Iterative Methods for Solving Linear Systems and Modeling Traffic on I-485

Fabiola Rojas and Dominic Kealoha, UNC Charlotte
Dr. Helen Li, College of Science

UNIVERSITY OF NORTH CAROLINA CHARLOTTE

Introduction

Our Objective:

- Use iterative methods to solve a large and singular (non-invertible) linear system.
- Model the flow of traffic volume through each exit on I-485's inner loop.
- Identify the most congested sections of the highway.

What Are Iterative Methods?

A method for finding approximate solutions to a system of linear equations ***via matrix multiplications.*** These iterative methods first make an initial guess for the solution, and then refine the guess through multiplication with a unique iterative matrix. These iterations continue until a satisfactory solution is found within a residual threshold.

Why Use Iterative Methods?

- These methods are time efficient, reaching convergence far sooner than direct methods when the problem size is huge.
- Can be used in cases where a matrix is too large to be solved without difficulty using direct methods and still have accurate results.
- Able to solve singular systems which could not be solved via direct methods.

Spectral Radius and Convergence

- From a linear system of equations $A\mathbf{x}=\mathbf{b}$, we can derive the iterative algorithm $\mathbf{x}^{(k)}=T\mathbf{x}^{(k-1)}+c$ and use an iterative method to converge to a solution.
- We can decide which method to use by either studying the structure of the matrix, or by finding the **spectral radius** for each method, which is the maximum of the absolute values of T's eigenvalues.
  - The smaller the spectral radius, the faster it will converge.
- **Convergence** is characterized by the residual between each iteration reaching a predefined error threshold, where the residual is defined by $\mathbf{r}=\mathbf{b}-A\mathbf{x}$.
  - The residual between each iteration should ideally get exponentially smaller.

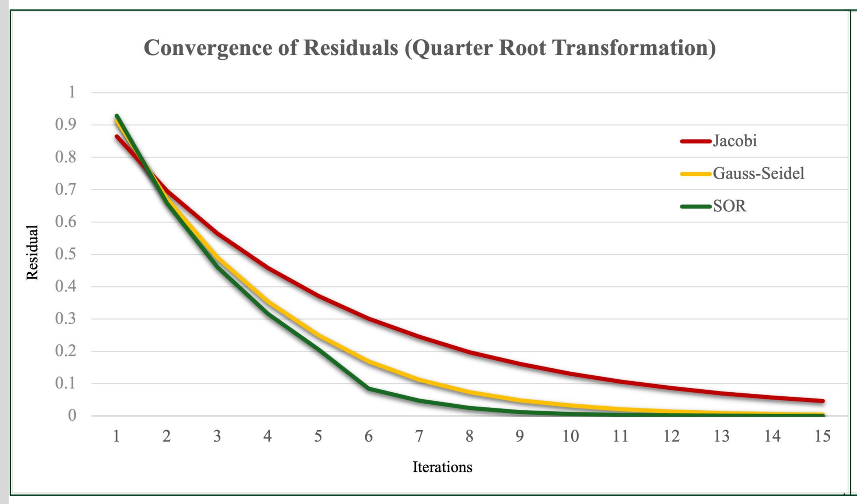


Illustration of convergence by showing how residual gets smaller after each iteration.

