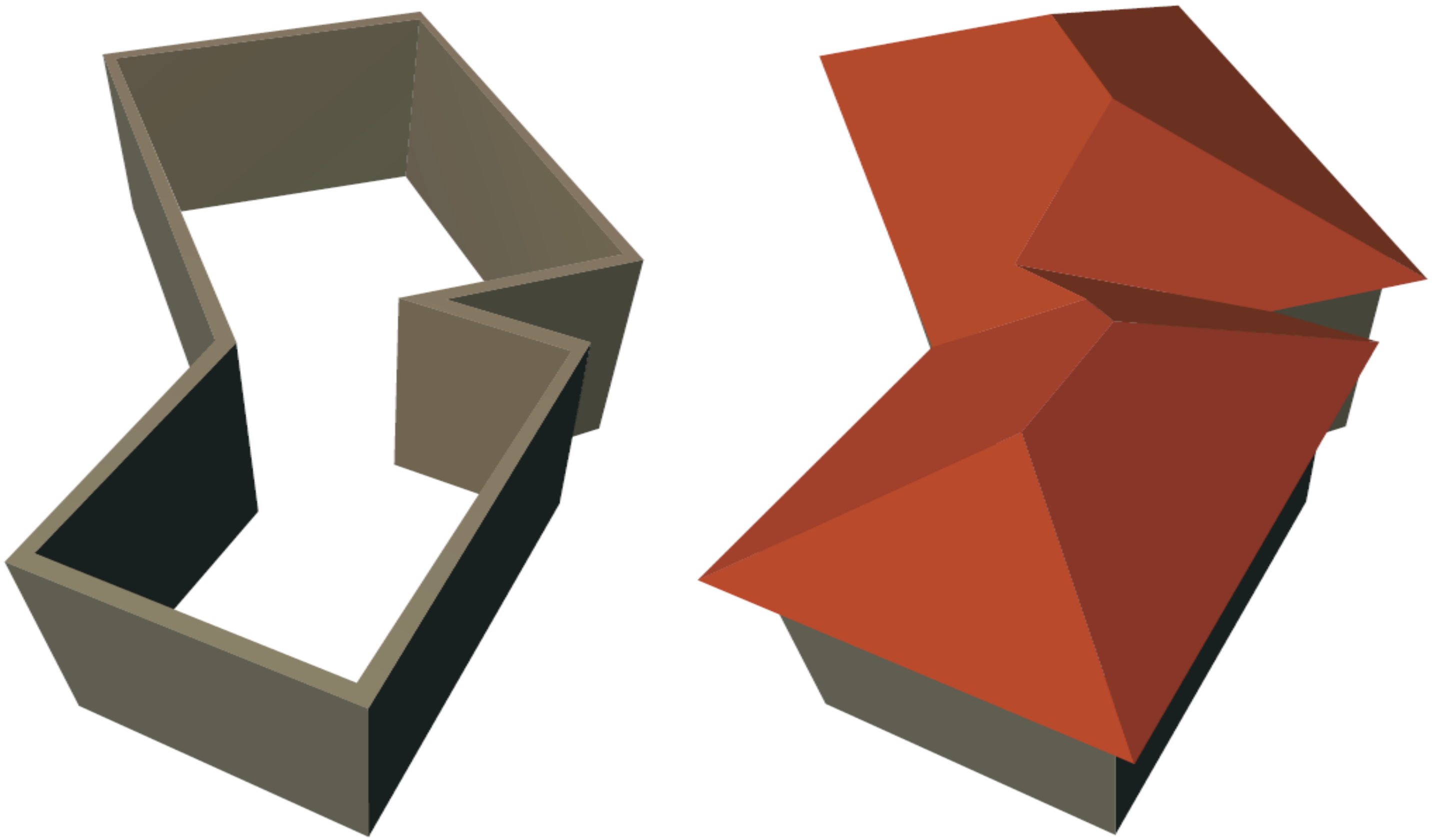
Straight Skeleton

Computational Geometry and Geometric Computing Seminar

Lavinia Dinu

Supervisor: Eric Berberich

How to fit a roof to these walls?



[SSA] Walls and fitted roof

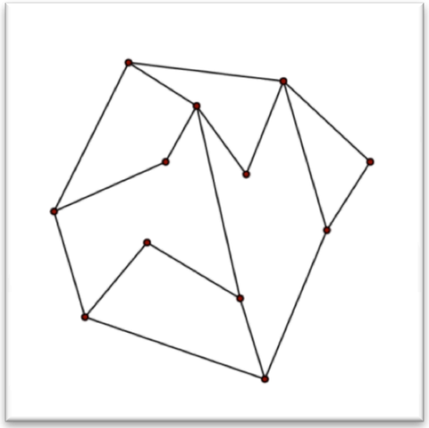
Outline

- Definitions and Basic Properties
- A Simple Algorithm
- Algorithm for Polygons
- Applications
  - Fold and Cut Problem
  - Roof Design

Definitions

Planar Straight Line Graph

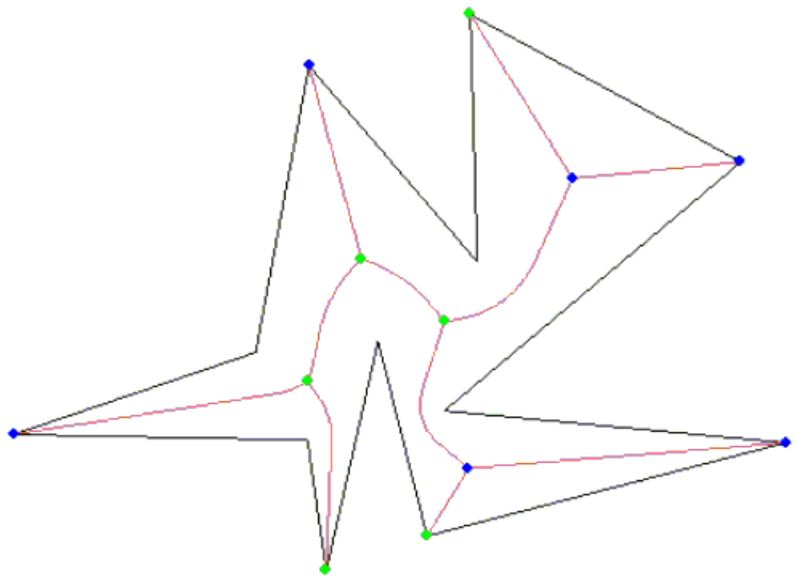
- Embedding of a planar graph in the plane
- Edges mapped to straight line segments
- No self-intersections
- Figures - connected components
- Terminals - vertices of degree one
- Try it yourself: www.planarity.net



