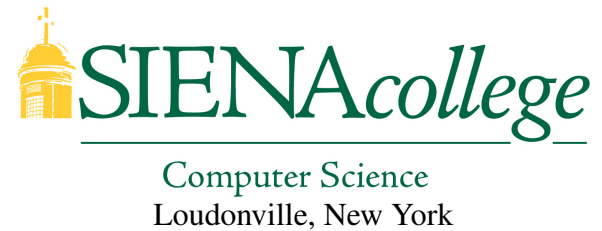


Teaching the Divide-and Conquer Closest Pair Algorithm Using a Map-Based Visualization

Jim Teresco



Nifty Assignment Presentation

37th CCSC Southeastern Regional Conference
Coastal Carolina University

November 4, 2023

Yet another Powerpoint-free presentation!

Agenda

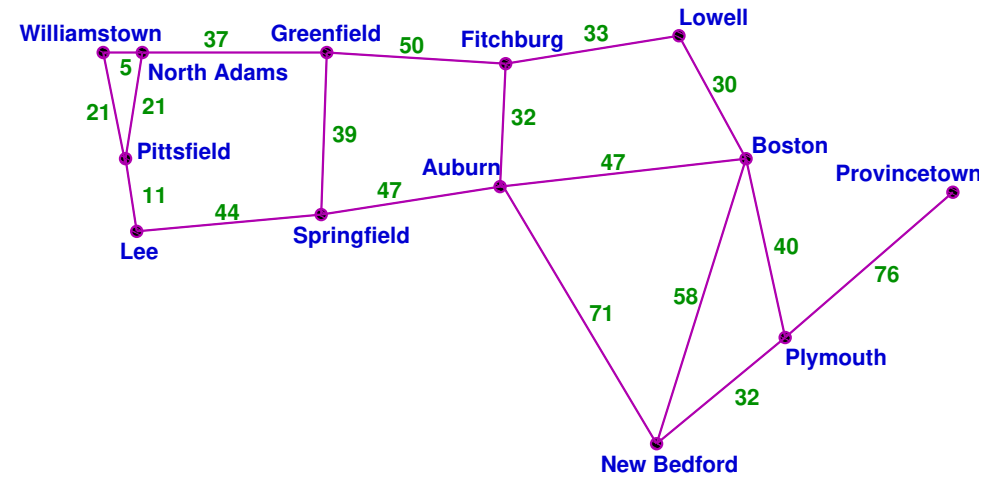
- A quick overview of METAL
 - Motivations
 - Travel Mapping
 - Bringing my “roadgeek” side to work: using Travel Mapping
 - Graph data derived from Travel Mapping project
 - Highway Data Examiner
 - METAL’s interactive algorithm visualizations
- METAL’s closest pairs learning module
- Wrapup/acknowledgements



These slides are at <https://courses.teresco.org/metal/talk.pdf>
and the big QR code above will take you there.

Early Motivation

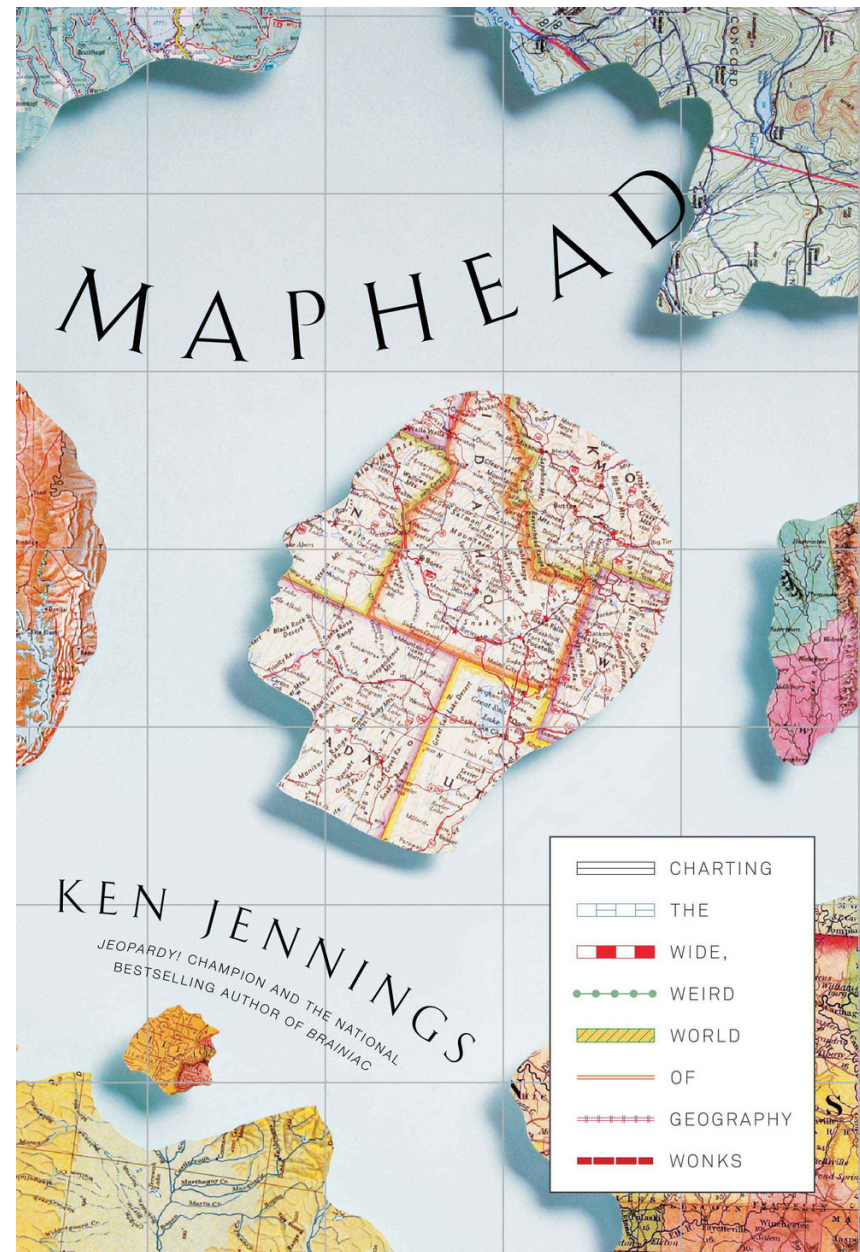
To find a good collection of data sets for teaching graph (and other) algorithms.



I want an example that's better than this!