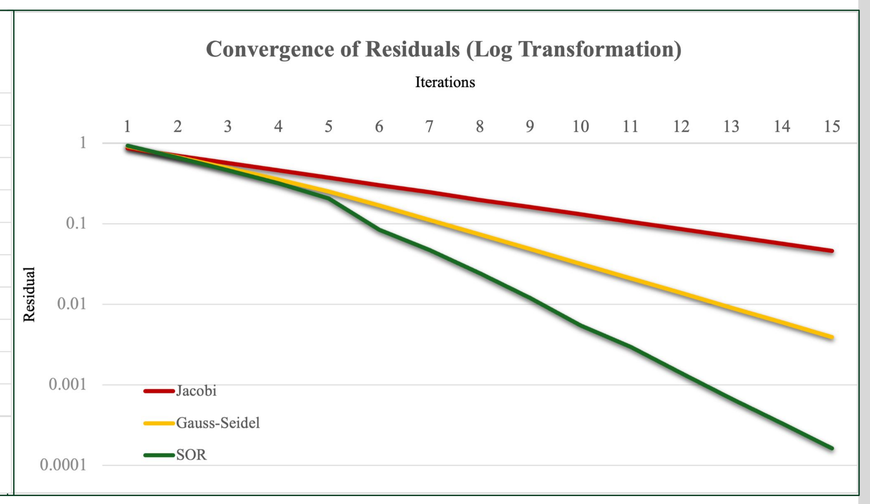
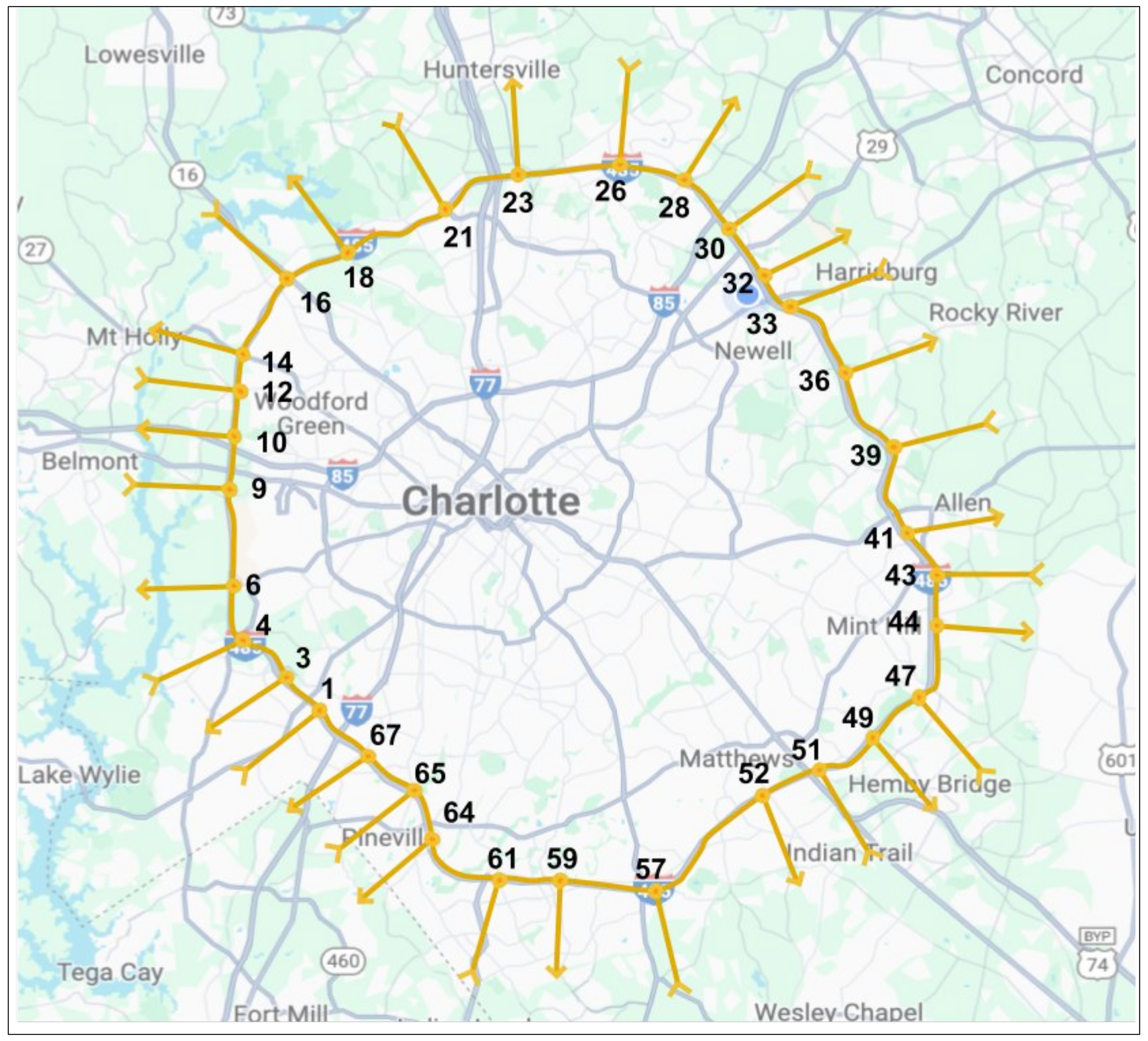


Illustration of convergence rate by showing how they converge exponentially fast, and how SOR converges the fastest.