[Wiki] A planar straight line graph

Skeleton

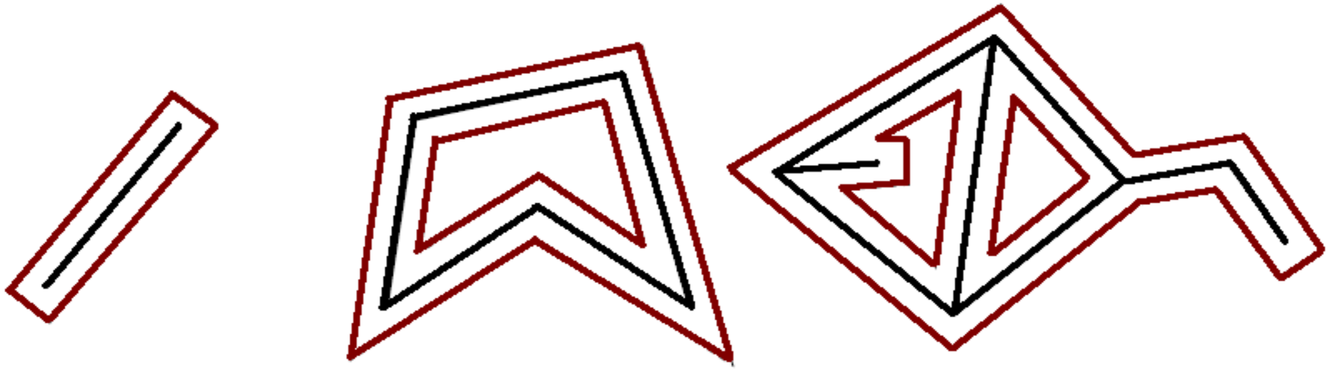
- Partition of plane into regions
- Reflects the geometric shape of the graph
- Arcs and nodes
- Medial axis – non-linear
  - Points that have more than one closest point on the boundary



[Wiki] Medial axis of a polygon

Straight Skeleton: Wavefronts

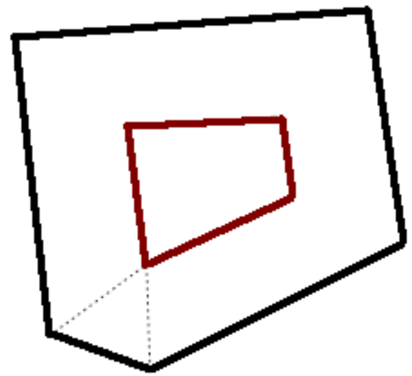
- Generate wavefronts
- Vertices move on angular bisectors
- Propagate wavefronts simultaneously



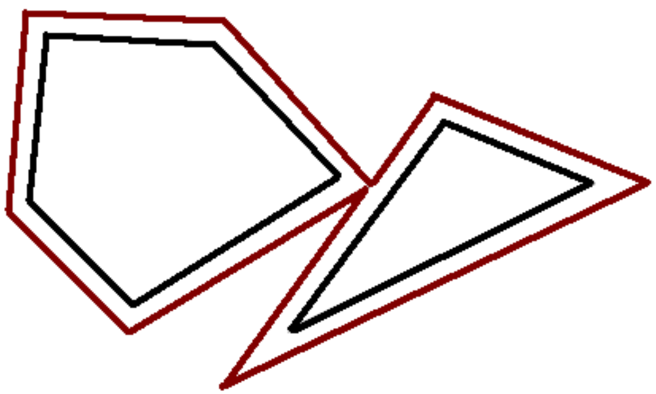
[SSG] Polygonal figures and initial wavefronts in red