- Variety of data sizes
 - small sets to be manageable for testing during development
 - larger sets to be interesting and for performance analysis
- Data source
 - freely and conveniently available data sets
 - a motivating real-world connection
- Meaningful visualizations of results

Before I was a CS Geek - I was a Map Geek

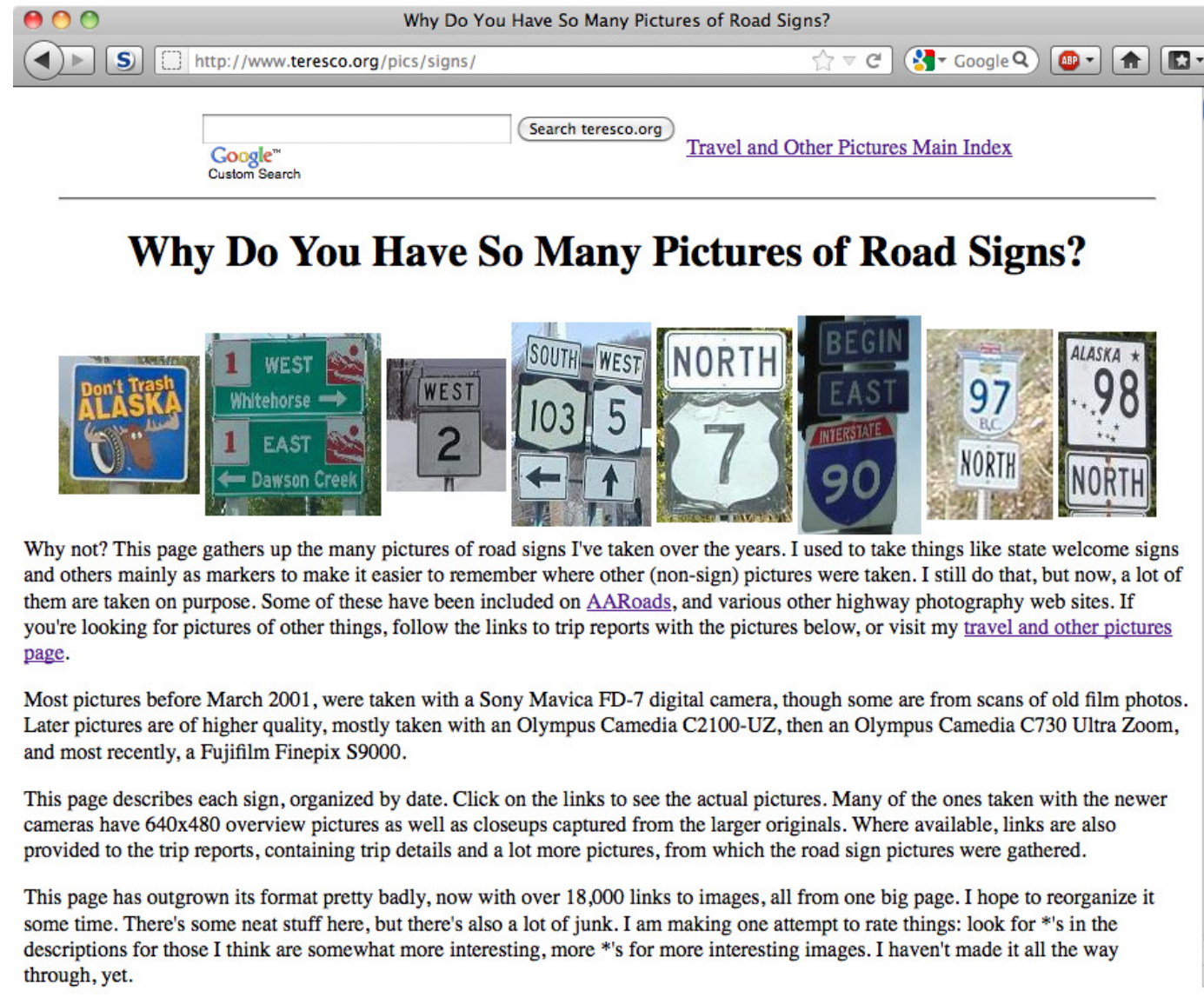


The Well-Worn Rand McNally Road Atlas Collection



Each marked up with a highlighter to track travels.

And I Used Highway Photography to Document my Travels



The screenshot shows a web browser window with the title "Why Do You Have So Many Pictures of Road Signs?". The address bar contains the URL "http://www.teresco.org/pics/signs/". Below the browser window, the page content includes a search bar with the text "Search teresco.org" and a link to "Travel and Other Pictures Main Index". The main heading is "Why Do You Have So Many Pictures of Road Signs?". Below the heading is a row of eight road signs: a "Don't Trash ALASKA" sign, a green directional sign for Whitehorse and Dawson Creek, a white directional sign for West, a white directional sign for South and West, a white directional sign for North, a blue and red Interstate 90 sign, a white directional sign for 97 North, and an Alaska state welcome sign for Highway 98 North. Below the signs is a paragraph of text explaining the purpose of the page and the equipment used to take the photos. The text is as follows:

Why not? This page gathers up the many pictures of road signs I've taken over the years. I used to take things like state welcome signs and others mainly as markers to make it easier to remember where other (non-sign) pictures were taken. I still do that, but now, a lot of them are taken on purpose. Some of these have been included on [AARoads](#), and various other highway photography web sites. If you're looking for pictures of other things, follow the links to trip reports with the pictures below, or visit my [travel and other pictures page](#).