After studying three iterative methods—Jacobi, Gauss-Seidel, and Successive Over-Relaxation (SOR)—we learned that SOR with optimal parameters is ideal:

- Fastest convergence rate
- Smallest spectral radius
- Weight parameter ω speeds up iterations
- Guarantees convergence for a symmetric, positive-definite, tridiagonal matrix

Modeling Traffic on I-485



Flow network demonstrating traffic through the inner loop of I-485 (clockwise), with alternating input and output data (via NCDOT) at each exit.

The matrix A that results from this flow network returns a spectral radius of 1, which makes it ineligible for iterative methods. To solve this problem, we can utilize the fact that it has an eigenvalue of 0 and eigenvector of [1 1 … 1 1]. This requires us to:

- delete the last column of matrix A → resulting in a new 32x31 matrix $\tilde{A}$,
- delete the last row of $\mathbf{x}$,
- define a new linear system $\tilde{A}\tilde{\mathbf{x}}=\mathbf{b}$,
- ensure a square coefficient matrix by multiplying each side of the system by the transpose of matrix $\tilde{A}$ → resulting in a 31x31 matrix and 31x1 vectors $\tilde{\mathbf{x}}$ and $\mathbf{b}$.

$$\begin{bmatrix} 1 & -1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & -1 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 & -1 \\ -1 & 0 & 0 & \dots & 0 & 0 & 1 \end{bmatrix} \longrightarrow \begin{bmatrix} 2 & -1 & 0 & \dots & 0 & 0 & 0 \\ -1 & 2 & -1 & \dots & 0 & 0 & 0 \\ 0 & -1 & 2 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 2 & -1 & 0 \\ 0 & 0 & 0 & \dots & -1 & 2 & -1 \\ -1 & 0 & 0 & \dots & 0 & -1 & 2 \end{bmatrix}$$

Resulting 32x32 matrix A, starting at the node at exit 1 and circling around to exit 67.

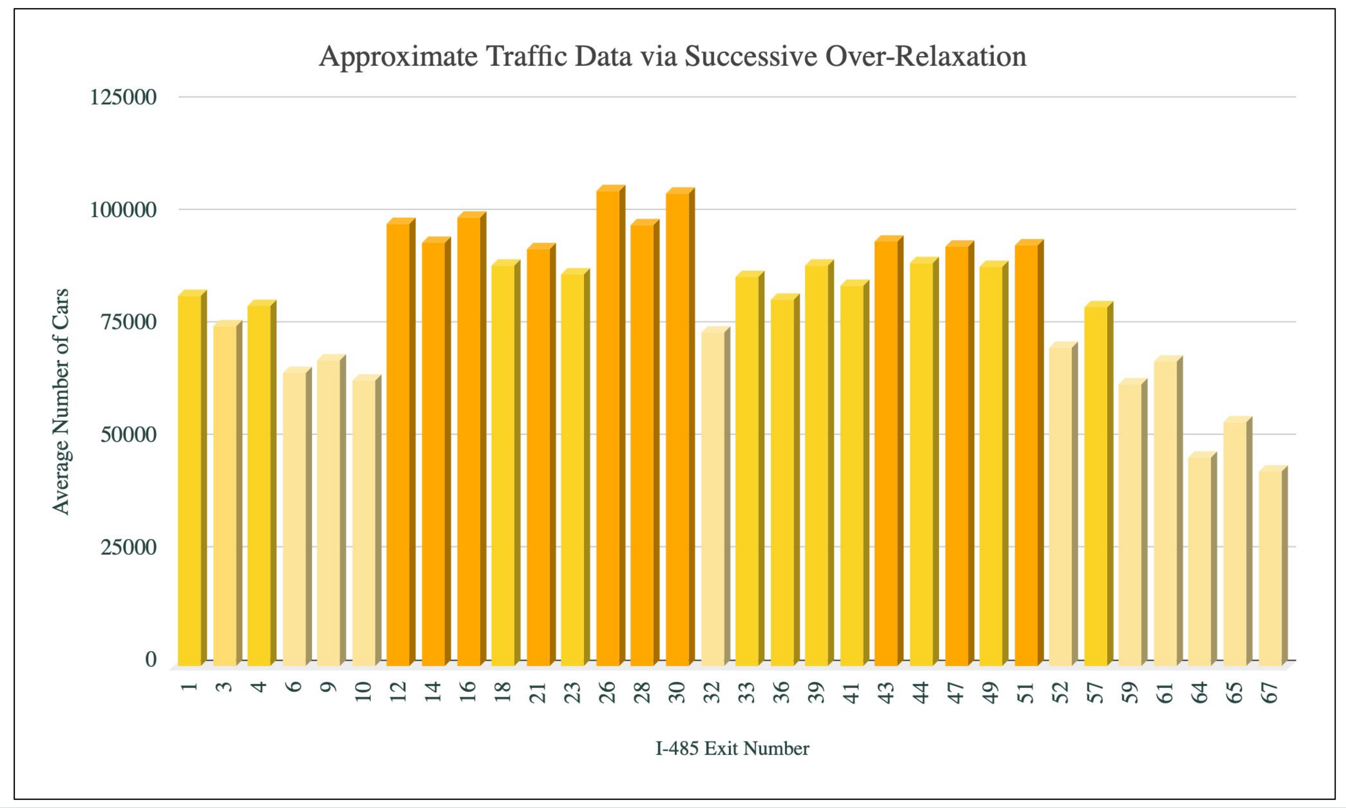
Modified matrix $\tilde{A}$ with spectral radii < 1:
SOR: 0.8215
Gauss-Seidel: 0.9904
Jacobi: 0.9952