Straight Skeleton: Events

- Edge event
  - Edge length decreases to zero



- Split event
  - Wavefront edge splits due to interference

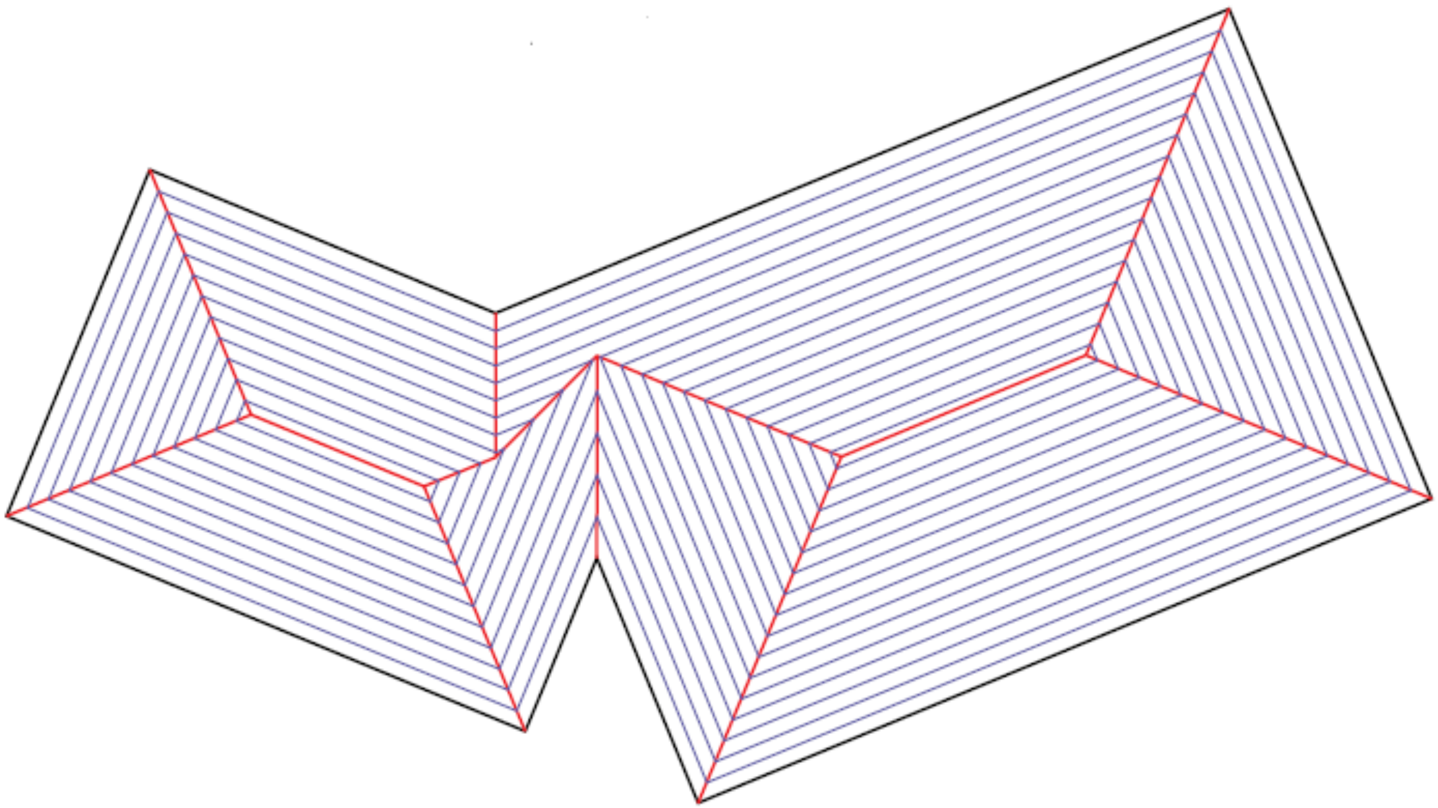


Straight Skeleton: Definition

-

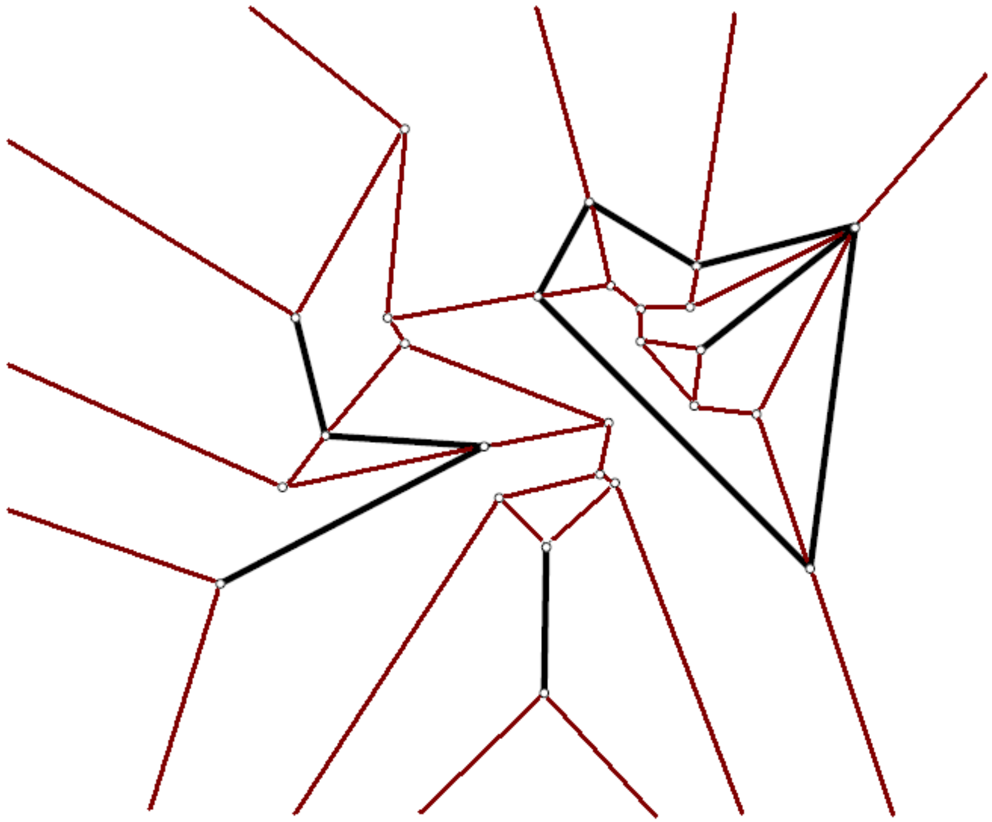
- Straight skeleton – angular bisectors traced by wavefront vertices
- Arcs and nodes
- Events ↔ nodes
- Face – area swept by an edge
- Region – union of faces of a figure

Straight Skeleton: Example



[SAS] Blue wavefronts and red skeleton of a polygon

Straight Skeleton: Example



[SSG] Red straight skeleton of graph

Properties

-
- Is a tree
- $2n + t - 2$ nodes, for n vertices and t terminals
- Restricting to the interior of a polygon with n vertices:
  - **n connected faces**
  - **$n - 2$ nodes**
  - **$2n - 3$ arcs**
- Does not have a Voronoi interpretation – the usual algorithms can not be applied