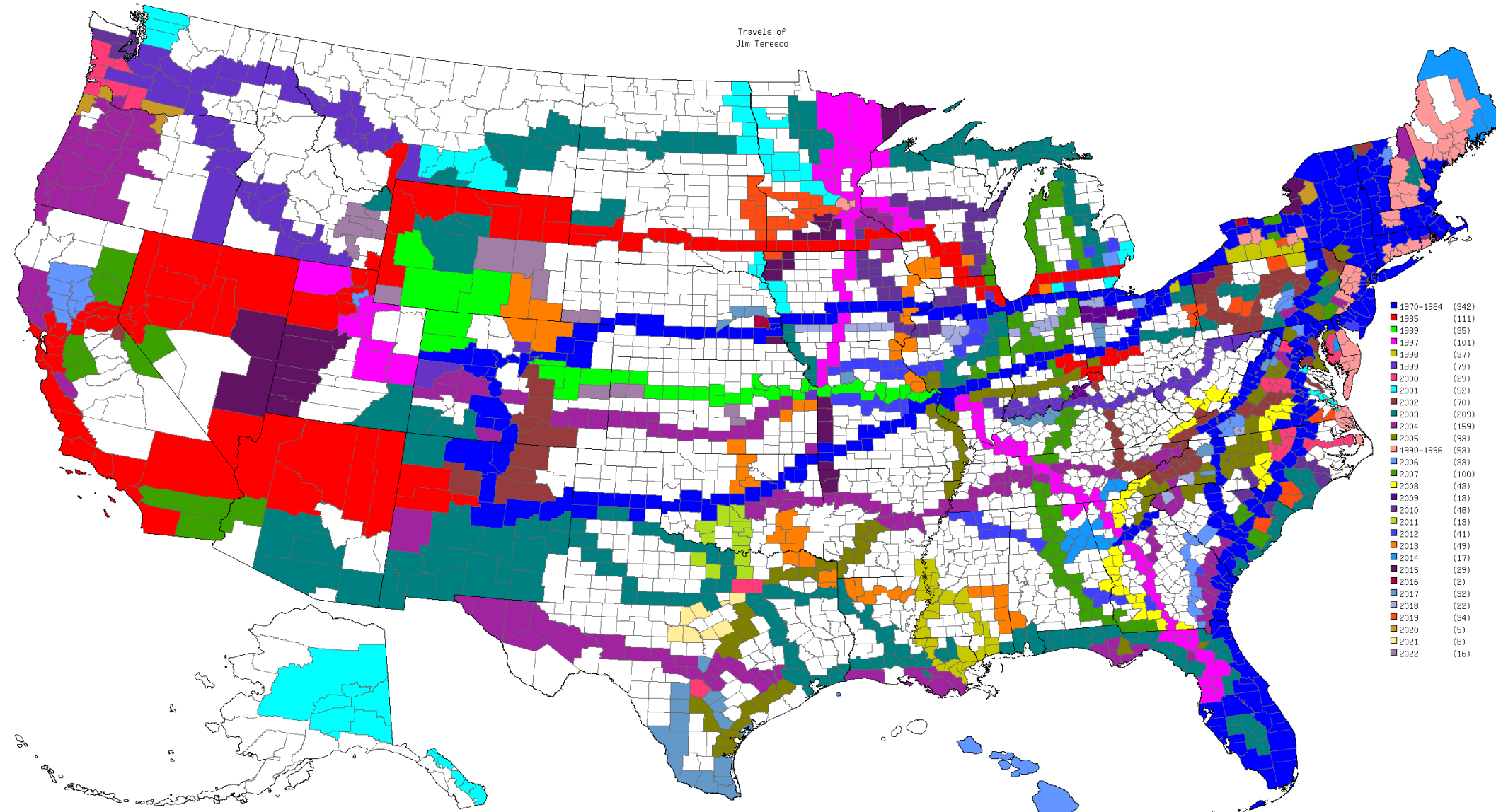
Most pictures before March 2001, were taken with a Sony Mavica FD-7 digital camera, though some are from scans of old film photos. Later pictures are of higher quality, mostly taken with an Olympus Camedia C2100-UZ, then an Olympus Camedia C730 Ultra Zoom, and most recently, a Fujifilm Finepix S9000.

This page describes each sign, organized by date. Click on the links to see the actual pictures. Many of the ones taken with the newer cameras have 640x480 overview pictures as well as closeups captured from the larger originals. Where available, links are also provided to the trip reports, containing trip details and a lot more pictures, from which the road sign pictures were gathered.

This page has outgrown its format pretty badly, now with over 18,000 links to images, all from one big page. I hope to reorganize it some time. There's some neat stuff here, but there's also a lot of junk. I am making one attempt to rate things: look for *'s in the descriptions for those I think are somewhat more interesting, more *'s for more interesting images. I haven't made it all the way through, yet.

(Knuth has road sign pictures, so why not me?)

And I Like to “Collect” Things



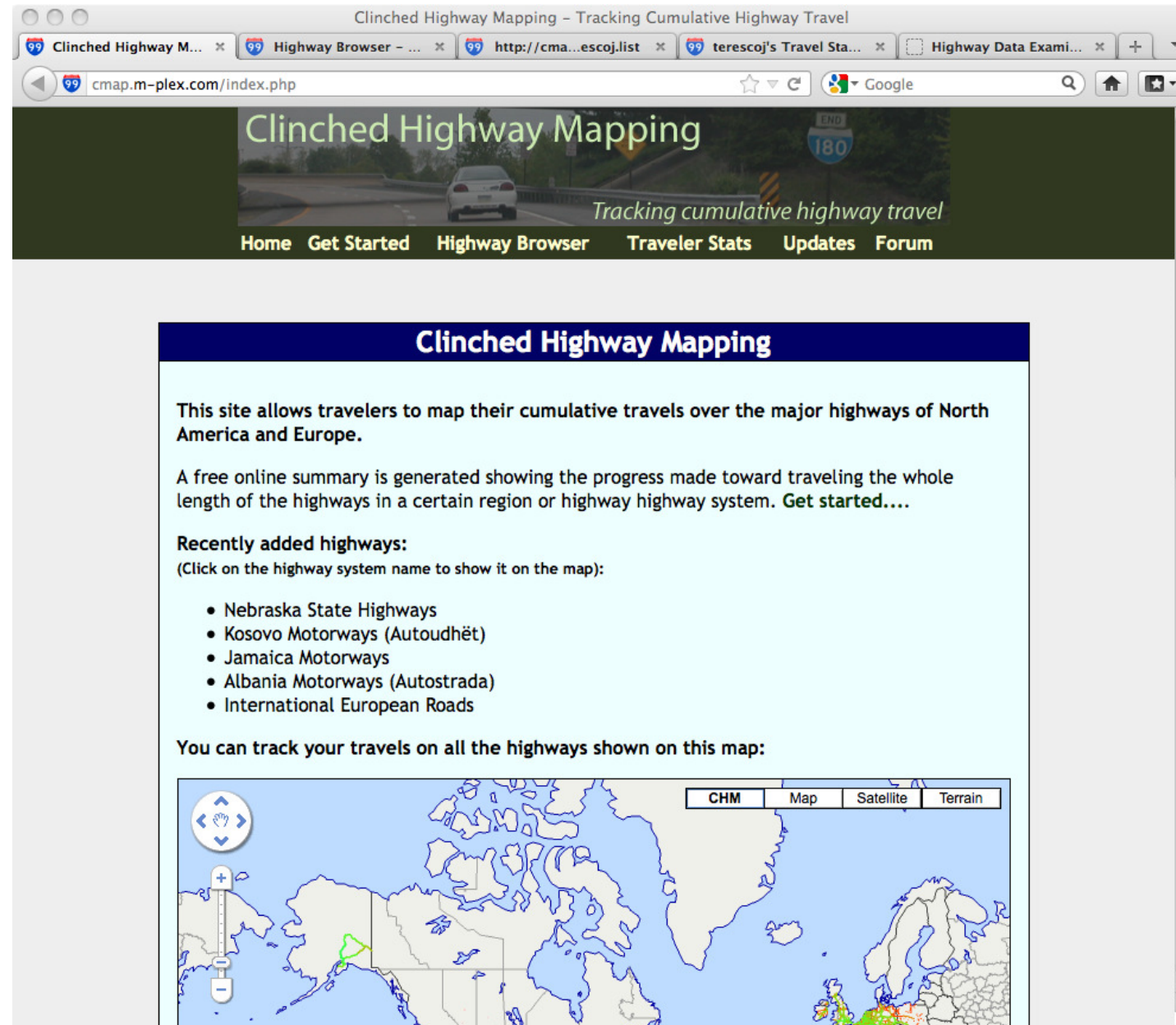
My current
county count:
1875 of 3144
(59.7%)

(no new ones on this trip..)

