Cartograms: Drawing Weighted Graphs as Maps
Gastvortrag Vorlesung Algorithmen II

Dr. Martin Nöllenburg

Institut für Theoretische Informatik
YIG Algorithmen zur Geovisualisierung

18.12.2012

Spatial Statistical Data

How to best visualize statistics about spatial data?

Example: population in the USA

Spatial Statistical Data

Karlsruhe Institute of Technology

How to best visualize statistics about spatial data?

Example: population in the USA

State	2011 pop.
Alabama	4,802,740
Alaska	722,718
Arizona	6,482,505
Arkansas	2,937,979
California	37,691,912
Colorado	5,116,796
Connecticut	3,580,709
Delaware	907,135
DC	617,996
Florida	19,057,542
Georgia	9,815,210
Hawaii	1,374,810
Idaho	1,584,985
Illinois	12,869,257
Indiana	6,516,922
Iowa	3,062,309
Kansas	2,871,238
Kentucky	4,369,356
Louisiana	4,574,836
Maine	1,328,188
Maryland	5,828,289
Massachusetts	6,587,536
Michigan	9,876,187
Minnesota	5,344,861
Mississippi	2,978,512
Missouri	6,010,688
Montana	998,199
Nebraska	1,842,641
Nevada	2,723,322
New Hampshire	1,318,194
New Jersey	8,821,155
New Mexico	2,082,224
New York	19,465,197
North Carolina	9,656,401
North Dakota	683,932
Ohio	11,544,951
Oklahoma	3,791,508
Oregon	3,871,859
Pennsylvania	12,742,886
Rhode Island	1,051,302
South Carolina	4,679,230
South Dakota	824,082
Tennessee	6,403,353
Texas	25,674,681
Utah	2,817,222
Vermont	626,431
Virginia	8,096,604
Washington	6,830,038
West Virginia	1,855,364
Wisconsin	5,711,767
Wyoming	568,158

a table?

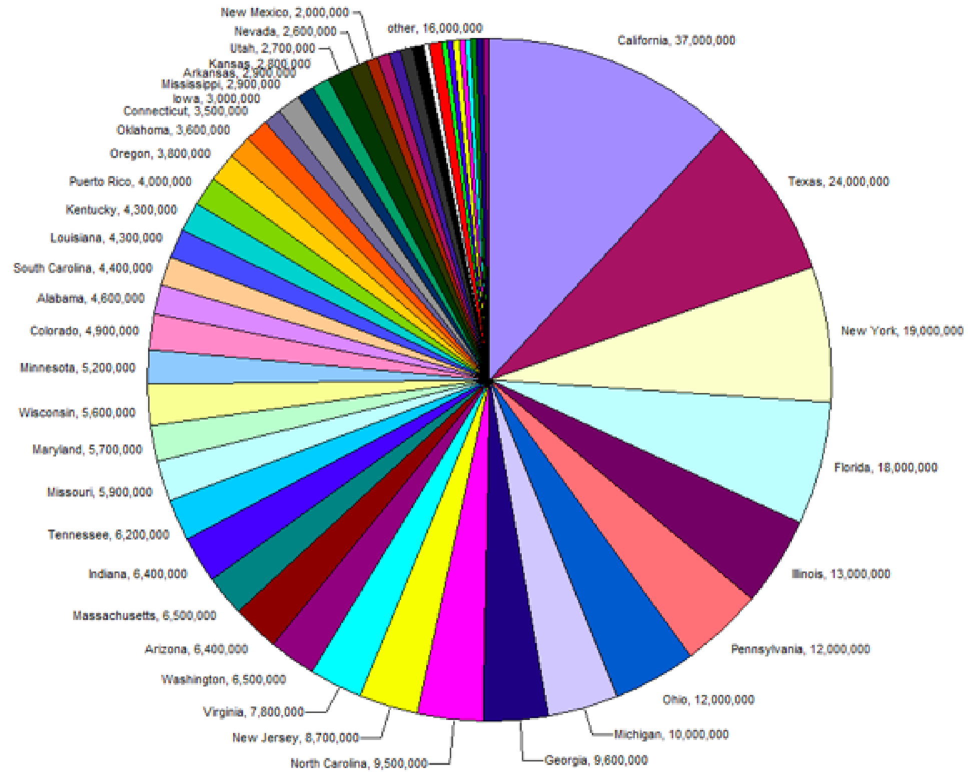
Spatial Statistical Data

How to best visualize statistics about spatial data?

Example: population in the USA

State	2011 pop.
Alabama	4,802,740
Alaska	722,718
Arizona	6,482,505
Arkansas	2,937,979
California	37,691,912
Colorado	5,116,796
Connecticut	3,580,709
Delaware	907,135
DC	617,996
Florida	19,057,542
Georgia	9,815,210
Hawaii	1,374,810
Idaho	1,584,985
Illinois	12,869,257
Indiana	6,516,922
Iowa	3,062,309
Kansas	2,871,238
Kentucky	4,369,356
Louisiana	4,574,836
Maine	1,328,188
Maryland	5,828,289
Massachusetts	6,587,536
Michigan	9,876,187
Minnesota	5,344,861
Mississippi	2,978,512
Missouri	6,010,688
Montana	998,199
Nebraska	1,842,641
Nevada	2,723,322
New Hampshire	1,318,194
New Jersey	8,821,155
New Mexico	2,082,224
New York	19,465,197
North Carolina	9,656,401
North Dakota	683,932
Ohio	11,544,951
Oklahoma	3,791,508
Oregon	3,871,859
Pennsylvania	12,742,886
Rhode Island	1,051,302
South Carolina	4,679,230
South Dakota	824,082
Tennessee	6,403,353
Texas	25,674,681
Utah	2,817,222
Vermont	626,431
Virginia	8,096,604
Washington	6,830,038
West Virginia	1,855,364
Wisconsin	5,711,767
Wyoming	568,158

a table?



a pie chart?

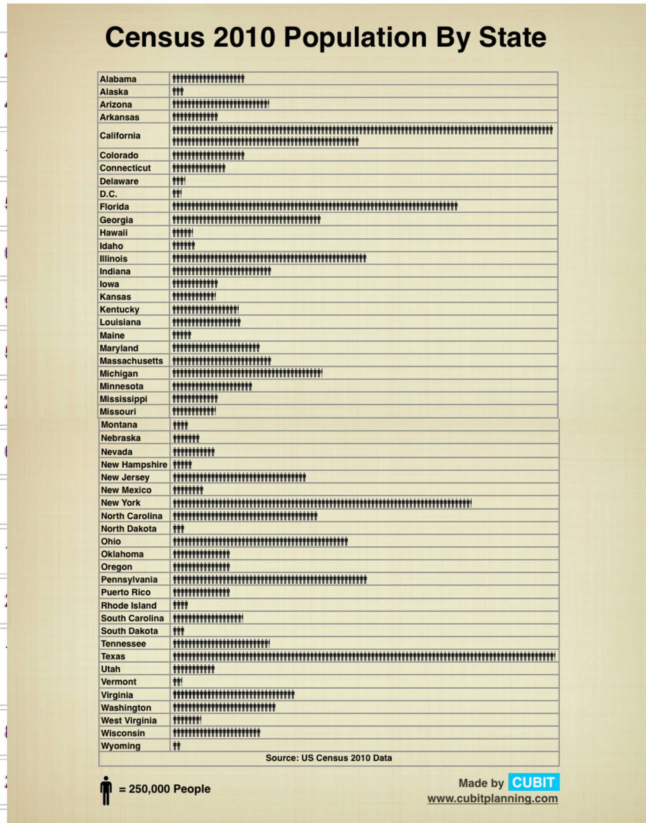
Spatial Statistical Data

How to best visualize statistics about spatial data?

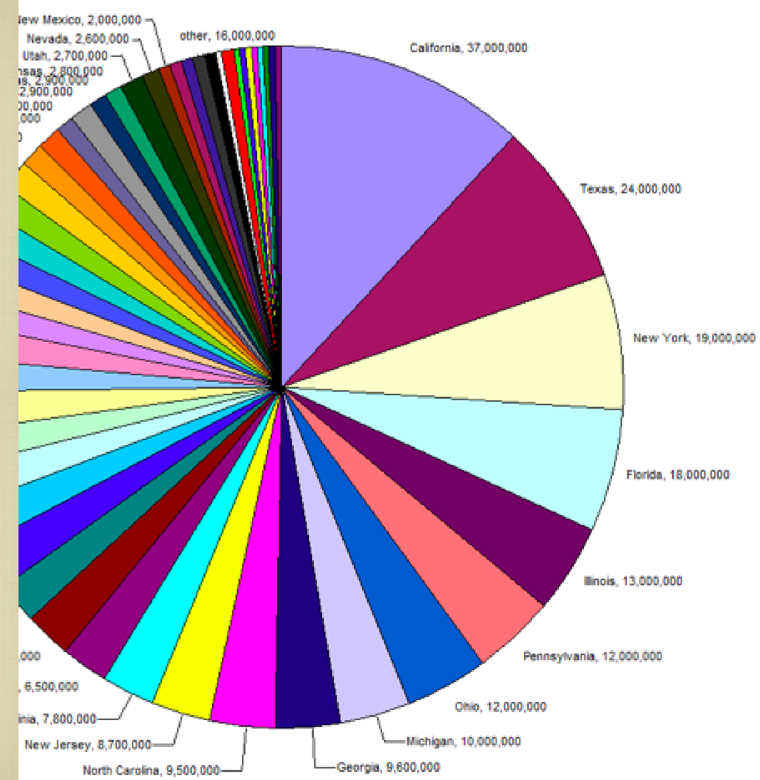
Example: population in the USA

State	2011 pop.
Alabama	4,802,740
Alaska	722,718
Arizona	6,482,505
Arkansas	2,937,979
California	37,691,912
Colorado	5,116,796
Connecticut	3,580,709
Delaware	907,135
DC	617,996
Florida	19,057,542
Georgia	9,815,210
Hawaii	1,374,810
Idaho	1,584,985
Illinois	12,869,257
Indiana	6,516,922
Iowa	3,062,309
Kansas	2,871,238

Kentucky	
Louisiana	
Maine	
Maryland	
Massachusetts	
Michigan	
Minnesota	
Mississippi	
Missouri	
Montana	
Nebraska	
Nevada	
New Hampshire	
New Jersey	
New Mexico	
New York	
North Carolina	9,656,401
North Dakota	683,932
Ohio	11,544,951



a bar chart?



a pie chart?

a table?