Basic Algorithm

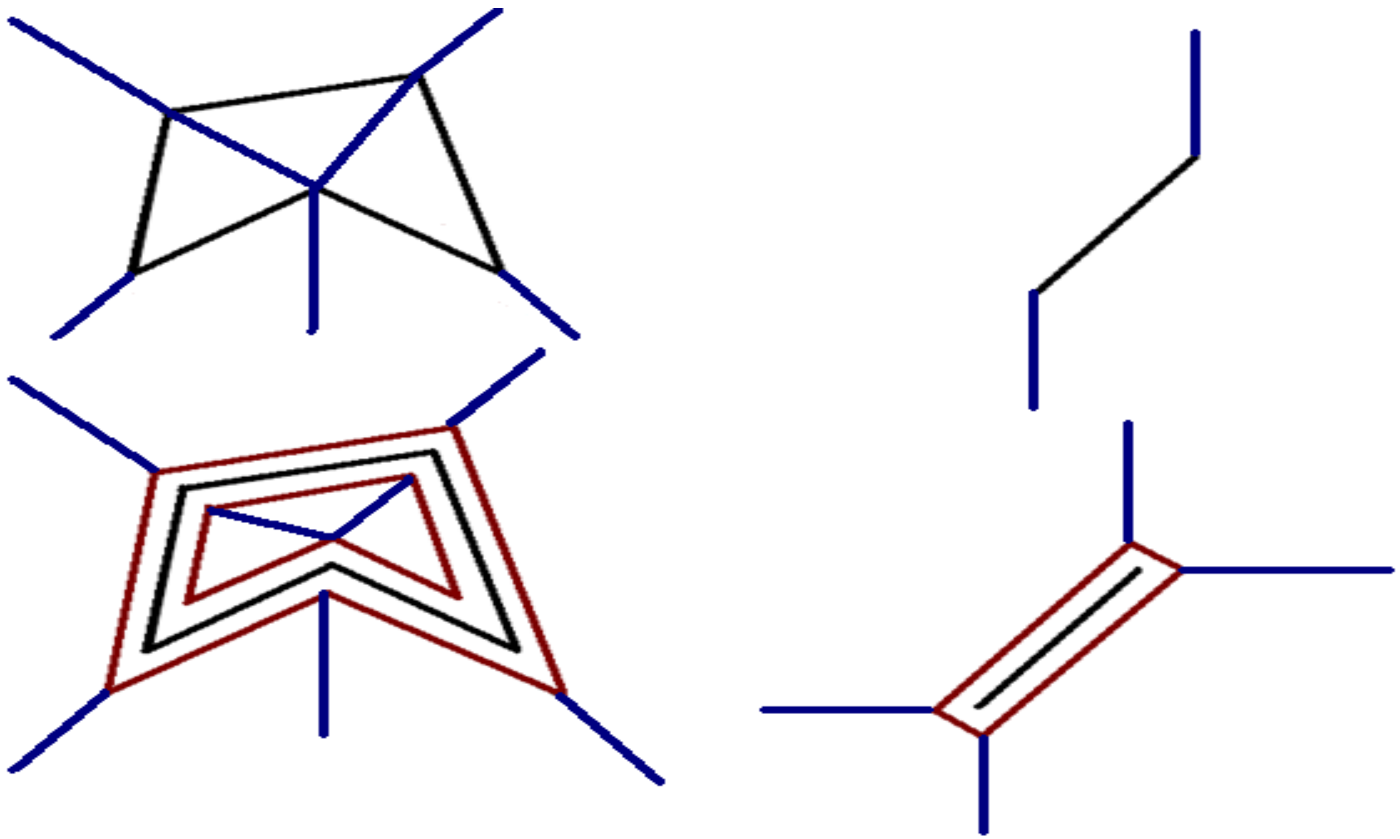
Idea

-
- Keep triangulation of areas not reached by any wavefront
- Vertices of wavefronts → vertices of triangulation
- Edge & split event → collapsing triangle
- Spokes – edges of triangulation
- Use priority queue of triangles
  - Sort by collapsing time

Initial triangulation

- Triangulating the vertices:
  - Initially $2n - 2$ triangles
  - Shared for wavefront vertices
- Triangulating the terminals
  - Duplicate into two vertices linked by edge
  - First keeps initial spokes
  - Second uses a new spoke that partitions the resulting quadrilateral
  - One new triangle is created
- $2n - 2 + t$ triangles

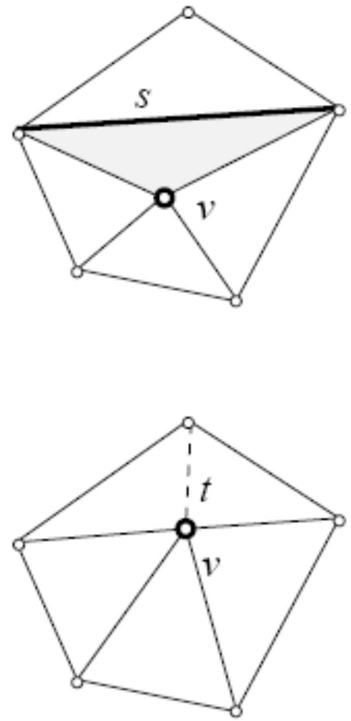
Initial triangulation



Triangulation of the graph (up) and triangulation of the initial wavefronts (down) for two figures. Blue edges represent spokes, red edges wavefronts.

Flip Event

-
- Vertex sweeps across a spoke *s*
- Remove *s*
- Insert a new spoke to preserve triangulation
- Caused only by reflex vertices
- No new skeleton node created
- Number of triangles unchanged
- $O(n^3)$ such events

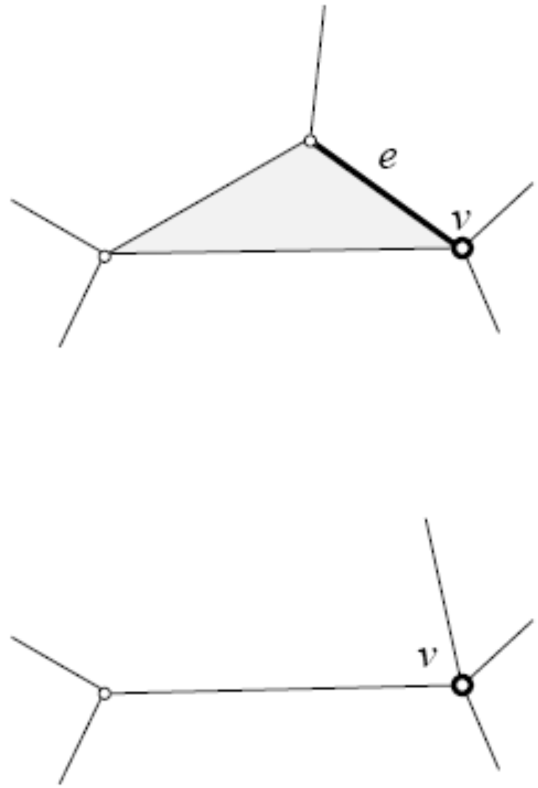


[SAS] Flip Event

Edge Event

-

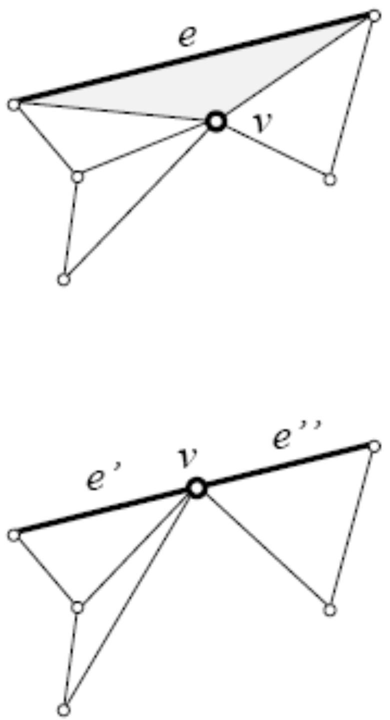
- *v* merges with another vertex
- *e* is removed
- Creates a node of the skeleton
- Removes one triangle
- Reflex vertex merges with convex vertex