<https://mob-rule.com/home>

As a “Roadgeek”, I Could Not Resist

The Clinched Highway Mapping Project



The screenshot shows a web browser window with the title "Clinched Highway Mapping - Tracking Cumulative Highway Travel". The address bar shows "cmap.m-plex.com/index.php". The page features a dark green header with the title "Clinched Highway Mapping" and a navigation menu with links: Home, Get Started, Highway Browser, Traveler Stats, Updates, and Forum. Below the header is a light blue content area with the following text:

Clinched Highway Mapping

This site allows travelers to map their cumulative travels over the major highways of North America and Europe.

A free online summary is generated showing the progress made toward traveling the whole length of the highways in a certain region or highway highway system. **Get started....**

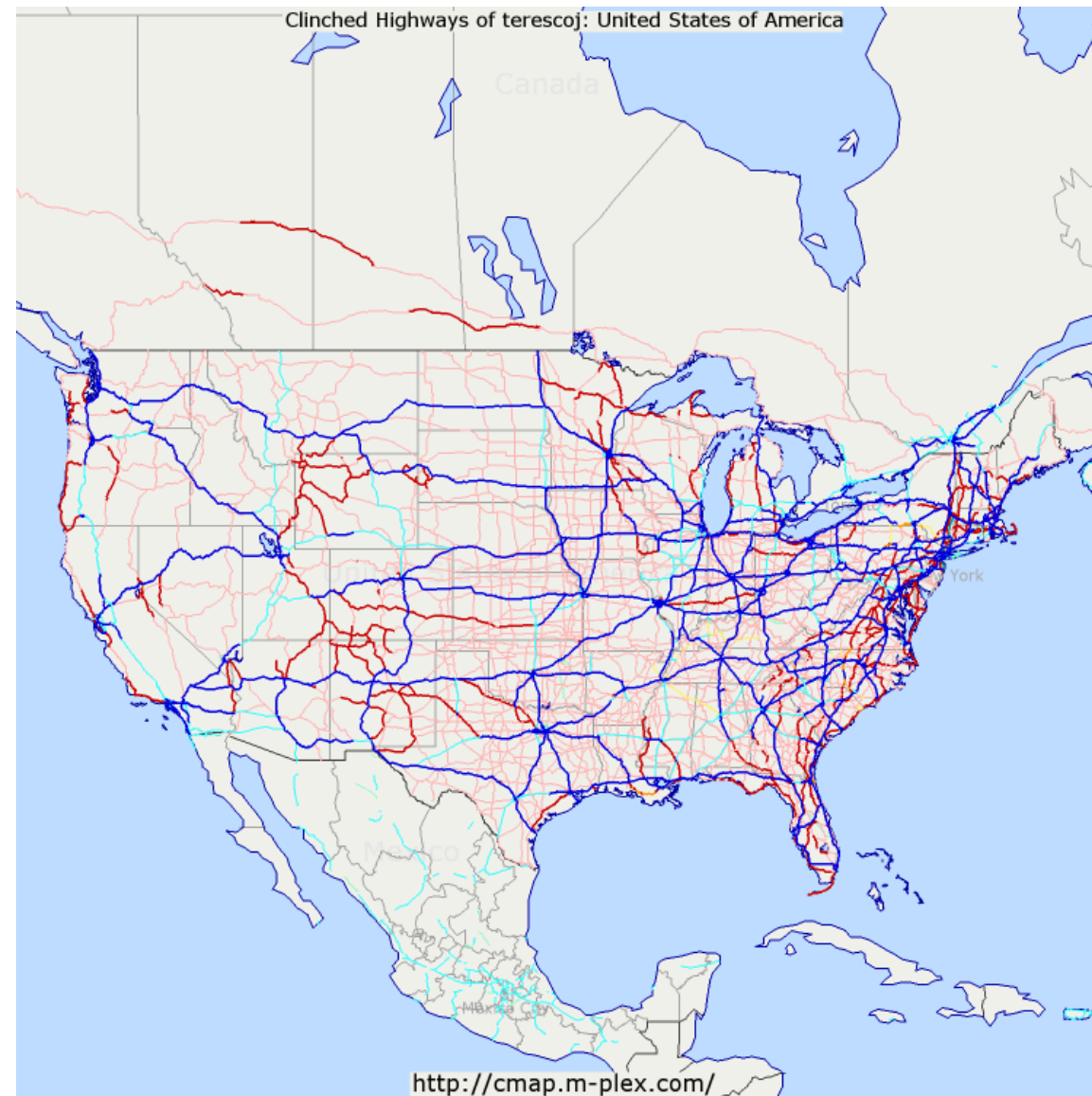
Recently added highways:
(Click on the highway system name to show it on the map):

- Nebraska State Highways
- Kosovo Motorways (Autoudhët)
- Jamaica Motorways
- Albania Motorways (Autostrada)
- International European Roads

You can track your travels on all the highways shown on this map:

Below the text is a map of North America and Europe. The map includes a navigation control on the left and a legend at the top right with buttons for "CHM", "Map", "Satellite", and "Terrain". The map shows a green line indicating a travel path across North America.

To Track Highway Segments I've Traveled



Bringing my “Roadgeek” Side to Work

The observations, questions, and ideas that have led to the development of the tools that have evolved into today’s

Map-Based Educational Tools for Algorithm Learning (METAL)

- Hey, some of my data structures examples and labs are pretty boring...
- Could I use CHM (now TM) data for class examples and assignments?
- If I could do that, could I plot it on maps like CHM does?
- Can I do all this before the Dijkstra’s algorithm lab I am not really happy with that I need to assign next week?
- Hmm, Dijkstra’s algorithm. I think I need graph data, not a bunch of individual routes.
- Who needs sleep for the next week? I can do this! I’ll create graphs and build a visualization tool too!
- All these years later, I’m still working on it!

CHM → Travel Mapping

- Clinched Highway Mapping went dormant in 2014...
 - my role there was as a data contributor and user, but not administrator
 - the site exists, but is not actively maintained
 - CHM users and contributors decided to build a replacement
- Summer 2015: Birth of the Travel Mapping project
 - main site: <https://travelmapping.net/>
 - GitHub organization: <https://github.com/TravelMapping>
 - my role is as primary developer and project manager
 - highway data is constantly improving and expanding: now over 66,000 commits to the highway data repository by 24 contributors
- Site users create and submit list of highway segments, e.g.,
https://github.com/TravelMapping/UserData/blob/master/list_files/terescoj.list
- Enjoy the maps and statistics of your and others' travels!
<https://travelmapping.net/user/mapview.php?u=terescoj&rg=SC>

METAL Graph Data

METAL generates graphs from TM data for many countries, regions, highway systems, and local areas.