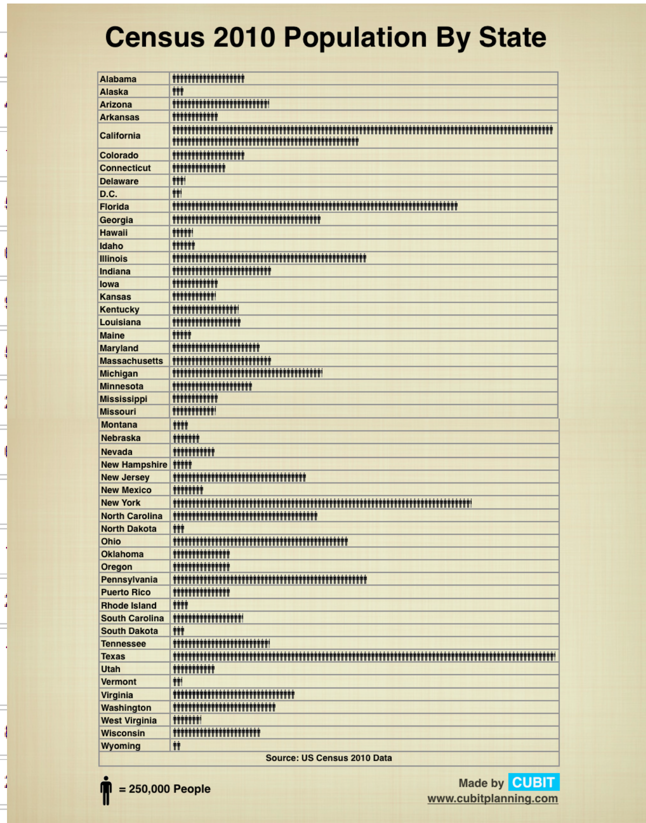
Spatial Statistical Data

How to best visualize statistics about spatial data?

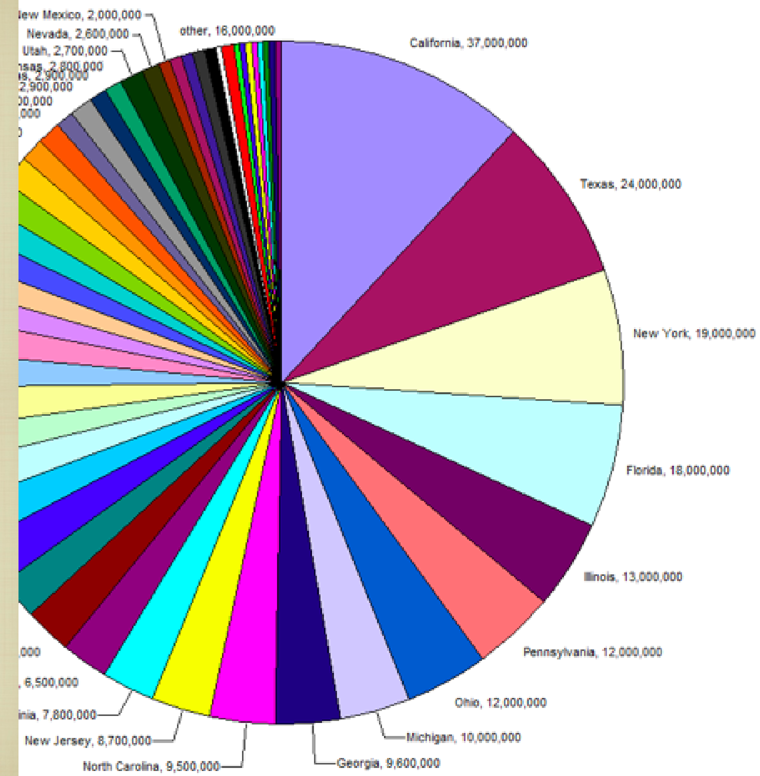
Example: population in the USA

State	2011 pop.
Alabama	4,802,740
Alaska	722,718
Arizona	6,482,505
Arkansas	2,937,979
California	37,691,912
Colorado	5,116,796
Connecticut	3,580,709
Delaware	907,135
DC	617,996
Florida	19,057,542
Georgia	9,815,210
Hawaii	1,374,810
Idaho	1,584,985
Illinois	12,869,257
Indiana	6,516,922
Iowa	3,062,309
Kansas	2,871,238
Kentucky	[illegible]
Louisiana	[illegible]
Maine	[illegible]
Maryland	[illegible]
Massachusetts	[illegible]
Michigan	[illegible]
Minnesota	[illegible]
Mississippi	[illegible]
Missouri	[illegible]
Montana	[illegible]
Nebraska	[illegible]
Nevada	[illegible]
New Hampshire	[illegible]
New Jersey	[illegible]
New Mexico	[illegible]
New York	19,…
North Carolina	9,656,401
North Dakota	683,932
Ohio	11,544,951

a table?



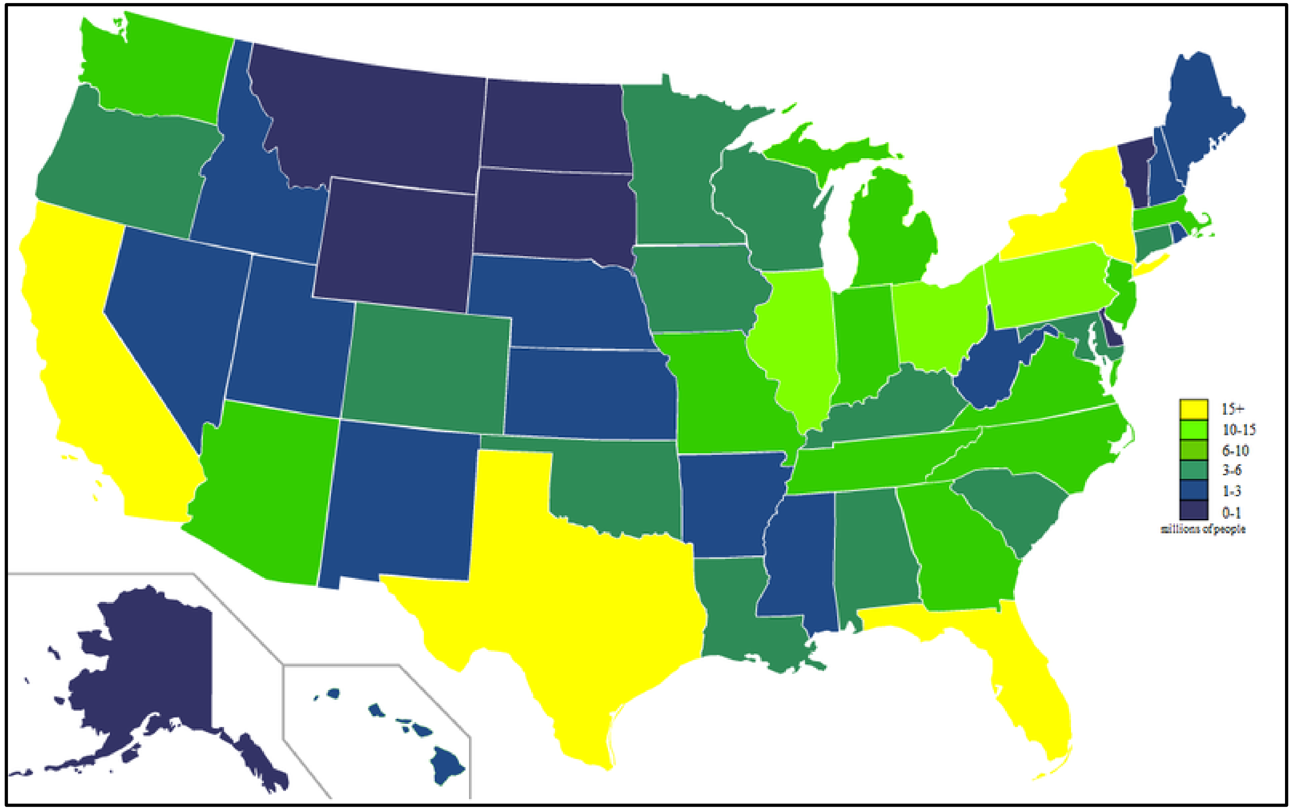
a bar chart?



a pie chart?

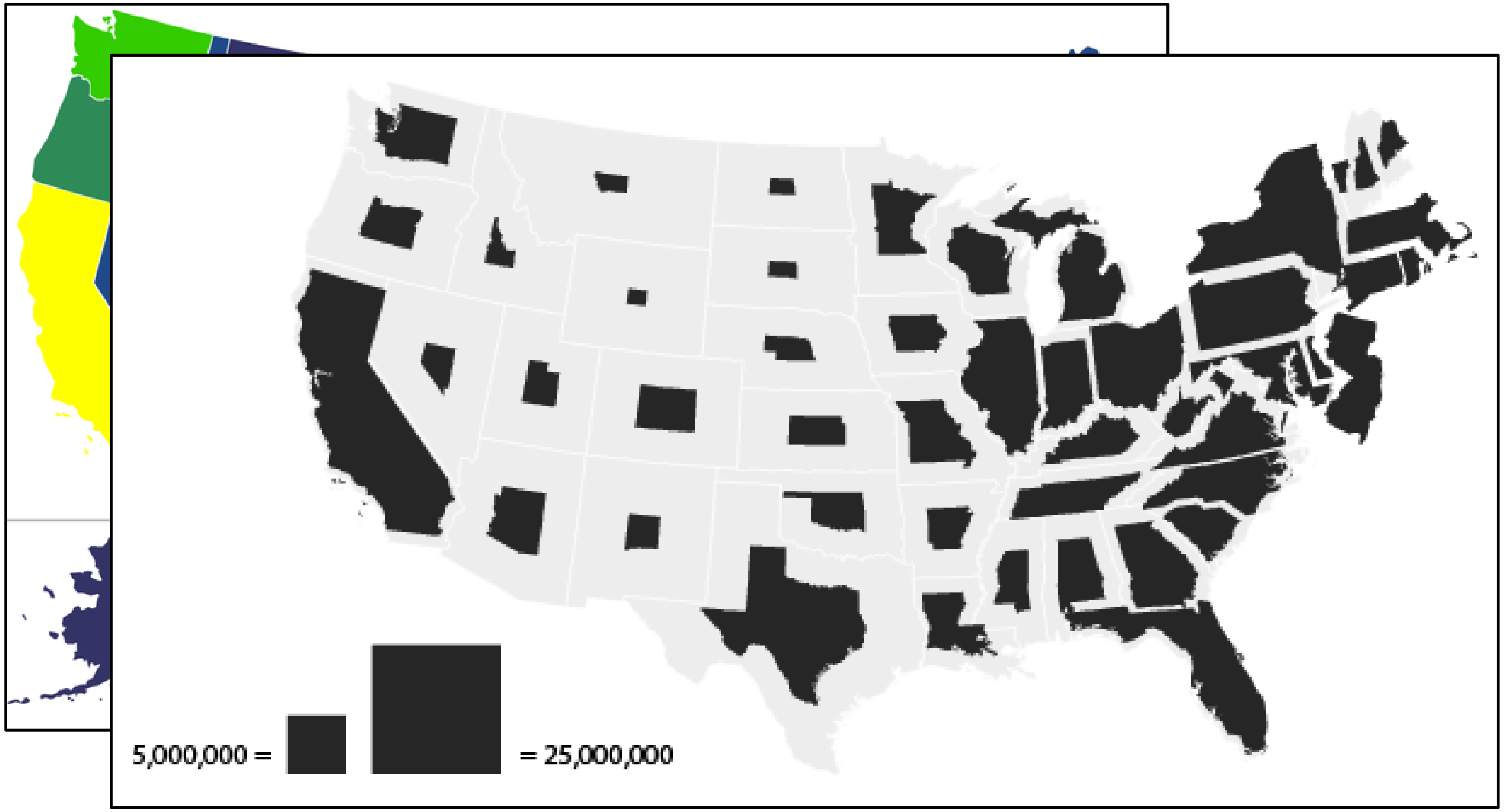
Problem: standard methods don't show the spatial patterns!