[SAS] Edge Event

Split Event

- v hits an edge e
- Create new edges e' and e''
- Remove e
- Duplicate v
- Assign the new edges to v
- Creates a node of the skeleton
- Removes one triangle
- Reflex vertex becomes two convex vertices



[SAS] Split Event

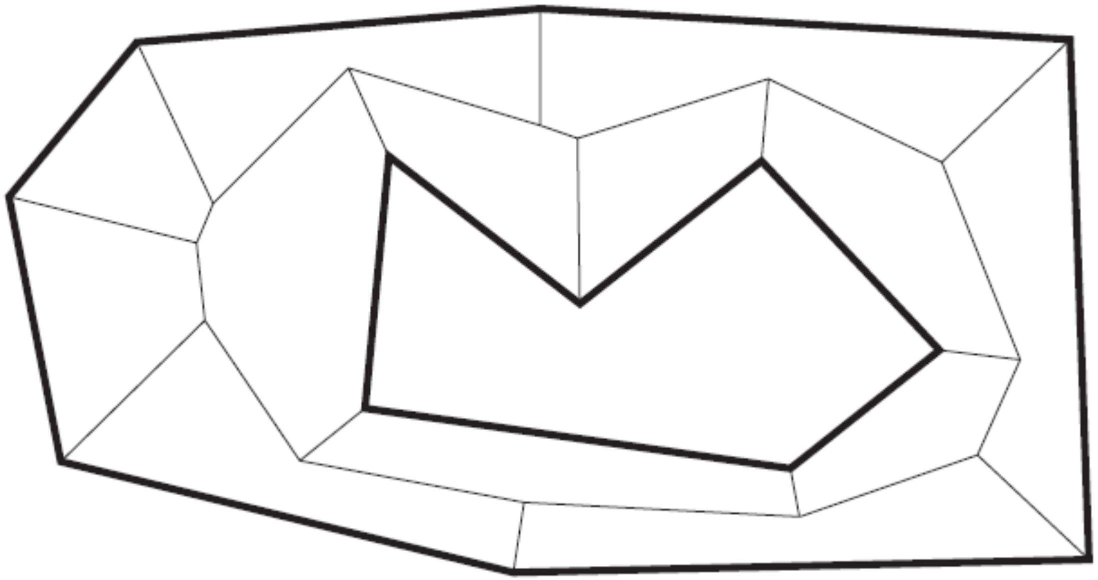
Complexity analysis

-
- Store at most $2n + t - 2$ triangles
- Reflex vertices disappear after a non-flip event
- $O(n^3)$ processed triangles
- Recompute the priority queue after non-flip events
  - At each event $O(n)$ triangles are updated
- $O(n^3 \log n)$ time and $O(n)$ space
- Typical case $O(n \log n)$

A Better Algorithm

Input

- Simple non-degenerate polygon
- Can have holes
- Interior polygon will have the opposite orientation



[SSI] Polygon with hole

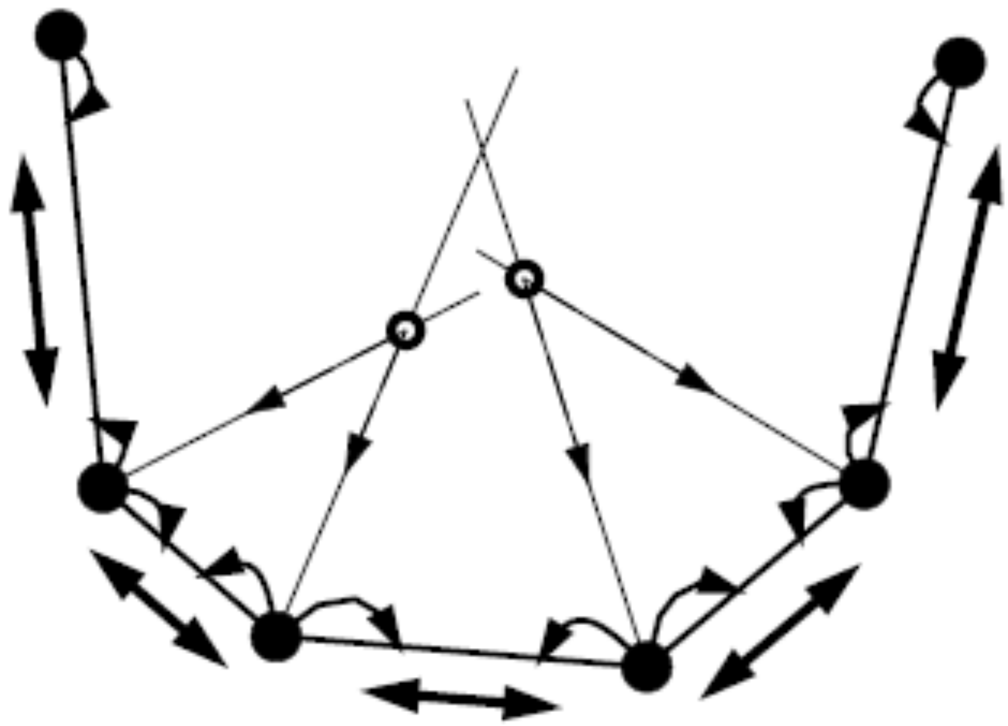
Idea & Data Structures

- Simulate the progression of the wavefronts
- Use set of circular lists of active vertices – SLAV
  - Each LAV – loop of vertices of a wavefront
  - Each vertex
    - References to both neighbors
    - References to adjacent edges in the original polygon
  - Doubly connected circular list of active vertices
- Priority queue
  - events