<https://travelmapping.net/graphs/>

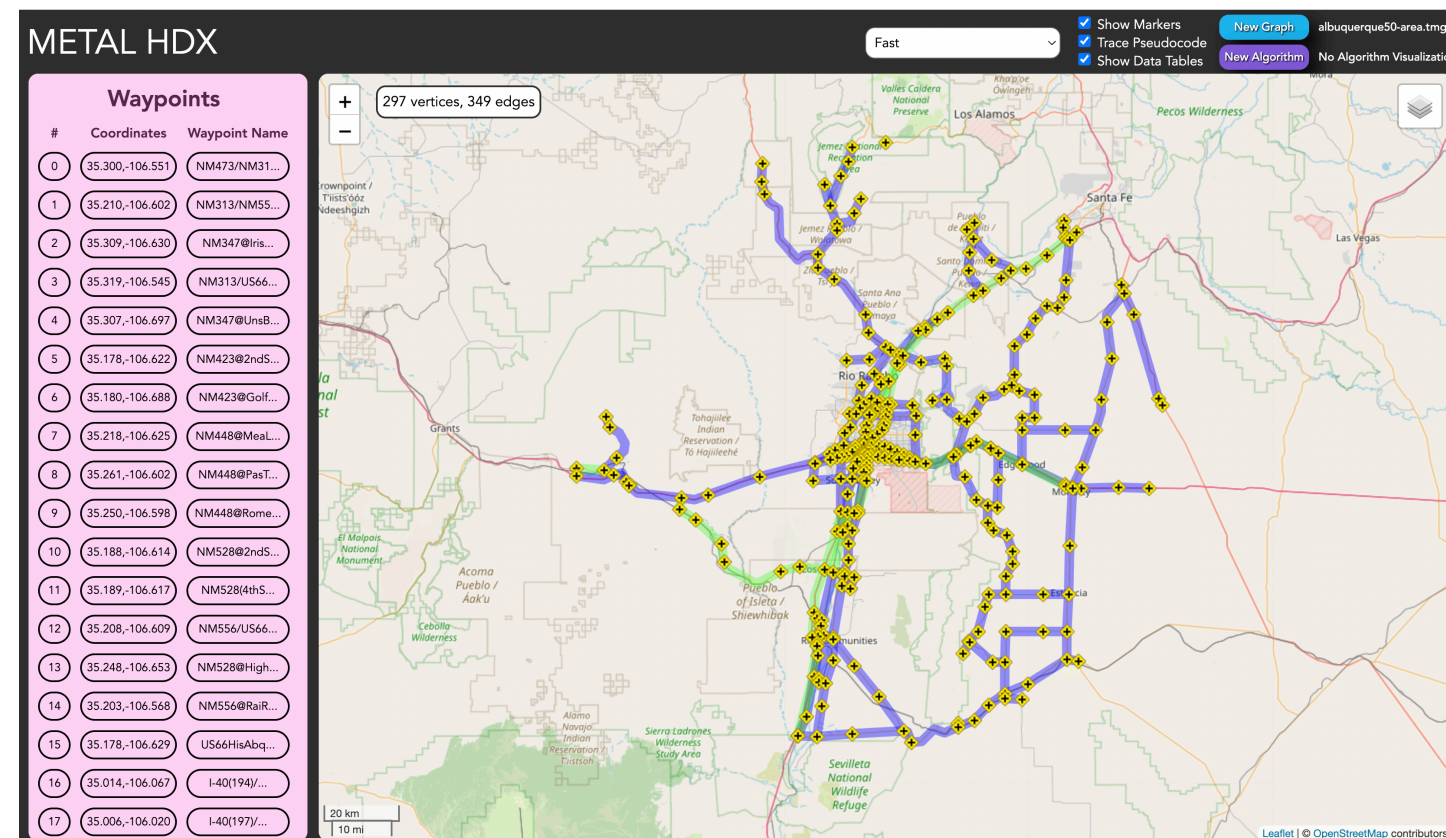
A selection from among the 1200+ graphs provided by METAL:

Data set	$ V $	$ E $
ABW-region: All Aruba highways	28	37
DC-region: All District of Columbia highways	97	94
YT-region: All Yukon highways	225	221
albuquerque50-area: All plotted routes within 50 miles of Albuquerque	298	350
siena50-area: All plotted routes within 50 miles of Siena College	1244	1464
SC-region: All South Carolina highways	3596	4575
west-indies: All plotted routes in the West Indies	4697	6027
usai-system: United States Interstate highway system	18,342	18,735
USA-national: All plotted NHS routes in the U.S.	73,055	78,266
USA-country: All plotted (state and national) highways in the U.S.	185,177	220,534
tm-master: All plotted highways Worldwide	625,257	767,013

Graphs are archived periodically to avoid disruptive changes.