Map-based Statistical Visualization



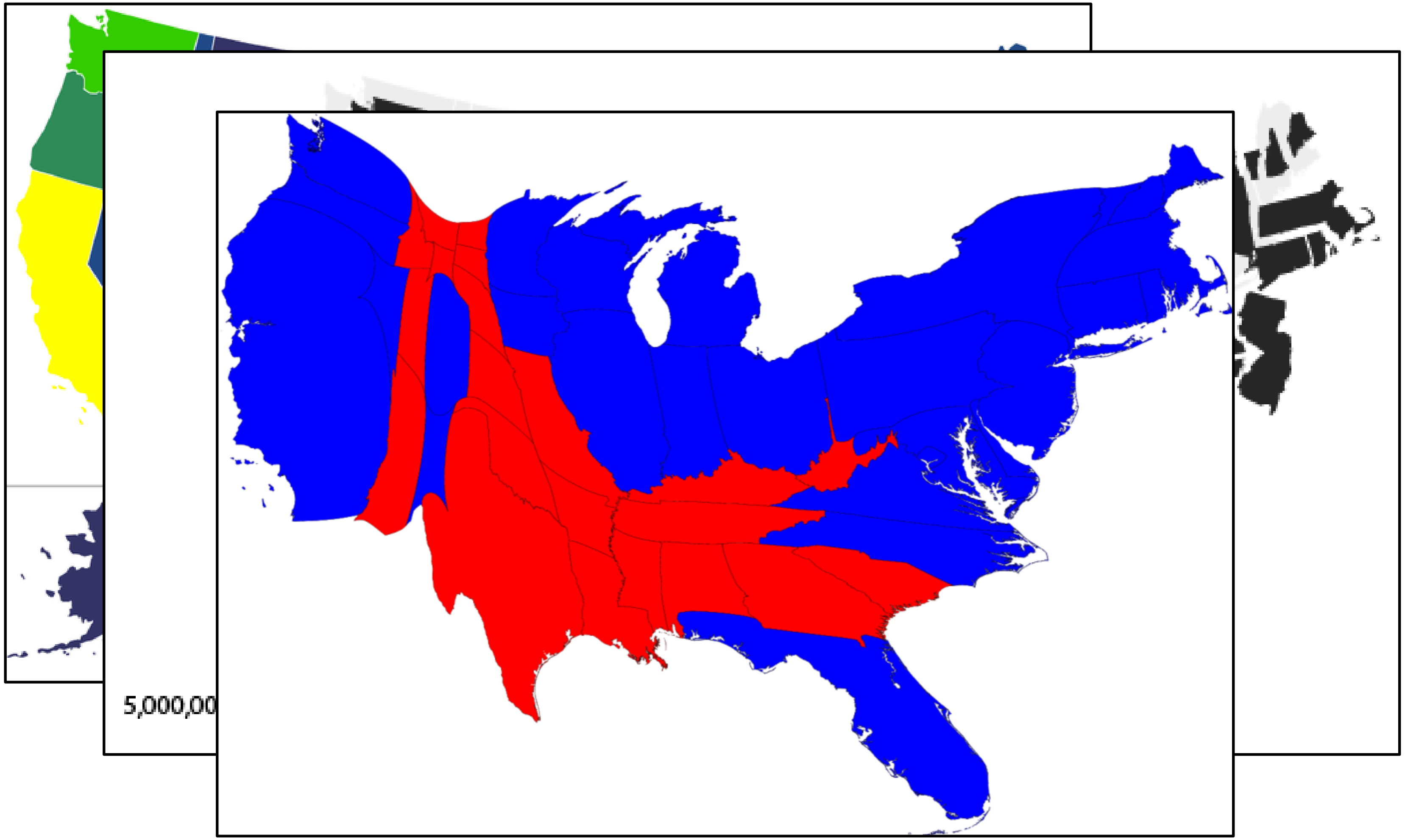
choropleth map: use colors to show stats

Map-based Statistical Visualization



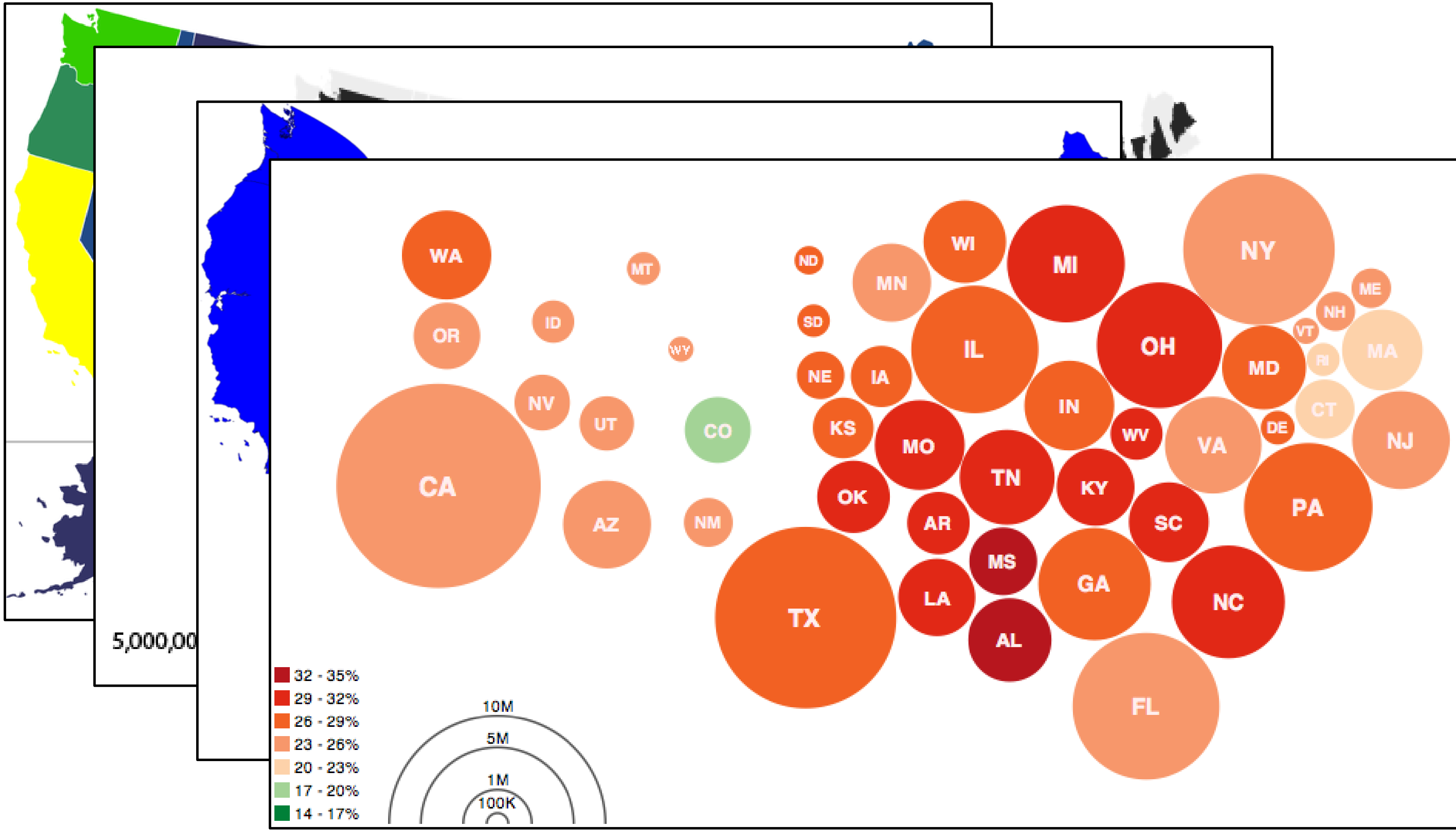
non-contiguous area cartogram:
area proportional to population

Map-based Statistical Visualization



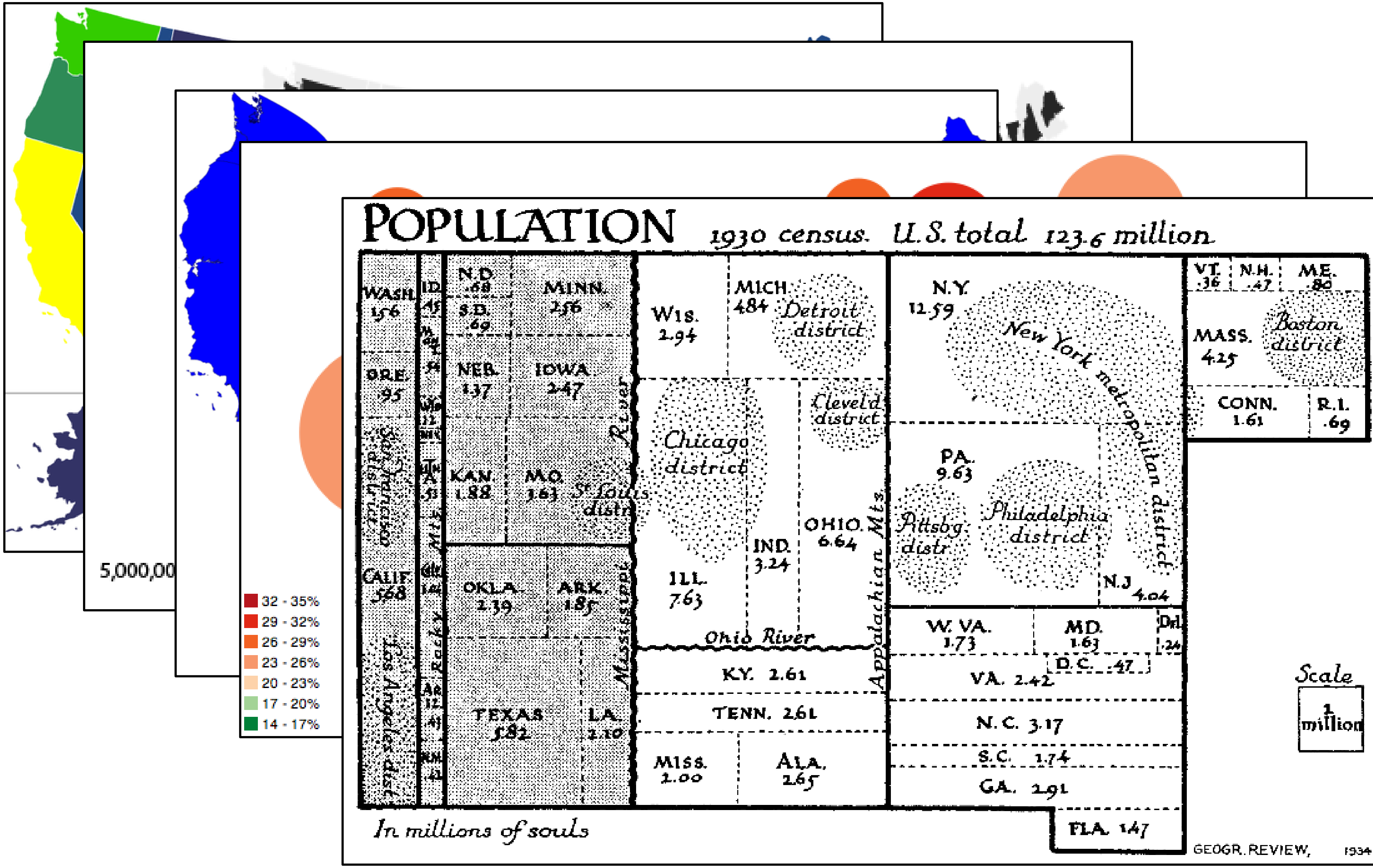
contiguous area cartogram: continuous deformation
(Gastner, Newman 2004)