Convex Polygons: Initialization

- Store all vertices in one LAV
- For each vertex
  - Add reference to the two incident edges
  - Compute the angle bisector
- Compute nearer intersection of bisector with adjacent ones
  - Add to priority queue
  - Sort by distance to the corresponding edge
  - Add reference to the corresponding vertices
  - Can be duplicities among the intersection points

Convex Polygons: Initialization



[SSI] Initialization

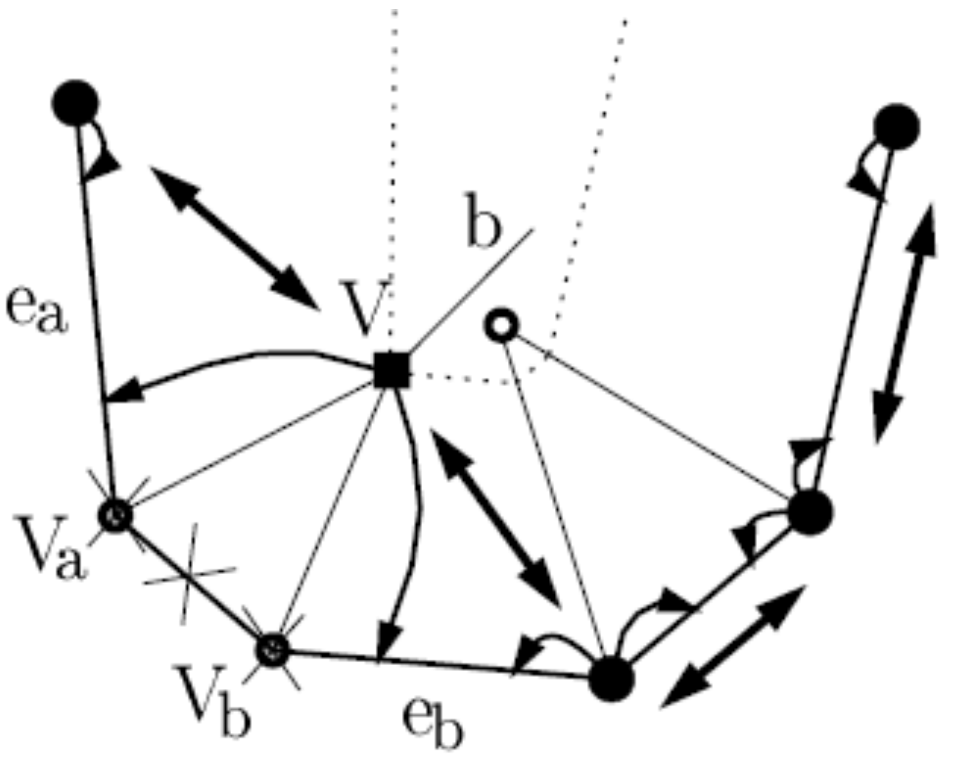
Convex Polygons – Loop Part 1

-
- Pop point I from the queue
  - Points to V_a and V_b
- Remove duplicities
  - If V_a and V_b processed then loop
- Process the edge event
  - Check if wavefront disappears – events happens inside triangle
    - V_c is the predecessor of V_a and the successor of V_b
    - Output arcs $V_a I$, $V_b I$ and $V_c I$

Convex Polygons – Loop Part 2

- Process normal edge event
  - Output arcs $V_a I$ and $V_b I$
  - Mark V_a and V_b as processed
  - Create new node in the place of I
  - Update the data structures – like the initialization
    - Insert into LAV
    - Link to edges
    - Compute bisector and intersections
    - Store nearer intersection
    - Can cause duplicates

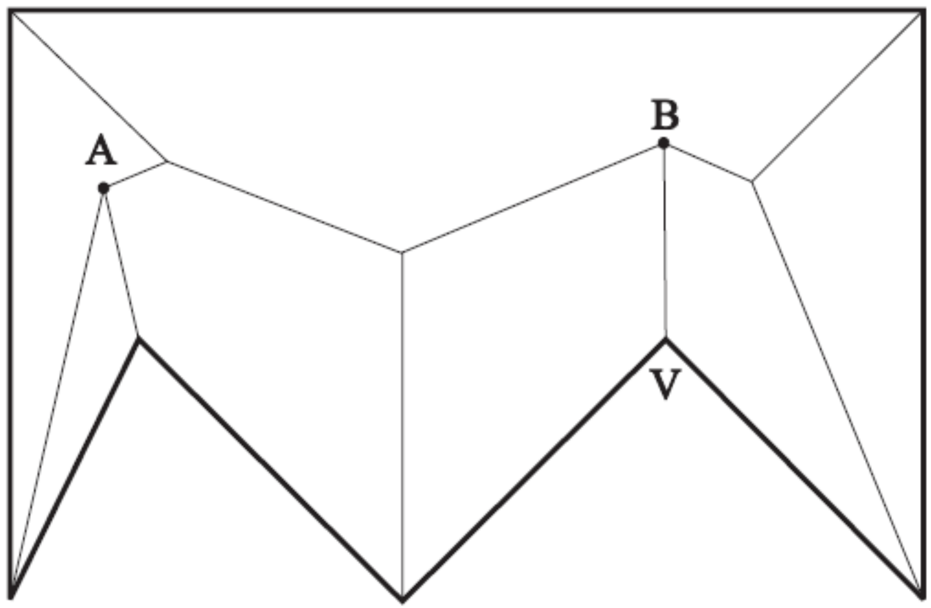
Convex Polygons – Loop Part 3



[SSI] First Step

Non-convex Polygons

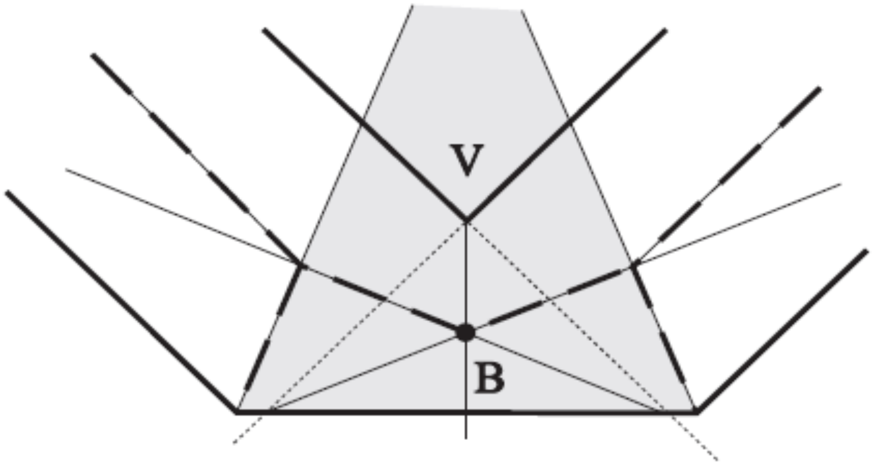
- Add event type attribute
  - Edge event or split event
- Reflex vertices can cause both types of events



