Algorithms for GIS
csci3225

Laura Toma
Bowdoin College
Spatial data types and models
Spatial data in GIS

- satellite imagery
- planar maps
- surfaces
- networks
- point cloud (LiDAR)
Spatial data in GIS

How to represent it?
**Vector model**

- Data is represented using points, lines and polygons
- Useful for data that has discrete boundaries, such as streets, maps, rivers

**Raster model**

- Data is represented as a surface modeled by a matrix of values (pixels)
- Useful for “continuous” data (ie data that varies continuously) such as satellite imagery, aerial photographs, surface functions such as elevation, pollution, population
Both models can be used for all data

http://gsp.humboldt.edu/olm/Lessons/GIS/08%20Rasters/Images/convertingdatamodels2.png
Both models can be used for all data
World is often modeled as a collection of vector and raster layers

- Streets, parcels, boundaries usually as **vector**
- Elevation, land usage usually as **raster**
Spatial data in GIS

- Satellite imagery
- Planar maps
- Surfaces
- Networks
- Point cloud (LiDAR)
Data structures for networks
Data structures for networks?

• How about this:
  • list of points, each point stores its coordinates
  • list of segments, each segment stores pointers to its vertices

  points
  
  segments

• What if you wanted to traverse a path starting at point 0?
  • search through the segment list looking for a segment that starts from a; you find (0,1)
  • search through the segment list looking for another segment from 1; you find (1,2)
Data structures for networks?

- How about this:
  - list of points, each point stores its coordinates
  - list of segments, each segment stores pointers to its vertices

Spaghetti data structure (like spaghetti, no structure, messy, inefficient eating)
Data structures for networks

- Need a **topological data structure** that allows to traverse paths efficiently
- Wait, a network is a graph!
- Use adjacency list

In practice, this adjacency list needs to be built
  - From raw data
Exercise

Assume you download US road data. It comes as a file that has the following format

- first the number of vertices and the number of edges
- then all the vertices and their geometric coordinates
- then all edges, where an edge is given through the indices of its vertices.

Sketch how you would build an adjacency list from it.

Analyze function of \(|V|\) vertices and \(|E|\) edges.

```
4
3
(1.1, 2.3)
(3.4, 2.1)
(2.6, 1.8)
(1.4, 8.2)
(0,1)
(1,2)
(2,3)
```
Spatial data in GIS

satellite imagery

planar maps

networks

point cloud (LiDAR)

surfaces

RASTER or VECTOR
Data structures for surfaces
Surfaces can have different topologies
Surfaces in GIS

- GIS deals with the surface of the Earth (or Mars, or…)
- The Earth is round, and its surface has the topology of a 2D sphere
Terrains

- A terrain is a function of two variables, $z(x,y)$. Meaning that, for a given $(x,y)$, there is a unique $z(x,y)$

- Put differently, a terrain is a surface in 3D such that any vertical line intersects it in at most one point (xy-monotone)
Terrains
Not terrains

http://paulbourke.net/
Terrains

- Most often terrains represent elevation, but they can also represent other Earth surface functions like rainfall, population, solar radiation, ...

Population surface: $z(x,y) = \text{population\_at}(x,y)$
Global Annual Average PM$_{2.5}$ Grids from MODIS and MISR Aerosol Optical Depth (AOD), 2010: North America

Satellite-Derived Environmental Indicators

Global Annual PM$_{2.5}$ Grids from MODIS and MISR Aerosol Optical Depth (AOD) data sets provide annual 'snap shots' of particulate matter 2.5 micrometers or smaller in diameter from 2001-2010. Exposure to these particles is associated with premature death as well as increased morbidity from respiratory and cardiovascular disease, especially in the elderly, young children, and those already suffering from these illnesses. The grids were derived from Moderate Resolution Imaging Spectroradiometer (MODIS) and Multi-angle Imaging Spectroradiometer (MISR) Aerosol Optical Depth (AOD) data. The raster grid cell size is approximately 50 sq. km at the equator, and the extent is from 70°N to 60°S latitude.