Map-based Statistical Visualization



Dorling cartograms: disks of proportional sizes

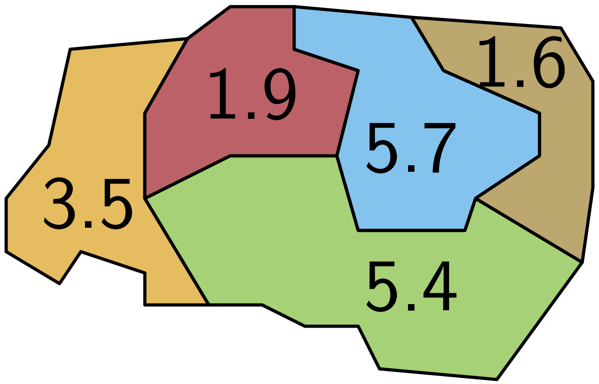
Map-based Statistical Visualization



rectangular cartograms: each region is a rectangle (Raisz 1934)

Problem Definition

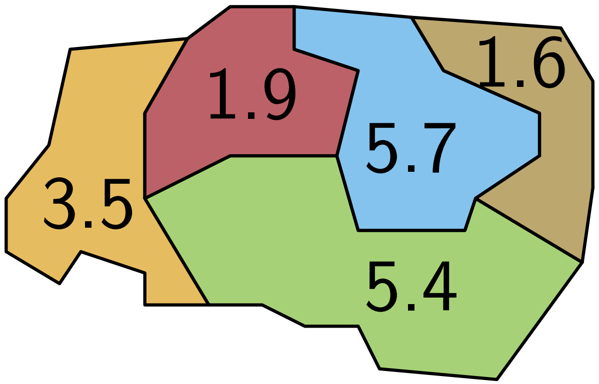
Input: political map M (subdivision of a rectangle), and a positive value w_i for each region R_i



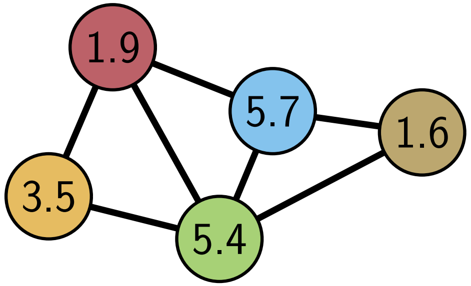
Problem Definition

Input: political map M (subdivision of a rectangle), and a positive value w_i for each region R_i

or: vertex-weighted inner-triangulated plane graph G dual to M, where each vertex v_i is a region R_i, edges connect adjacent regions, and vertex weights are w_i



or

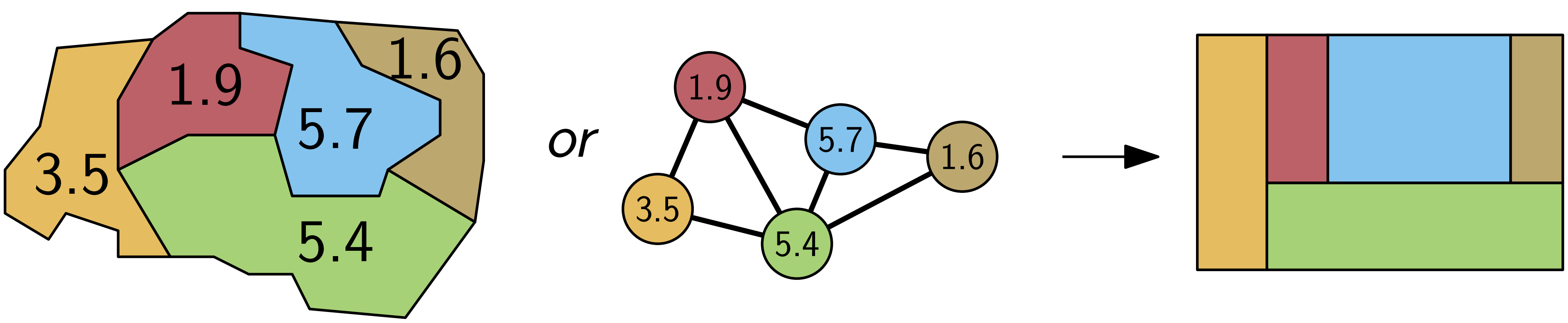


Problem Definition

Input: political map M (subdivision of a rectangle), and a positive value w_i for each region R_i

or: vertex-weighted inner-triangulated plane graph G dual to M, where each vertex v_i is a region R_i, edges connect adjacent regions, and vertex weights are w_i

Goal: distorted map M' equivalent to M so that $|R_i| = w_i$



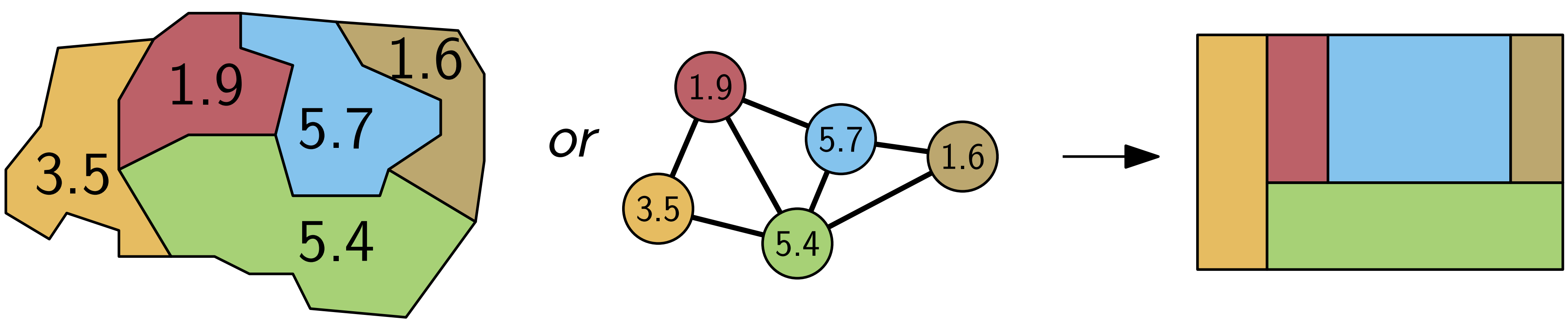
Problem Definition

Input: political map M (subdivision of a rectangle), and a positive value w_i for each region R_i

or: vertex-weighted inner-triangulated plane graph G dual to M, where each vertex v_i is a region R_i, edges connect adjacent regions, and vertex weights are w_i

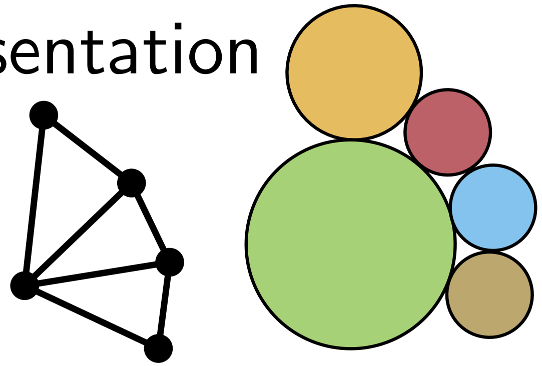
Goal: distorted map M' equivalent to M so that $|R_i| = w_i$

or: area-proportional contact representation of G, where each vertex v_i is represented as a geometric shape s_i of area w_i and two shapes s_i, s_j touch iff $v_i v_j \in E$



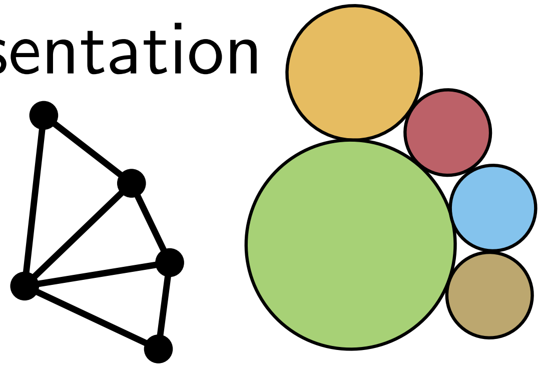
What is known for unweighted graphs?

- every planar graph has a disk contact representation [Koebe 1936]

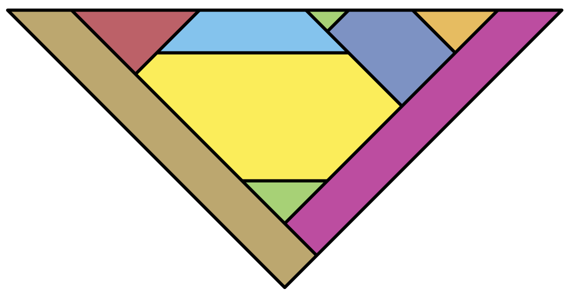


What is known for unweighted graphs?

- every planar graph has a disk contact representation [Koebe 1936]

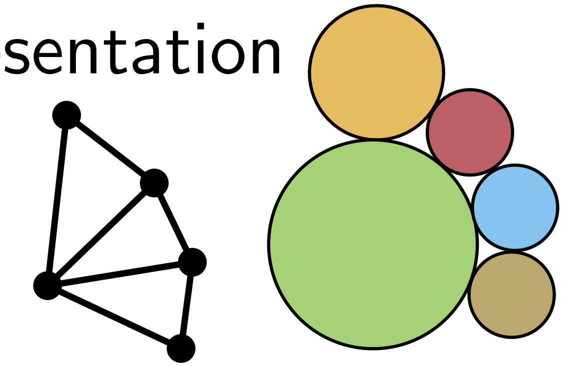


- every planar graph has a hole-free contact representation of convex hexagons [Gansner et al. 2010]

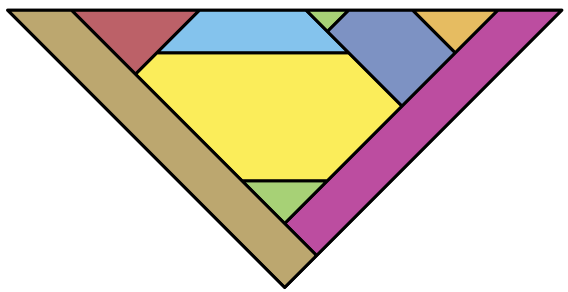


What is known for unweighted graphs?