[SSI] Reflex vertices causing edge event A and split event B

Determine split events

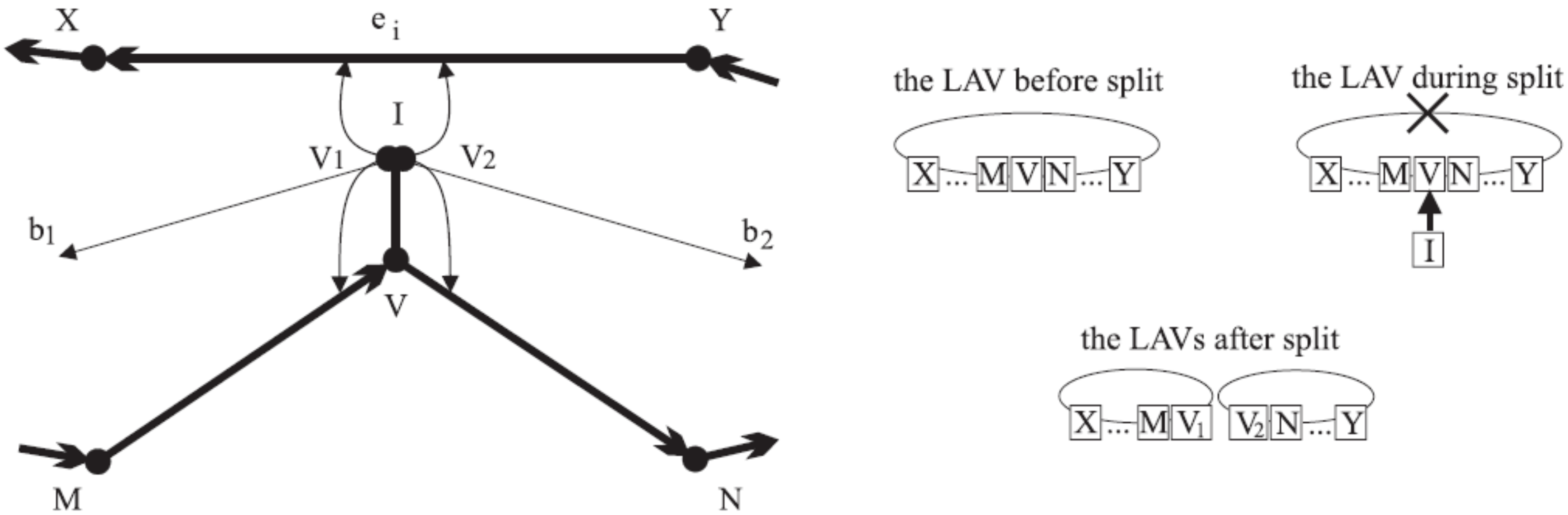
- Point B at a split event
  - Same distance to three straight lines
- For each reflex vertex V and line segment e_i
  - Compute a possible B_i
  - Test if contained in feasible area
  - Choose closest B_i for each V



[SSI] Split event and feasible area

Processing split events

- Output arc VI
- The LAV is split into two
- Duplicate V



[SSI] Updating the LAVs

Complexity Analysis

-
- Edge events
  - $O(n \log n)$ time
  - $O(n)$ storage
- Split events
  - $O(nm)$ time, m reflex vertices
    - For each reflex vertex, look at all other edges to find split event
  - $O(n)$ storage
- Total $O(nm)$ with $O(n)$ storage

Applications

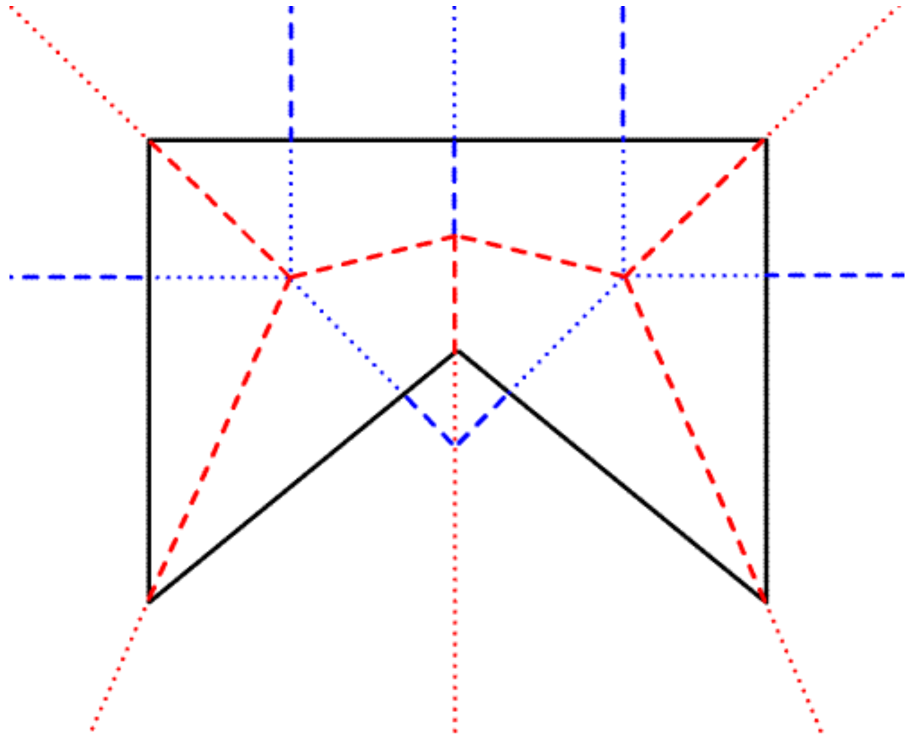
Applications

- Decomposition of concave polygons into convex regions
  - Preprocessing step for shape matching, Minkovski sum
- Interpolating polygonal curves
- Fold and cut problem
- Roof design
- Terrain reconstruction