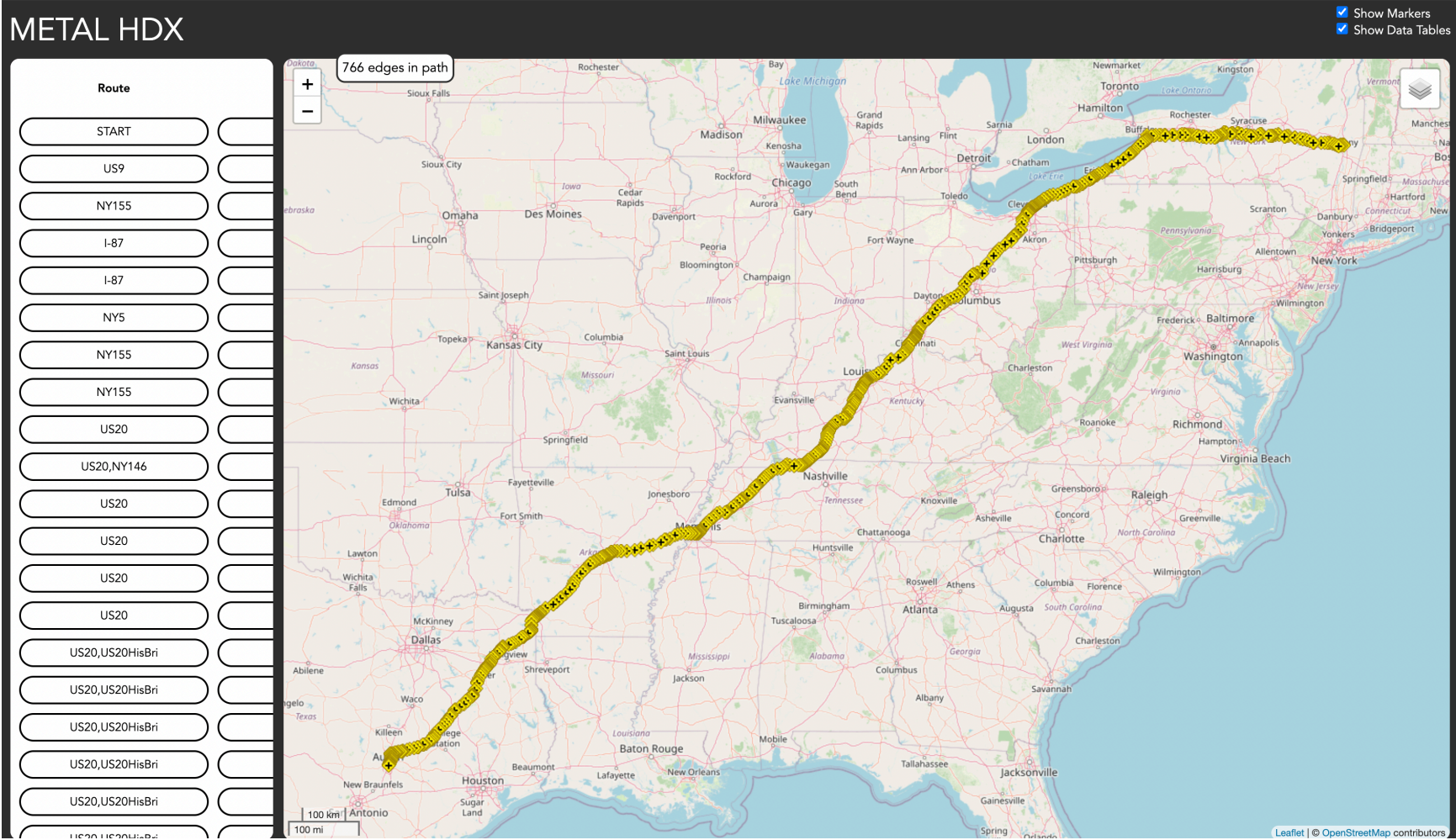
Visualizing the Results: Leaflet and OpenStreetMaps

- Leaflet provides a free application programmer interface (API) to overlay custom data on OpenStreetMap (and other) map tiles
 - API's Javascript classes and methods are fairly easy to learn and use
 - Formerly used Google Maps, but free non-commercial use terminated
- METAL's **Highway Data Examiner (HDX)** overlays TM/METAL data
 - Graph of plotted highways within 50 miles of Albuquerque:



A Shortest Path Result

HDX can also visualize the results of programs that use METAL data.



Shortest path from Siena College to Austin, Texas, generated using “Dijkstra’s Road Trip” reference solution, displayed in HDX as a “pth” file.

METAL's Interactive Algorithm Visualization

- Interactive visualizations of algorithms rather than just the data and results
- The focus of a 2018 SIGCSE paper
- Implemented in HDX

METAL HDX

Resume Very Fast

Show Markers Trace Pseudocode Show Data Tables

New Graph DE-region.tmg
New Algorithm Dijkstra's Algorithm

Algorithm Visualization Status

Pseudocode

```
pq ← new Priority Queue
pq.add(start, null, 0)
while not tree.contains(end)
  if pq.isEmpty
    error: no path
  (to, via, d) ← pq.remove()
  if tree.contains(to)
    discard (to, via) // on removal
  else
    tree.add(to, via, d)
    for each e=(to, v) // neighbors
      if tree.contains(v)
        discard (v, e) // on discovery
      else
        pq.add(v, e, d+len(e))
```

Visiting #216 DE36@MenSchRd found via DE36

Undiscovered: 692 V, 834 E

Priority Queue size = 10; max = 10; avg = 4.8

216→658	218→663	259→213	215→289
DE36	US13	DE16	DE16, DE36
20.023	19.735	16.672	16.633

Spanning Tree: 33 V, 32 E

Discarded on discovery: 3 E

Discarded on removal: 3 E

32 Shortest Paths Found So Far

Place	Distance	Arrive From	Via
DE18..D/DE	0.000		(START)
DE18@CalRd	1.034	DE18..D/DE	DE18
DE18@AtIRd	1.973	DE18@CalRd	DE18
DE18..raRd	3.459	DE18@AtIRd	DE18

735 vertices, 879 edges

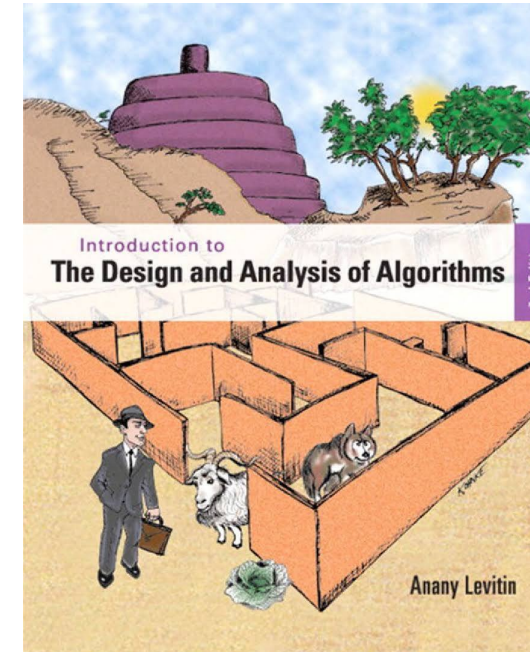
10 km 5 mi

Leaflet | © OpenStreetMap contributors

My Nifty Assignment

Divide and Conquer Closest Pairs AV and Learning Module

- Goal: learn the divide and conquer closest pairs algorithm more quickly and at a deeper level
- 60-90 minute in-class, lab, or homework activity
- First used in Spring 2023 Analysis of Algorithms course
- Based on the algorithm as presented in Levitin's text



Let's go to the learning module



<https://bit.ly/47hjV3q>

Key METAL Project Components

We have seen METAL's main components.

