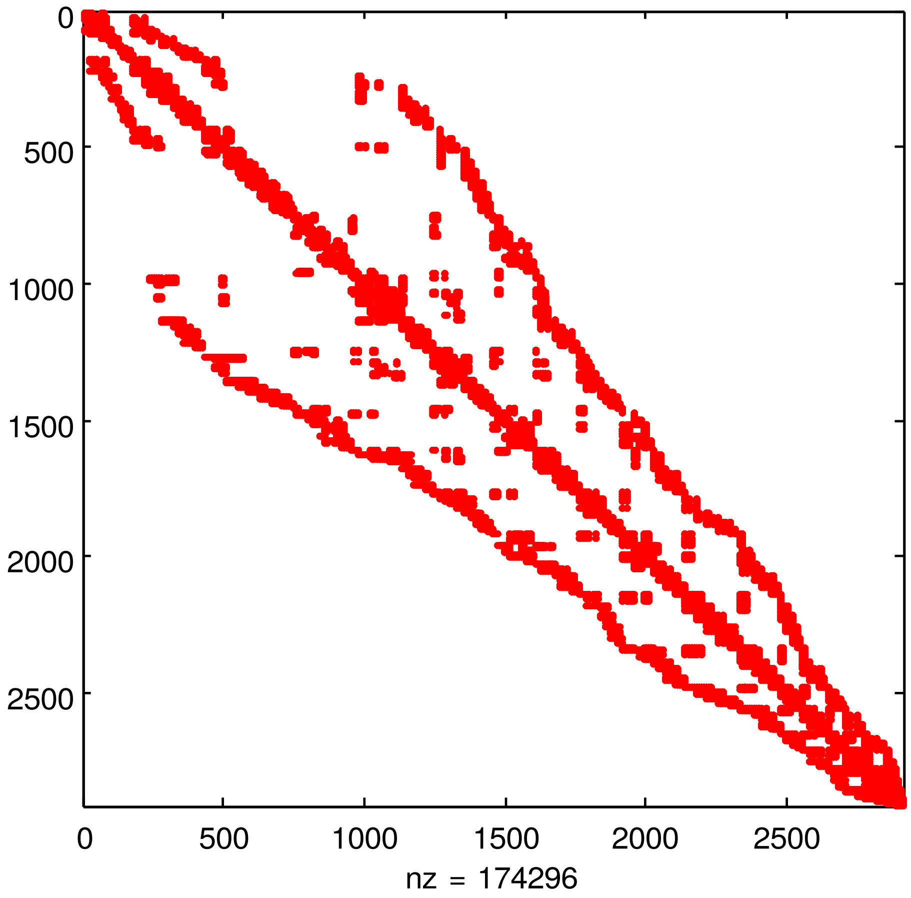
Examples of Sparse Matrices



http://www.cise.ufl.edu/~davis/matrices.html

The Sparse Matrix

$$\text{Dimension}: 2910 \times 2910 \quad \text{NNZ}: 174296$$



MATLAB Command

- [X,FLAG, RELRES, ITER]=gmres (A, B, RESTART, TOL, MAXIT, M)
- [X,FLAG, RELRES, ITER]=Y(A, B, TOL, MAXIT, M)
- Y for cgs, pcg, bicgstab.
- FLAG = 0 , convergent to the desire TOL within MAXIT.
- FLAG=1 , iterated MAXIT times, did not converge.
- FLAG=2 , preconditioner M was ill-conditioned.
- FLAG=3 , stagnated.
- FLAG=4 , one of the scalar quantities calculated during the method Y became too small or too large to continue computing.

Efficiency of Algorithms

	GMRES		CGS	
	Preconditioned	Without Preconditioned	Preconditioned	Without Preconditioned
FLAG	0	0	3	4
ITER	6510	37007	2070	2
RELRES	9.943e-07	1.000e-6	3.0126e-04	0.231

Efficiency of Algorithms

	PCG		BICGSTAB	
	Preconditioned	Without Preconditioned	Preconditioned	Without Preconditioned
FLAG	0	0	0	0
ITER	887	3470	1266.5	8061.5
RELRES	9.8646e-07	9.7e-07	1.55e-07	8.48e-07

Conclusion

- The Preconditioned Conjugate Gradient Method gives the better result.
- If the matrix is not in symmetric positive definite, then this method may not works.
- For unsymmetric matrix, GMRES provides the better result.

Thank You