Fold and Cut Problem

- Fold a piece of paper and cut along a line
- Any possible shape can be obtained
- Creases are:
  - The straight skeleton
  - Perpendicular lines to the given shape reflected at the skeleton
- [PR] Applet to generate the folds

Fold and Cut Problem

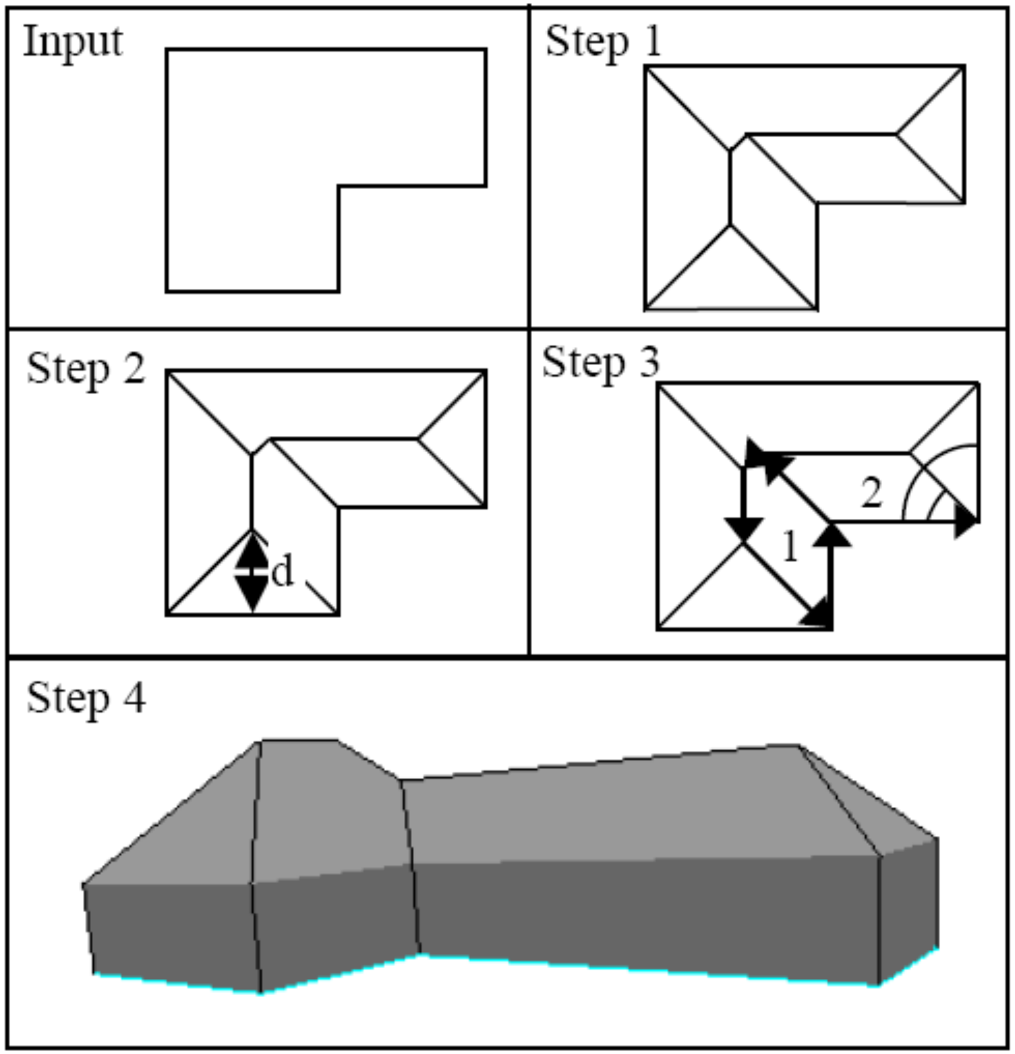


[FC] Fold Pattern for a figure

Roof Design: Hip Roof

- All facets have the same slope
- Algorithm:
  - Construct straight skeleton
  - Determine distance of each vertex from supporting edge
  - Determine roof planes
  - Raise vertices accordingly
- Can be adapted for different slopes
  - Change propagation speed of wavefronts

Roof Design: Hip Roof

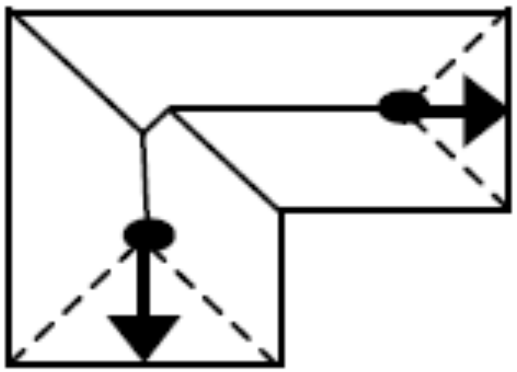


[AGF] Steps for creating a hip roof

Roof Design: Gable Roof

- Has an small gable on top
- Algorithm
  - Find a vertex created by bisectors from the original polygon
  - Move it to the midpoint of the supporting edge

[Wiki] Gable roof



[AGF] Creating a gable roof

Summary

-

- New type of skeleton
- Linear structure
- Algorithm in $O(n^2)$
- Existing subquadratic algorithm
  - Tailor-made data structures
- Applications in various areas

Sources

- [SSG] O. Aichholzer, F. Aurenhammer. Straight Skeleton for General Polygonal Figures in the Plane
- [ANT] O. Aichholzer, F. Aurenhammer, D. Alberts, B.Gaertner. A Novel Type of Skeleton for Polygons
- [SSI] P. Felkel, S. Obdrzalek. Straight Skeleton Implementation
- [AGF] R. G. Laycock, A. M. Day. Automatically Generating Roof Models from Building Footprints
- [SAS] D. Eppstein. A Subquadratic Algorithm for the Straight Skeleton
- [FC] http://erikdemaine.org/foldcut/
- [PR] http://cgm.cs.mcgill.ca/~athens/cs507/Projects/2003/EricBiunno/
- [Wiki]Wikipedia

Thank you for your attention!

Questions?