- every planar graph has a disk contact representation [Koebe 1936]

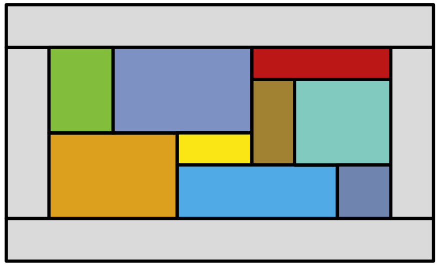


- every planar graph has a hole-free contact representation of convex hexagons [Gansner et al. 2010]



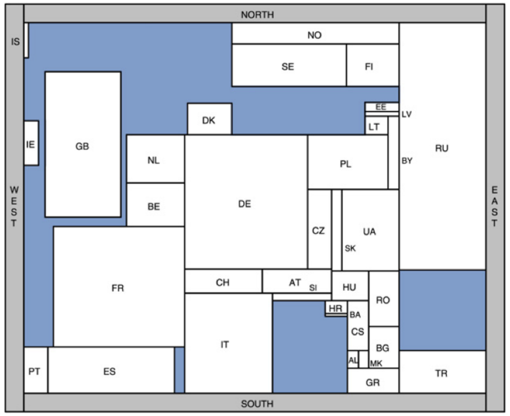
- every planar graph satisfying that
  1. every inner face is a triangle, the outer face a quadrangle
  2. there are no separating triangles

  has a rectangular dual [Koźmiński, Kinnen 1985]



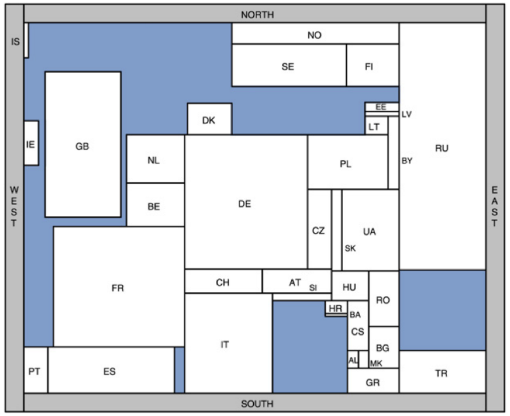
What is known for vertex-weighted graphs?

- rectangular cartograms with low error [van Kreveld, Speckmann 2005]

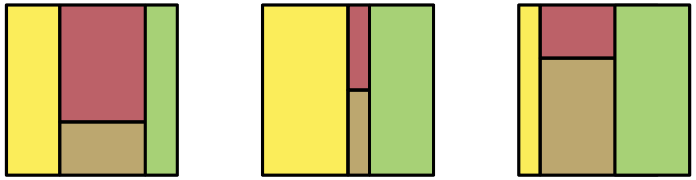


What is known for vertex-weighted graphs?

- rectangular cartograms with low error
  [van Kreveld, Speckmann 2005]

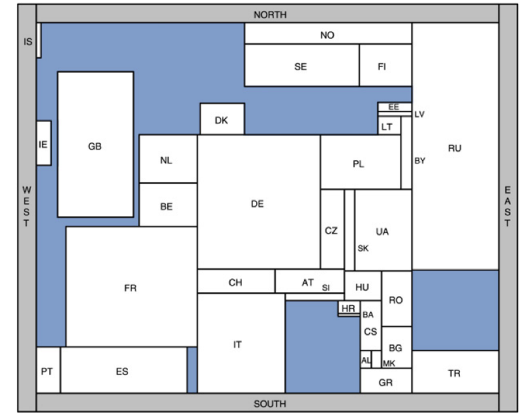


- characterization of area-universal rectangular cartograms
  [Eppstein et al. 2012]

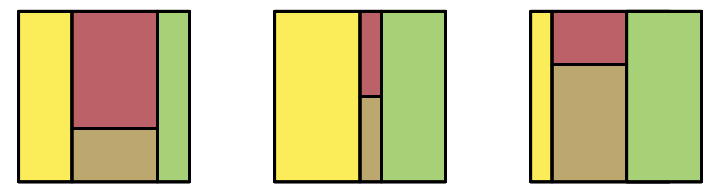


What is known for vertex-weighted graphs?

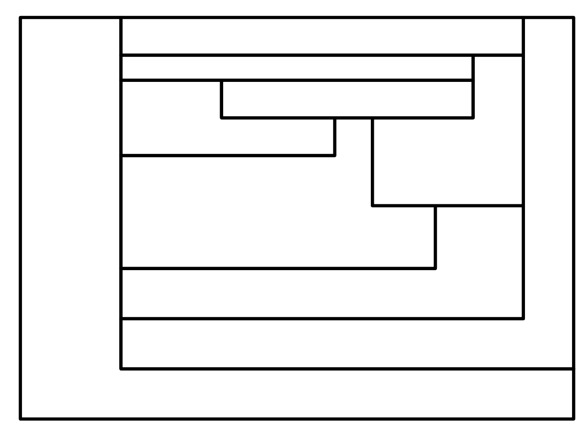
- rectangular cartograms with low error [van Kreveld, Speckmann 2005]



- characterization of area-universal rectangular cartograms [Eppstein et al. 2012]



- every inner-triangulated planar vertex-weighted graph has a rectilinear cartograms with octagons [Alam et al. 2012]
  octagons are actually necessary for some graphs [Yeap, Sarrafzadeh 1993]



Rectilinear cartograms with 10-sided polygons

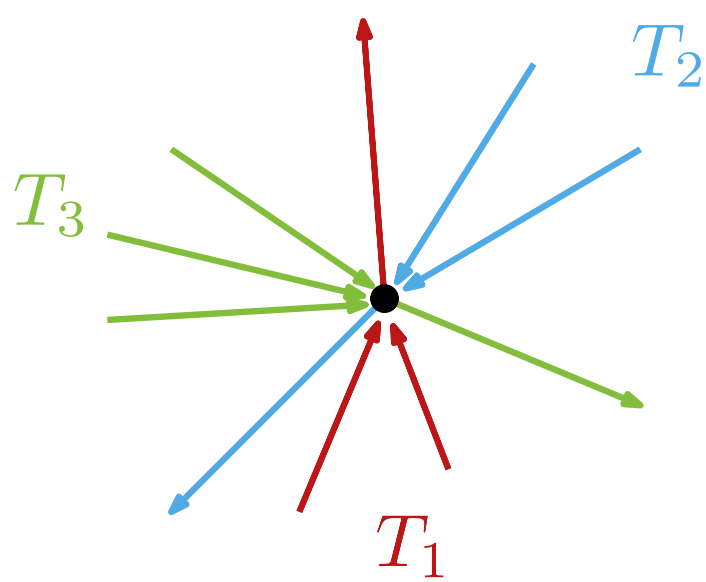
Md. J. Alam, T. Biedl, S. Felsner, A. Gerasch, M. Kaufmann, S. G. Kobourov. *Linear-time algorithms for proprtional contact representations.* Proc. ISAAC 2011.

thanks to Md. Jawaherul Alam for letting me use some of his slides

Schnyder Realizer

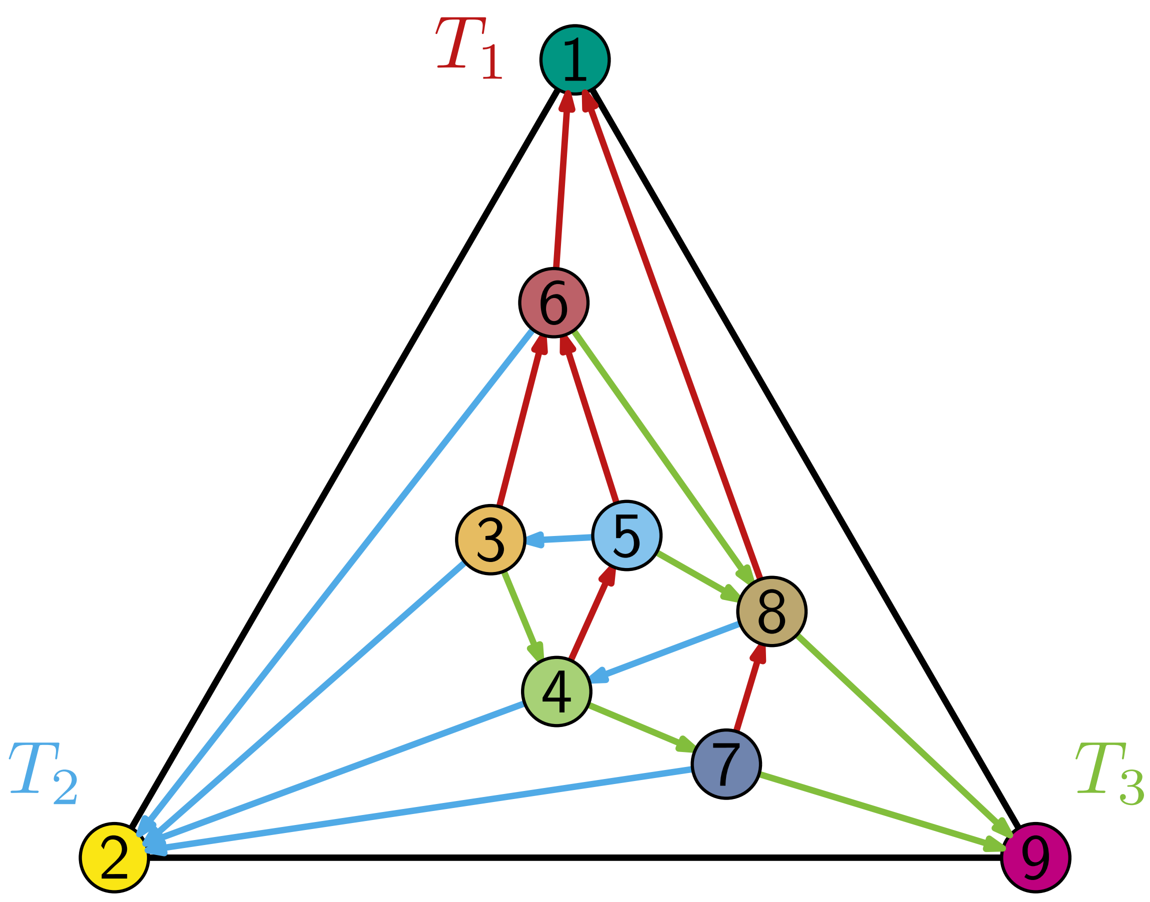
Let G be a fully triangulated planar graph. A **Schnyder realizer** partitions the internal edges into three sets T_1, T_2, T_3 of directed edges so that

- every internal vertex v has exactly one edge in each T_i^{out}
- the ccw ordering of edges around any v is $T_1^{\text{in}}, T_3^{\text{out}}, T_2^{\text{in}}, T_1^{\text{out}}, T_3^{\text{in}}, T_2^{\text{out}}$