Map Credit: CIESIN, Columbia University, 2010.
Modeling terrains: Digital terrain models

- In practice, terrain data comes as a set of sample points \{(x,y)\} and their sampled z-values
- A digital terrain model = sample points + interpolation method + data structure
With DTM we can do terrain modeling

- watershed maps
- visibility maps
- least-cost paths

Risks of floods, landslides, eruptions, erosion
Costs/benefits of roads, buildings, dikes, hydropower plants, solar power plants
Possible locations of (pre-)historical roads and settlements
Analysis of animal behaviour and evolution
Digital terrain models

Rasters

TINs (vector)

images from Herman Haverkort
Terrain as a raster (grid)

A raster terrain is a matrix of (elevation) values

- Samples
  - uniform grid
- Data structure
  - matrix
- Interpolation method
  - nearest neighbor, linear, bilinear, splines, krigging, IDW, etc
Grids with nearest neighbor interpolation
Grids with nearest neighbor interpolation
Grids with nearest neighbor interpolation
Linear interpolation
Linear interpolation

Terrain: mesh of triangles on grid points
Raster with nearest neighbor vs linear interpolation

http://c1.zdb.io/files/2009/03/10/9/9700e9183d96ccb416b81b053887fef0.gif
Grids in practice

- Grid elevation data can be obtained from aerial imagery
  - image = raster
  - SAR interferometry: by combining Synthetic Aperture Radar (SAR) images of the same area it is possible to go from color to elevation maps
- Massive amounts of aerial imagery available
- Grid elevation data from LiDAR point clouds
Grids in practice

- Elevation data sources
  - GTOPO dataset
    - whole Earth at 1km resolution
    - http: ???
  - SRTM 90m elevation data for entire world
    - can download tiles anywhere in the world
  - SRTM 30m data available for the entire USA (50+GB)
  - Recently, elevation from LiDAR point clouds
    - below 2m resolution
    - Huge!
  - Grids available in a variety of formats
Exercise

Consider an area of 300km-by-300km to be represented as a raster (grid) at:

A. 100m resolution
B. 10m resolution
C. 1m resolution

How big (how many points) is the grid in each case?

Answer:
Terrain as a TIN (triangulated irregular network)

- **Samples**
  - points arbitrarily distributed, variable resolution
- **Interpolation method**
  - linear
- **Data structure**
  - need a topological structure for triangular meshes
Why TINs?

- Uniform resolution = waste on flat areas
- Variable resolution ==> fewer points
The differences bewtween a DEM and a TIN data set

http://www.geophysik.uni-kiel.de/~sabine/BsAs2000/DEM-TIN.gif
The 2D projection of a triangulated terrain is a triangulation.

A TIN is equivalent to a planar triangulation, except points have heights.
Topological data structures for TINs

What do we expect to do on a TIN?

- walk along an edge/triangle path
- given an edge, find the two faces that are adjacent to this edge
- walk along the boundary of a face (triangle)
- find all edges and all triangles incident to a point

A good data structure for TINs should do all these fast

The 2D projection of a triangulated terrain is a triangulation.
### Topological data structures for TINs

<table>
<thead>
<tr>
<th>Edge-based</th>
<th>Triangle-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>• arrays of vertices, edges and triangles</td>
<td>• arrays of vertices and triangles (edges are not stored explicitly)</td>
</tr>
<tr>
<td>• every vertex stores:</td>
<td>• every vertex stores:</td>
</tr>
<tr>
<td>• its coordinates</td>
<td>• its coordinates</td>
</tr>
<tr>
<td>• every edge stores:</td>
<td>• every triangles stores:</td>
</tr>
<tr>
<td>• 2 references to its adjacent vertices</td>
<td>• 3 references to its incident vertices</td>
</tr>
<tr>
<td>• 2 references to its adjacent triangles</td>
<td>• 3 references to its adjacent triangles</td>
</tr>
<tr>
<td>• every triangle stores:</td>
<td>• Note: CGAL uses triangle-based</td>
</tr>
<tr>
<td>• 3 references to its 3 edge</td>
<td></td>
</tr>
</tbody>
</table>
Data structures for TINs

- **edge-based**
- **triangle-based**
Data structures for TINs

edge-based

triangle-based

Analysis? Is one better than other?