1. Large set of graphs for a variety of highway systems generated from TM highway data
 - real world connection and visualization on maps
 - wide variety of sizes
2. HDX to visualize graphs
 - graph data as provided by METAL
 - also can visualize data in other formats generated by programs that use the graphs, such as shortest paths, sets of points, and lists of points
3. Interactive algorithm visualizations within HDX
 - Our experience: the *scalability* matters: small graphs to see the details and larger to see big picture behavior (think: BFS vs. DFS)

Recent METAL Enhancements

- Graph variants and file formats
 - graph files with traveler data
 - partitioned graph file format
 - intersection-only graphs
- User interface and usability
 - redesigned and more intuitive interface
 - conditional breakpoints
 - more execution speeds
- Many algorithm visualizations
 - see current list at the right
- Instructor support
 - graph archives
 - query string parameters to load graphs, select AVs and parameters
 - just started: creation of learning modules

Vertex-Only Algorithms

- Vertex Extremes Search
- Vertex Closest/Farthest Pairs
- Brute-Force Convex Hull
- All Points Closest Pairs
- Vertex Closest Pairs Recursive
- Quadtree Construction
- Space-Filling Curve Traversals
- Brute Force Traveling Salesman
- Twice-Around-the-Tree TSP Approximation
- Recursive Coordinate Bisection Partitioner
- Compute Partition Stats

Edge-Only Algorithms

- Edge Extremes Search

Graph Algorithms

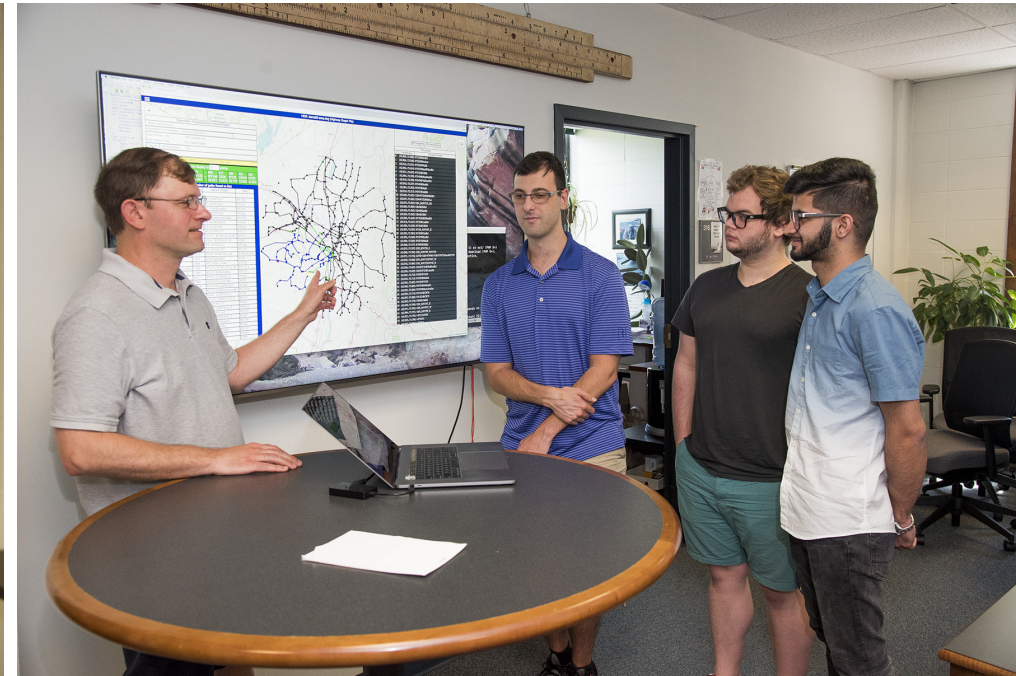
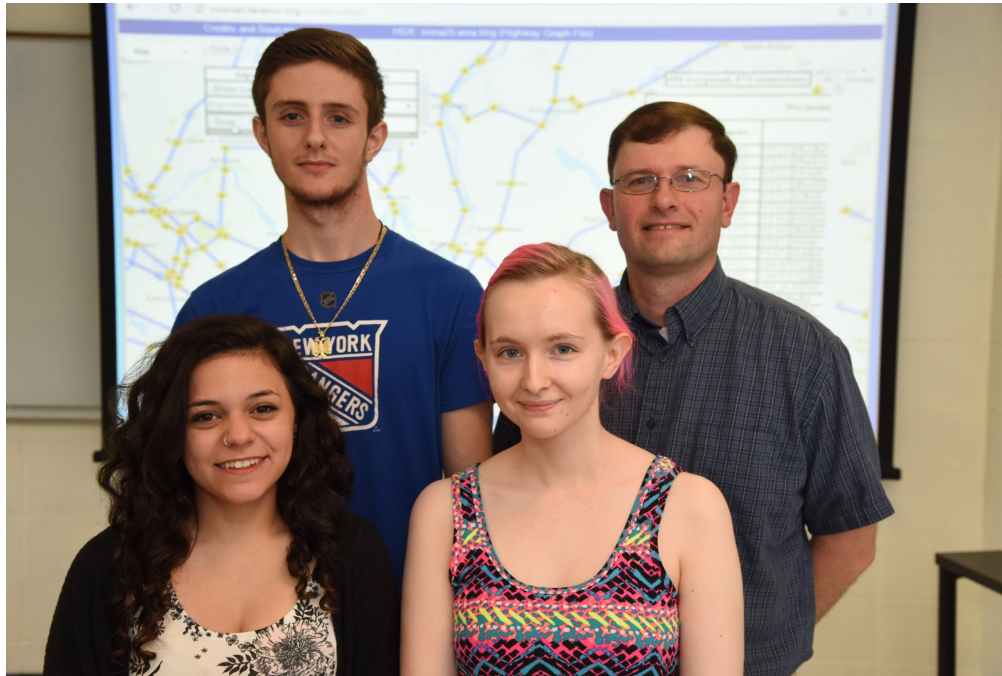
- Vertex Min/Max Degree Search
- Graph Traversals/Connected Components
- ✓ Dijkstra's Algorithm
- Prim's Algorithm
- Kruskal's Algorithm
- Recursive Depth First Search
- Simple Bridge Detection
- Welsh-Powell Graph Coloring