- **We implement our python code to calculate the values for $\tilde{\mathbf{x}}$.** The number of cars per day along each segment of I-485 is actually determined by adding the 32nd element c back onto each element of $\tilde{\mathbf{x}}$.
- The c value would be the median of the $\tilde{\mathbf{x}}$ values.

Collected Data and Results

- To calculate $\tilde{\mathbf{x}}$, the average number of cars on each highway segment, we used the SOR iterative method (ω≈1.8) because its iterative matrix T has the smallest spectral radius, meaning it will converge faster than the other iterative methods we studied.
- This approximation was derived from the number of cars coming in and out of the exits surrounding the highway segment.
- **The busiest segments span exits 26-30, which are near the UNCC area.**

	Spectral Radius	# of Iterations
Jacobi	0.9952	3822
Gauss	0.9904	1511
SOR	0.8215	99



The approximate number of cars per day traveling between each exit along I-485.

Conclusions

- Iterative methods are more reasonable than direct methods for solving large (singular) linear systems in real world applications.
- Network flows can create possibilities for applications regarding traffic data, and iterative methods make it possible to process that much data.
- The most congested areas appear to be in the northern and eastern hemispheres of I-485, indicating greater population density or lack of alternative routes.
- This matrix is scalable and can be used to determine new infrastructure opportunities or alternative routes in case of exit shutdown.

References

Chapter 5 Iterative Methods for Solving Linear Systems. Courtesy of the University of Pennsylvania, accessed 25 March 2024, https://www.cis.upenn.edu/~cis5150/cis515-12-sl5.pdf

Lay, D. C., Lay, S. R., & McDonald, J. (2016). Linear algebra and its applications. Pearson.

NCDOT. (n.d.). NCDOT Annual Average Daily Traffic (AADT) Mapping Application. ArcGIS web application. https://ncdot.maps.arcgis.com/apps/webappviewer/index.html?id=964881960f0549de8c3583bf46ef5ed4

7.3 The Jacobi and Gauss-Seidel Iterative Methods. Courtesy of the University of Notre Dame, accessed 25 March 2024, https://www3.nd.edu/~zxu2/acms40390F12/Lec-7.3.pdf.

7.4 Relaxation Techniques for Solving Linear Systems. Courtesy of the University of Notre Dame, accessed 25 March 2024, https://www3.nd.edu/~zxu2/acms40390F13/Lec-7.4-5.pdf.

This project is supported by the NSF CAREER Award: DMS-1847770.