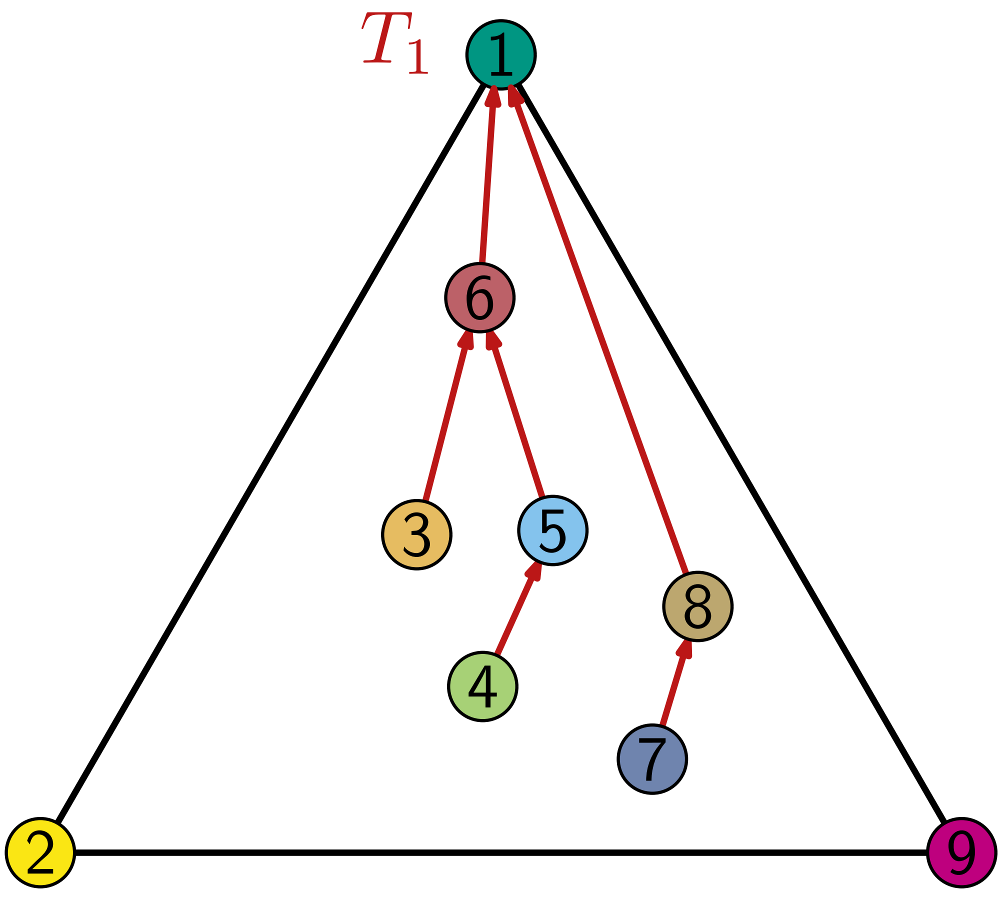
Schnyder Realizer

Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



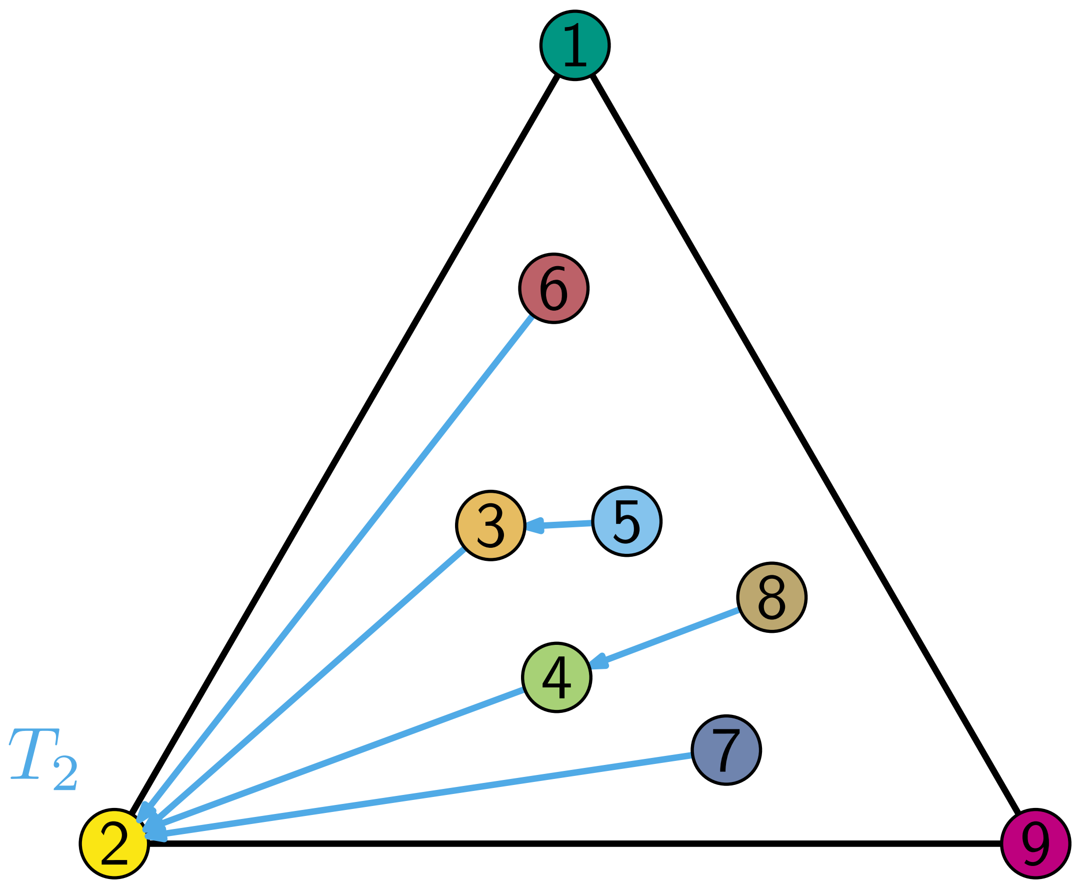
Schnyder Realizer

Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



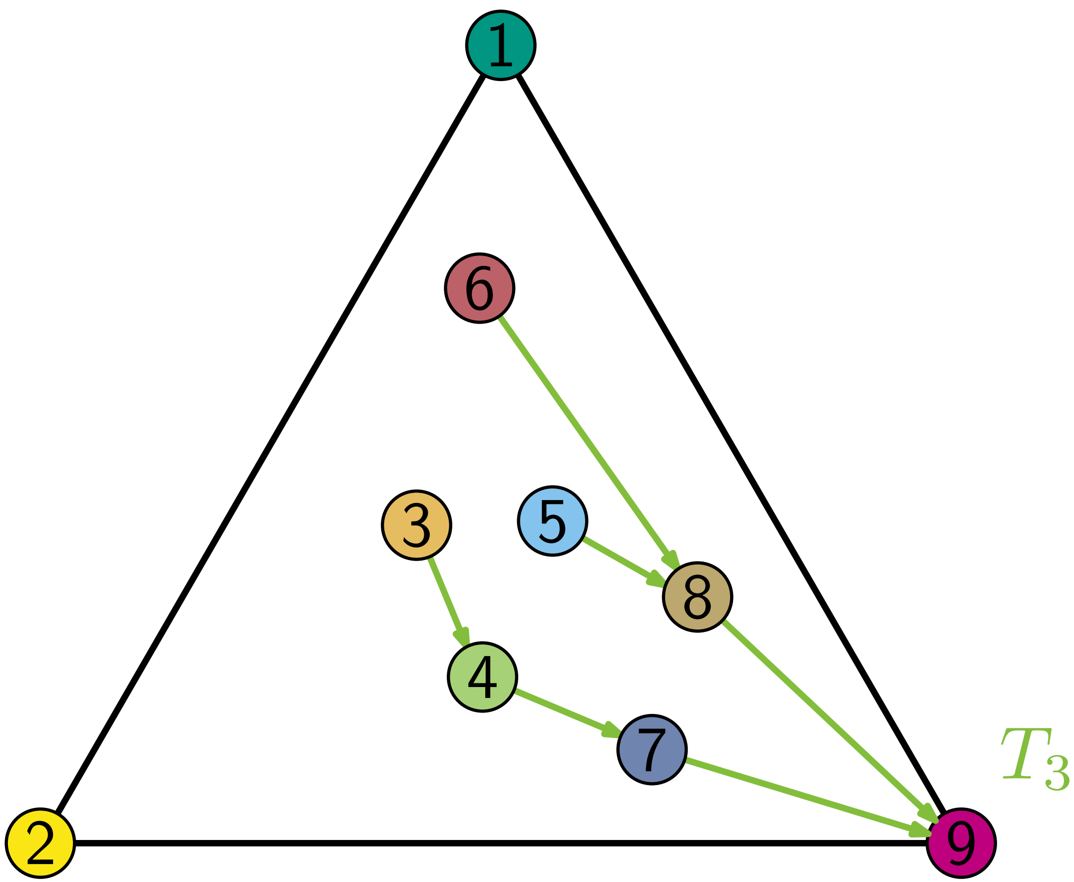
Schnyder Realizer

Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



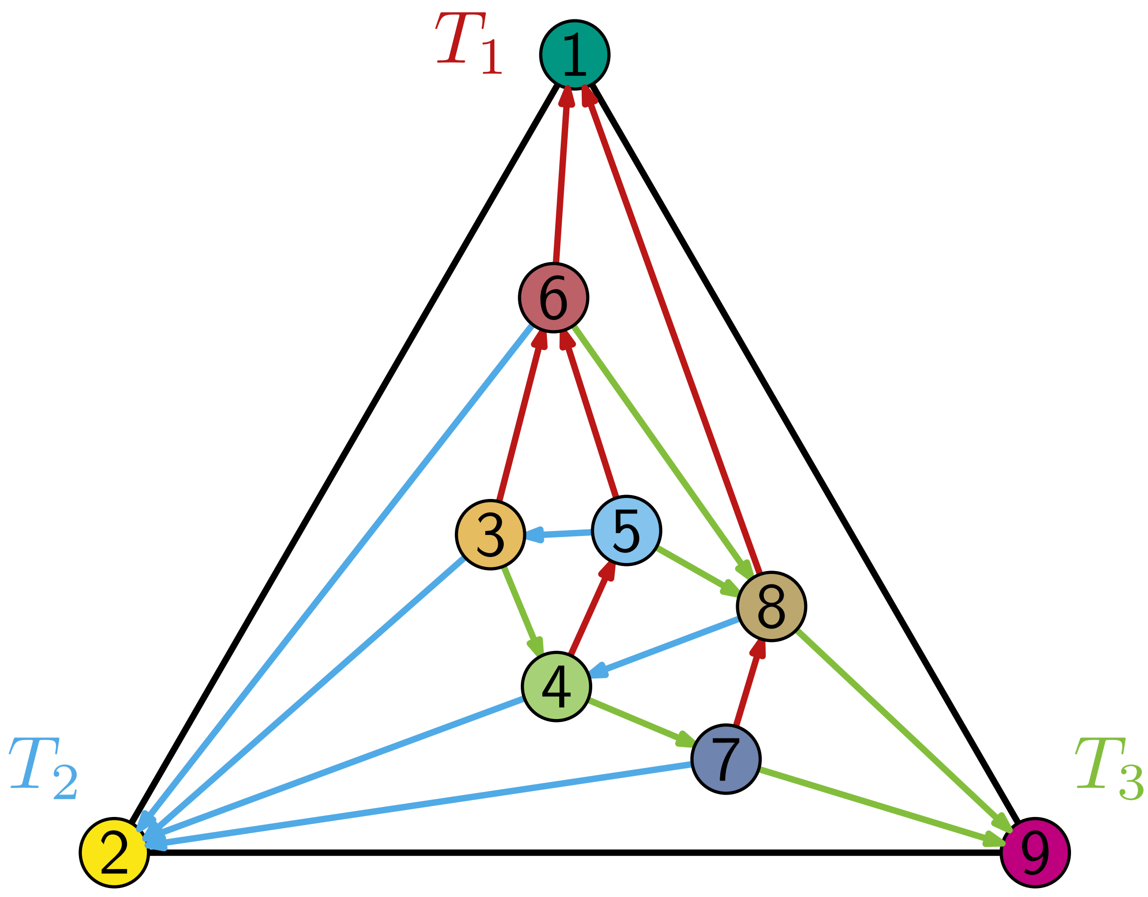
Schnyder Realizer

Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



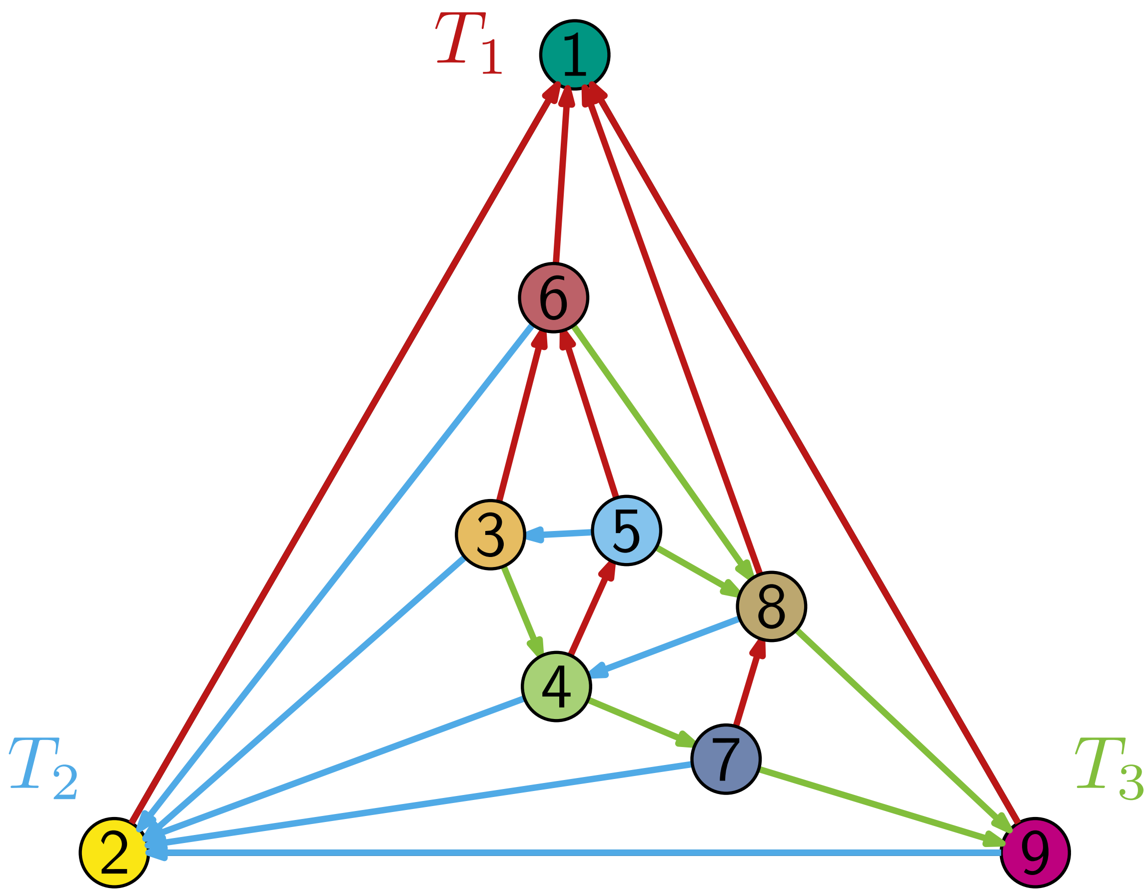
Schnyder Realizer

Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



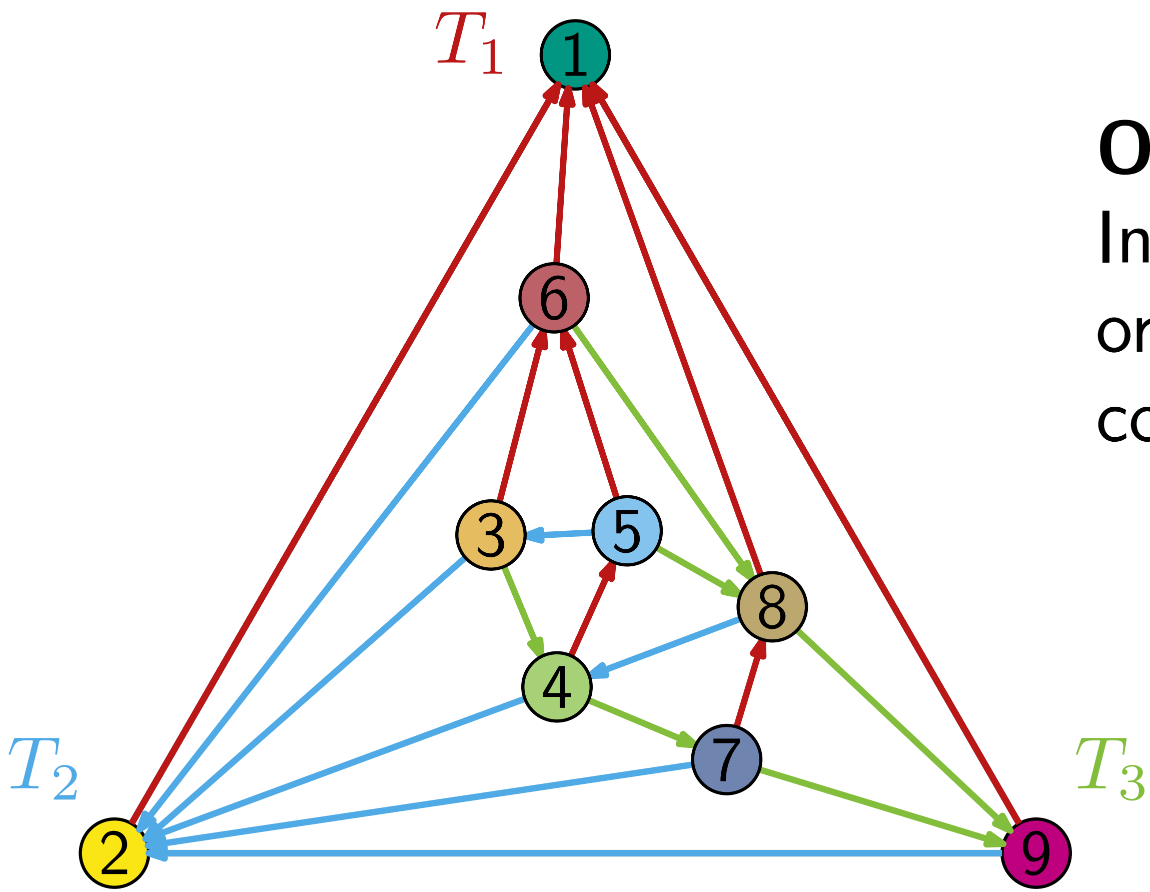
Schnyder Realizer

Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



Schnyder Realizer

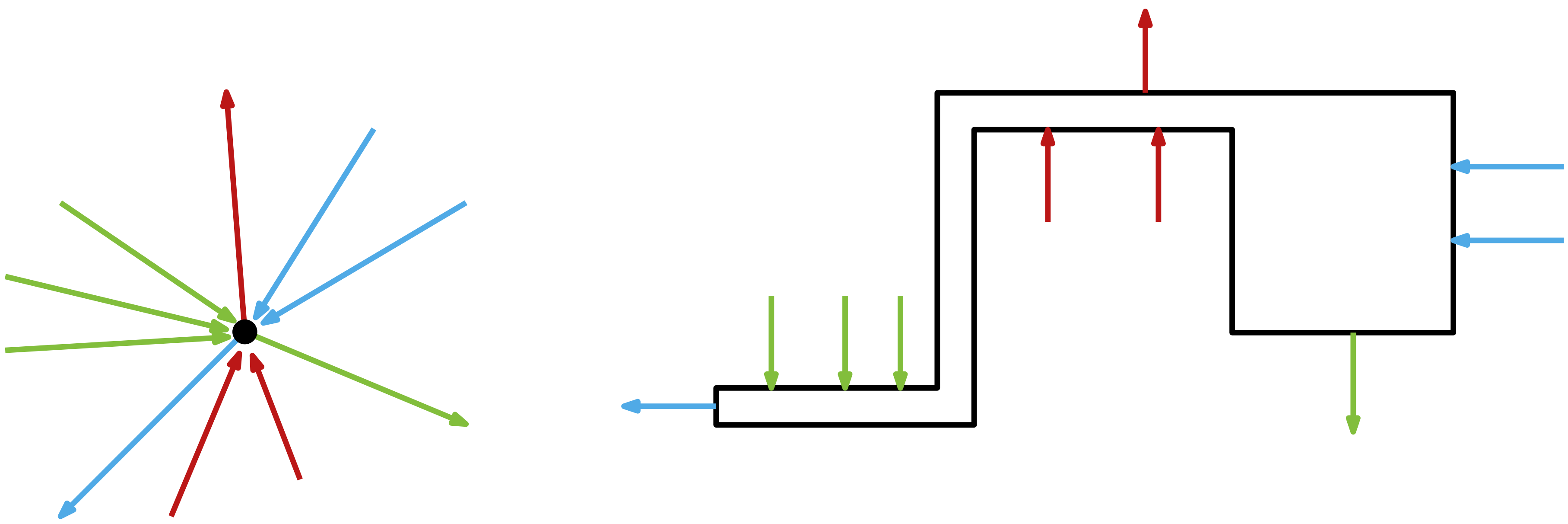
Theorem: Each set T_i $(i = 1, 2, 3)$ is a spanning tree of the inner vertices and one outer vertex.
Every triangulated graph has a Schnyder realizer and it can be computed in $O(n)$ time.



Observation:
In the left-to-right DFS order of T_1, parents in T_2 come before their children.