Acknowledgements

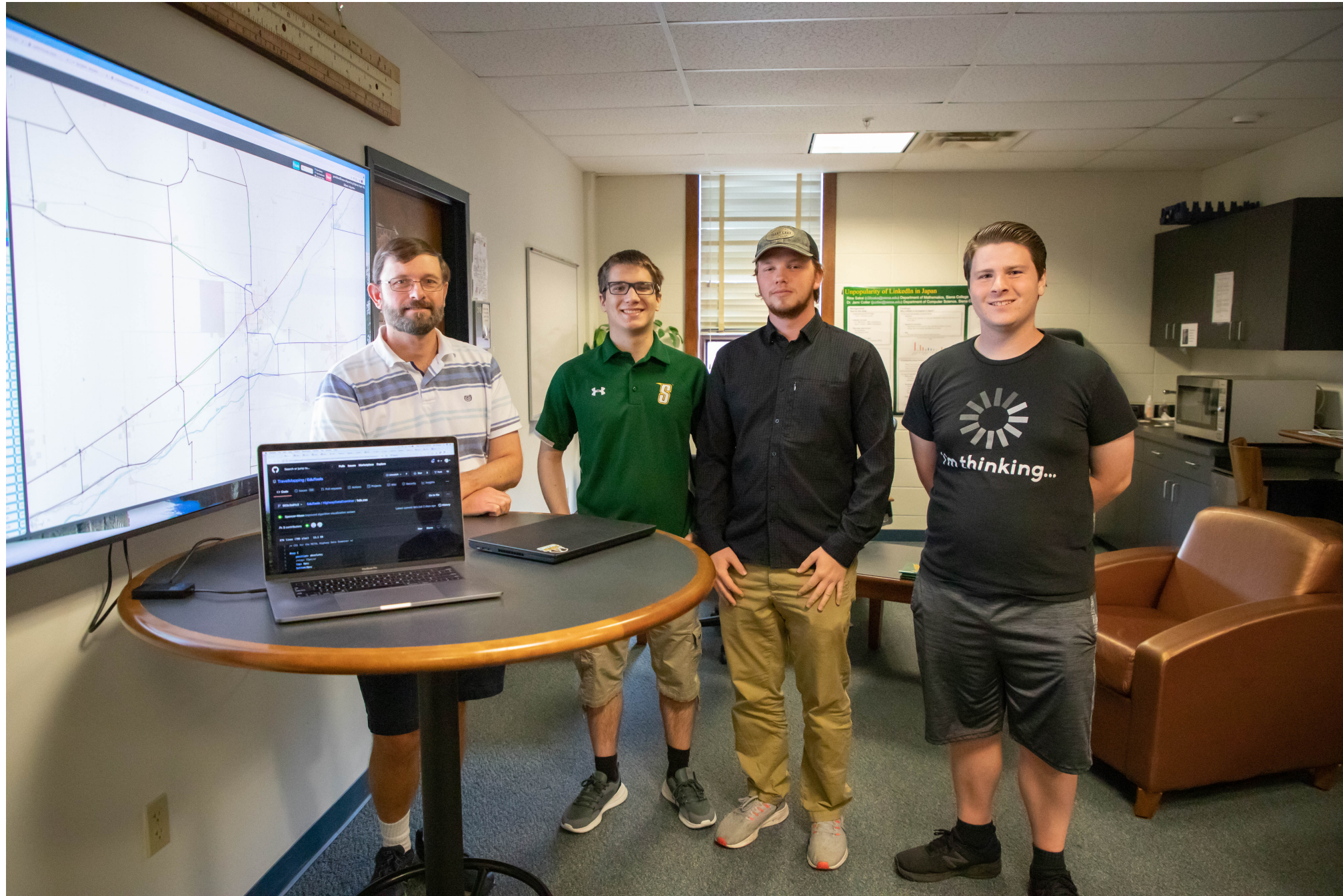
- TM credits at <https://travelmapping.net/credits.php>
- Thank you to the Saint Rose students who have contributed to the project, especially Razie Fathi (now at Smith and still a key project contributor), Dan Priddle, Kevin Bayly, Shipra Goel, Rashmi Reddy Podduturi, and Paul Amodeo.
- Thank you to my Summer 2017, Summer 2018, and Summer 2019, Summer 2021, Summer 2022, and Summer 2023 students: MariaRose Bamundo, Arjol Pengu, Clarice Tarbay, Michael Dagostino, Abdul Samad, Eric Sauer, Zach Goodsell, Tyler Gorman, Alissa Ronca, Bailey Cross, Spencer Moon, Luke Jennings (x2), Mark Verra, and Michael Plekan (x2). Their work was made possible through the support of the Summer Scholars Program, Center for Undergraduate Research and Creative Activity (CURCA), at Siena College.



Acknowledgements



Acknowledgements



Acknowledgements



Use It!

- METAL has been used various courses
- See <https://courses.teresco.org/metal/using.html> for examples including
 - an introductory assignment early in data structures
 - an introductory assignment early in algorithms
 - usage in a “Best Of” data structure
 - convex hull class example
 - assignment to analyze convex hulls
 - Dijkstra’s Road Trip
 - threading intro
- METAL data and HDX are publicly available
- Instructor resources – gradually being developed into learning modules – available on request
- **Please contact me if you want to give it a try!**

New Feature: Graphs with Traveler Information

Spring 2019: Implemented an idea that had been tossed around for a while.

- Idea: take advantage of the traveler data from the TM project
- TM users indicate which highway segments they have traveled
- Those segments correspond directly to the edges in METAL graphs

New Feature: Graphs with Traveler Information

- Augmented the `collapsed` format graphs to a new format called `traveled`
- Header includes the number of unique travelers across all graph edges

```
TMG 2.0 traveled
786 871 148
```

This is from `rmu25-area-traveled.tmg`, indicating 148 unique travelers.

- Each graph edge includes a hexadecimal string with enough bits to represent each traveler

```
653 640 PA978 0000000000004000400100000040000000840
345 254 I-376,US22,US30 11FBC75BD5EDEDDB5E7FF5BEECF587B227B8FD
```

We can see not too many ‘1’ bits on the segment of Route 978, but many more on the segment that carries I-376, US 22, and US 30 through the Pittsburgh area.

- Traveler usernames are included in the last line to allow mapping of bits to users, if desired

Spring 2019 Project

- Part of a final project for *Analysis of Algorithms*

http://courses.teresco.org/cs385_s19/probsets/ac/

- Each individual or group was required to use METAL data in some form for at least one algorithm studied
- Projects included
 - simple variations on searches: most visited/traveled, least visited/traveled
 - variations on Dijkstra's/Prim's/graph traversals that gave weight to heavily- or lightly-traveled segments
 - convex hull of the places visited by a particular traveler
- Presented as a session in Siena's year-end "Academic Showcase" and was heavily attended!

Current Work, Plans, and Ideas for the Future

- Continue to add new data sets and improve the quality of existing sets
- More examples/assignments/labs/teaching modules built around METAL's data and visualizations
- Expanded and improved interactive algorithm visualization
 - including traveler information
 - recursion
 - parallel algorithms
 - more complex algorithms
 - student algorithm “implementations”
- More studies measuring the effectiveness of METAL

Interested?

`http://courses.teresco.org/metal/`



Motivation

"An algorithm must be seen to be believed, and the best way to learn what an algorithm is all about is to try it."
- Donald E. Knuth, *The Art of Computer Programming, Volume 1* (1997), p. 4.

Nearly any topic in a course working with data will be more interesting for students if the data has connections to the real world and if they can visualize the data and results in a meaningful way. When using graph data, such as in a data structures or algorithms class, it needs to be small enough to be manageable, but large enough to be interesting. This might consist of a small road system, airline flight schedules, or even the layout of a campus or building. The *Map-based Educational Tools for Algorithm Learning (METAL)* project provides data and visualization capabilities for this purpose. METAL's graph data sets range in size from a few vertices and edges to several hundred thousand. Its data is derived from the [Travel Mapping \(TM\) Project](#), and represents highway systems from around the world. Its map-based visualization capabilities are built with [Leaflet Maps](#). METAL's interactive code-level algorithm visualization capabilities with debugger-like controls are intended to aid student understanding of graph and other algorithms. Students can implement graph algorithms themselves and display, in map form, the results of computations using those algorithms when applied to METAL data.

Wouldn't it be more fun and interesting to work with this graph