- Storing topology: equivalent
- Memory: ??
Data structures for TINs

How much memory does a topological structure for a TIN need?

A. edge-based
B. triangle-based

Denote

\( n \) = number of points in the TIN
\( e \) = number of edges
\( f \) = number of triangles (faces)

There is a formula that connects \( e, f \) and \( n \).
Detour through planar graphs and Euler characteristic
Let $P$ be a set of points in the plane.

A triangulation is a partition of the plane into regions such that all regions are triangles.
A triangulation is a graph: $V =$ the points, $E =$ the edges
Planar graphs

A graph is called planar if it can be drawn in the plane such that no two edges intersect except at their endpoints.

Such a drawing is called a planar embedding of the graph.

Example: \( V = \{a,b,c,d\}, E = \{(a,b), (a,c), (a,d), (b,c), (c,d), (d,a)\} \)

Drawing the graph in the plane is called embedding.
Let’s come up with different embeddings of the graph.
Planar graphs

A graph is called \textbf{planar} if it can be drawn in the plane such that no two edges intersect except at their endpoints.

Such a drawing is called a \textbf{planar embedding} of the graph.

Two drawings of the same graph.

Since there exists a planar embedding, the graph is planar.
Planar graphs

A graph is called **planar** if it can be drawn in the plane such that no two edges intersect except at their endpoints.

Such a drawing is called a **planar embedding** of the graph.

Note: Edges can be represented as simple curves in the drawing

http://people.hofstra.edu/geotrans/eng/methods/img/planarnonplanar.png
Planar graphs

A planar graph introduces a subdivision of the plane into regions called faces, which are polygons bounded by the graph’s edges.

planar graph

triangulation

All faces (except the “outside” face) are triangles
The following relation exists between the number of edges, vertices and faces in a connected planar graph: \( v - e + f = 2 \).
$V - E + F = 2$.

This equation is known as Euler's polyhedron formula.\[^1\] It corresponds to the Euler characteristic of the sphere (i.e. $\chi = 2$), and applies identically to spherical polyhedra. An illustration of the formula on some polyhedra is given below.

<table>
<thead>
<tr>
<th>Name</th>
<th>Image</th>
<th>Vertices $V$</th>
<th>Edges $E$</th>
<th>Faces $F$</th>
<th>Euler characteristic: $V - E + F$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tetrahedron</td>
<td><img src="image" alt="Tetrahedron" /></td>
<td>4</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Hexahedron or cube</td>
<td><img src="image" alt="Hexahedron or cube" /></td>
<td>8</td>
<td>12</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Octahedron</td>
<td><img src="image" alt="Octahedron" /></td>
<td>6</td>
<td>12</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>Dodecahedron</td>
<td><img src="image" alt="Dodecahedron" /></td>
<td>20</td>
<td>30</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>Icosahedron</td>
<td><img src="image" alt="Icosahedron" /></td>
<td>12</td>
<td>30</td>
<td>20</td>
<td>2</td>
</tr>
</tbody>
</table>
Euler formula

The following relation exists between the number of edges, vertices, and faces in a connected planar graph:

\[ v - e + f = 2. \]

Notes:

- For \( c \) connected components: \( v - e + f - c = 1 \)
- \( v - e + f = 2 \) also true for any **convex polyhedral surface** in 3D
- \( v - e + f \) is called the Euler characteristic, \( X \)
- \( X \) is an invariant that describes the shape of space
  - it is \( X = 2 \) for planar graphs and convex polyhedra
  - can be extended to other topological spaces
The surfaces of nonconvex polyhedra can have various Euler characteristics:

<table>
<thead>
<tr>
<th>Name</th>
<th>Image</th>
<th>Vertices $V$</th>
<th>Edges $E$</th>
<th>Faces $F$</th>
<th>Euler characteristic: $V - E + F$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tetrahemihexahedron</td>
<td><img src="image" alt="Tetrahemihexahedron" /></td>
<td>6</td>
<td>12</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>Octahemiocahedron</td>
<td><img src="image" alt="Octahemiocahedron" /></td>
<td>12</td>
<td>24</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>Cubohemiocahedron</td>
<td><img src="image" alt="Cubohemiocahedron" /></td>
<td>12</td>
<td>24</td>
<td>10</td>
<td>-2</td>
</tr>
<tr>
<td>Great icosahedron</td>
<td><img src="image" alt="Great icosahedron" /></td>
<td>12</td>
<td>30</td>
<td>20</td>
<td>2</td>
</tr>
</tbody>
</table>
From Euler formula to size of triangulations

- A triangulation is a planar graph, so \( n-e+f = 2 \) [Euler]
- Furthermore, each triangle has 3 edges and each edge is in precisely 2 triangles (assuming the outside face is a triangle). This means \( 3f = 2e \).
- We get:
  - the number of faces in a triangulation with \( n \) vertices is \( f = 2n-4 \)
  - the number of edges in a triangulation with \( n \) vertices is \( e = 3n-6 \)
- If the outside face is not triangulated it can be shown that
  - \( e < 3n-6, \ f < 2n-4 \)
  - Intuition: Given \( n \) points, the planar graph with largest number of edges and faces is a complete triangulation.

Theorem:
A triangulation with \( n \) vertices has at most \( 3n-6 \) edges and at most \( 2n-4 \) faces.
Known results

- Any planar graph has a planar straight-line drawing where edges do not intersect [Fary’s theorem].

- A graph is planar iff it has no subgraphs isomorphic with K5 or K3,3 [Kuratowski's theorem].

- Any planar graph has at least one vertex of degree $\leq 5$.

- Computationally: There are a number of efficient algorithms for planarity testing that run in $o(n^3)$, but are difficult to implement.
End Detour
The problem: How much memory do we need to store a TIN into a topological structure?

- edge-based
- triangle-based

Denote

- \( n \) = number of points in the TIN
- \( e \) = number of edges
- \( f \) = number of triangles (faces)

Answer:

We’ll use that: a triangulation with \( n \) vertices has \( e \leq 3n-6 \) and \( f \leq 2n-4 \)
<table>
<thead>
<tr>
<th>Grid</th>
<th>TIN</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pros:</strong></td>
<td><strong>Pros:</strong></td>
</tr>
<tr>
<td>implicit topology</td>
<td>variable resolution</td>
</tr>
<tr>
<td>implicit geometry</td>
<td>space efficient (potentially)</td>
</tr>
<tr>
<td>simple algorithms</td>
<td></td>
</tr>
<tr>
<td>readily available in this form</td>
<td></td>
</tr>
<tr>
<td><strong>Cons:</strong></td>
<td><strong>Cons:</strong></td>
</tr>
<tr>
<td>uniform resolution ==&gt; space waste</td>
<td>need to built the TIN (grid → TIN)</td>
</tr>
<tr>
<td>space becomes prohibitive as resolution increases</td>
<td>stored topology takes space</td>
</tr>
<tr>
<td></td>
<td>more complex programming (pointers..);</td>
</tr>
</tbody>
</table>
We have an elevation grid for an area of 300km-by-300km at 1m resolution. The elevation values are represented as floating point numbers (4B).

A. How much space does the grid use (in GB)?

B. Assume the grid undergoes a process of simplification, so that 90% of the grid points are eliminated, leaving 10% of the points. These points are represented as a TIN with a topological edge-based structure. How much space does the TIN use (in GB)?

Answer:
Grid-to-tin simplification
Summary

- Data models: raster and vector
- Networks
- Terrains
  - rasters
  - TINs
- Topological structures for TINs
- Planarity
- Euler formula $V-E+F=2$
- A triangulation is a planar graph, and $e < 3n$ and $f < 2n$
- Grid or TIN?