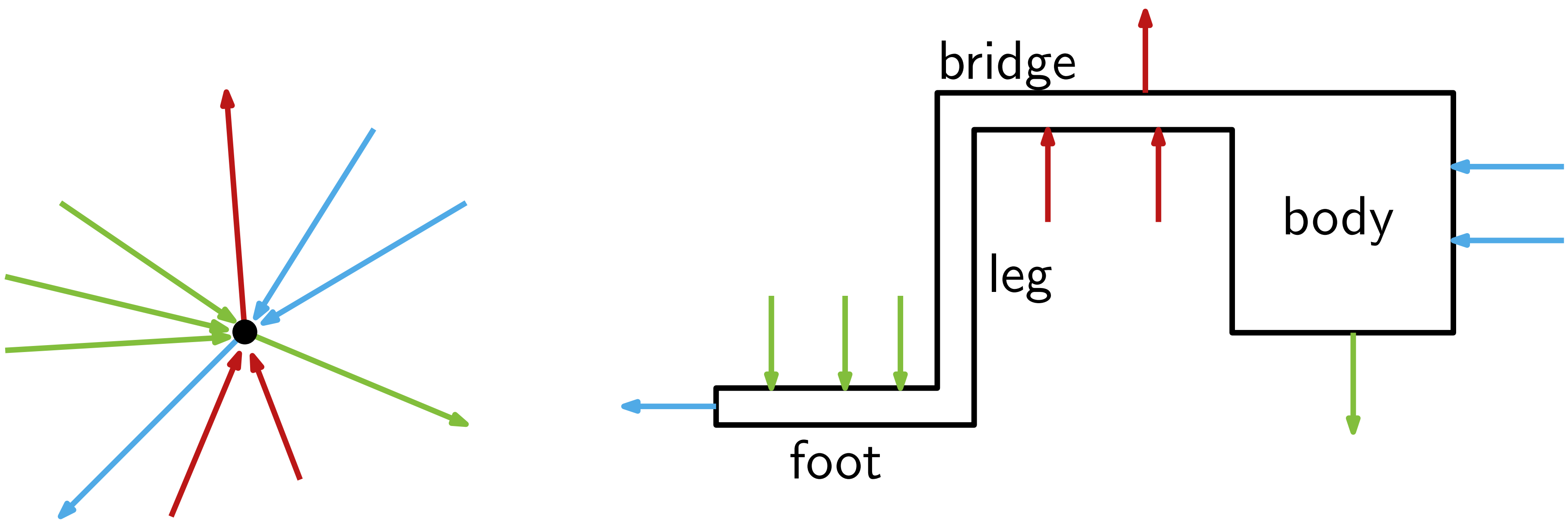
Polygon shapes

Every vertex is represented by a rectilinear 10-gon.



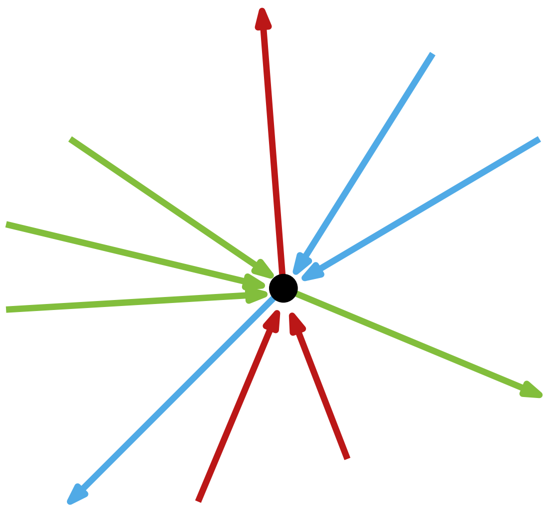
Polygon shapes

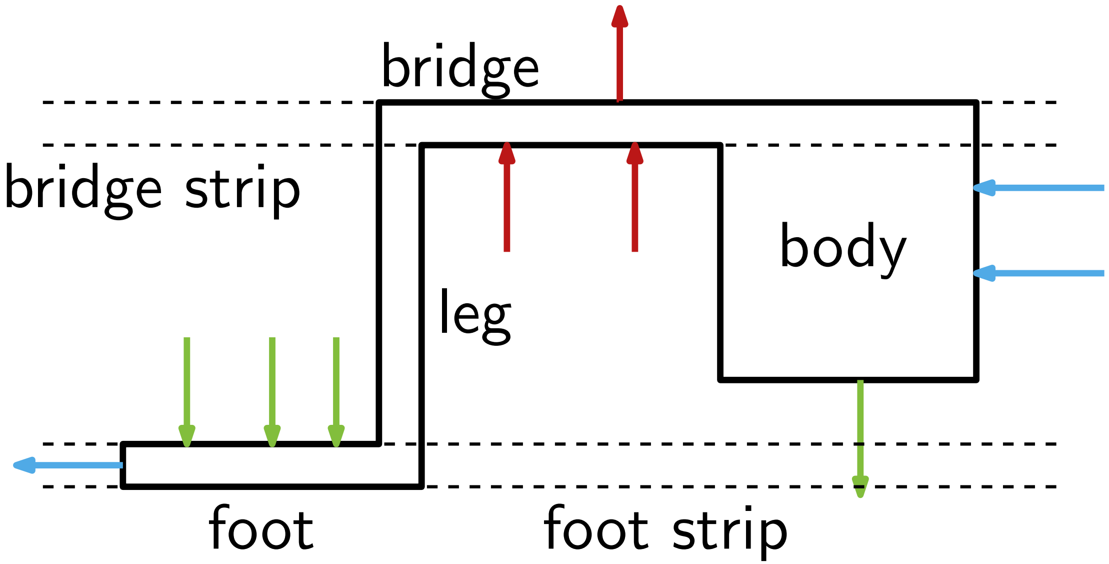
Every vertex is represented by a rectilinear 10-gon.



Polygon shapes

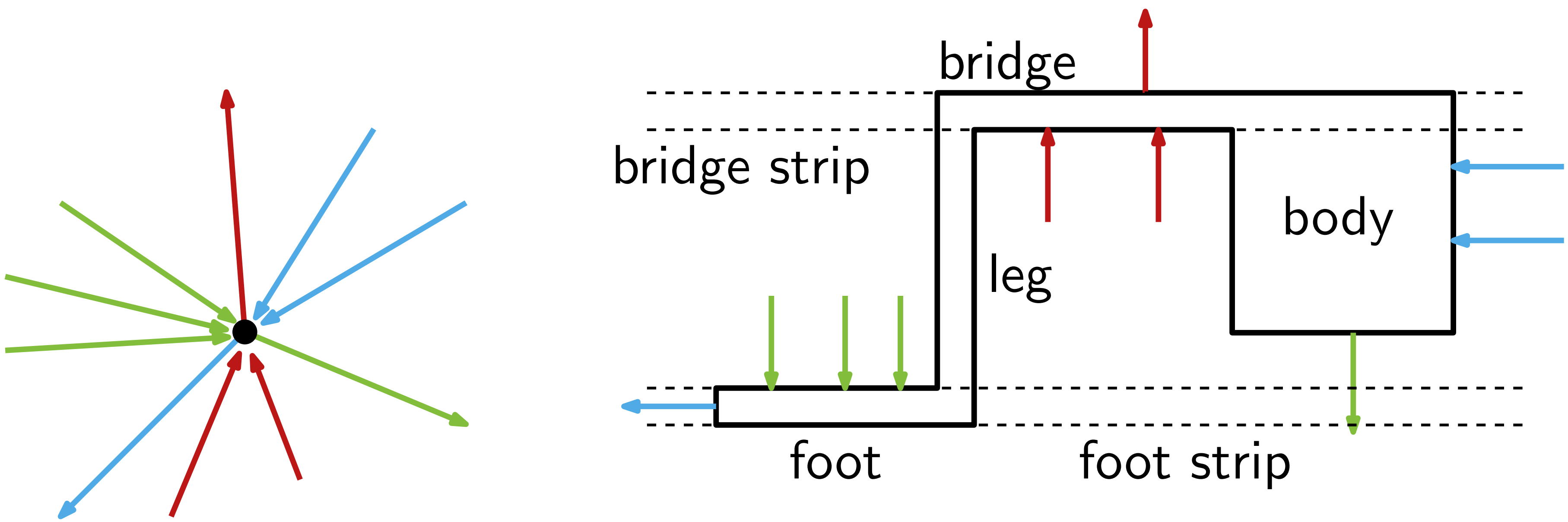
Every vertex is represented by a rectilinear 10-gon.





Polygon shapes

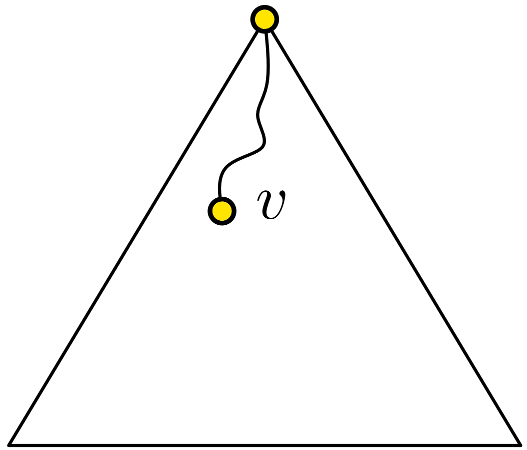
Every vertex is represented by a rectilinear 10-gon.



Foot, leg, and bridge use small area;
the body carries almost all weight.

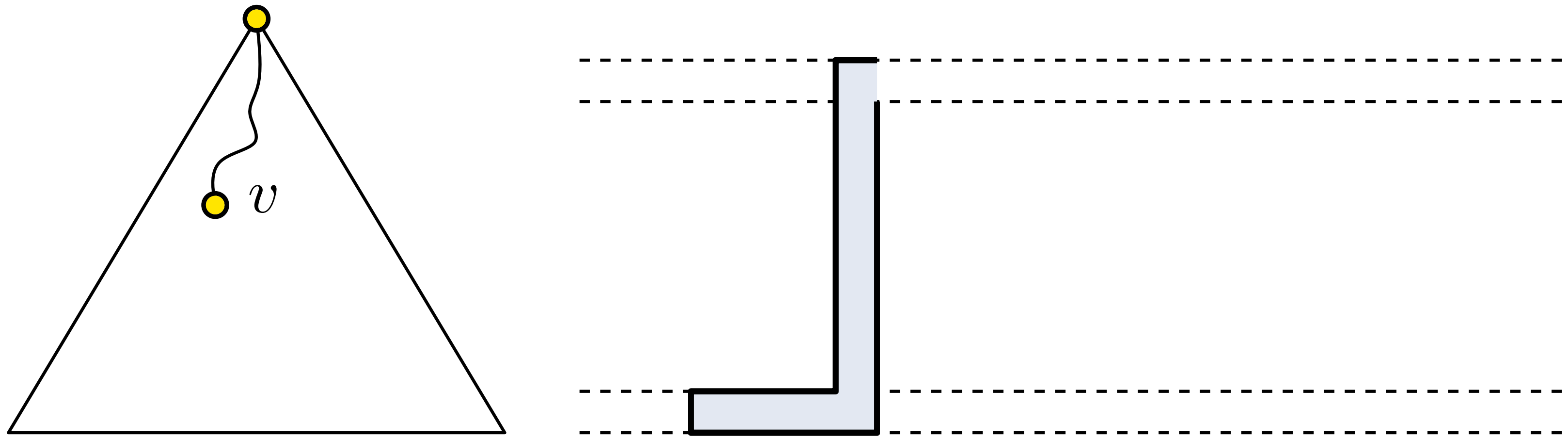
Layout Algorithm

Compute polygons in left-to-right DFS order of T_1.



Layout Algorithm

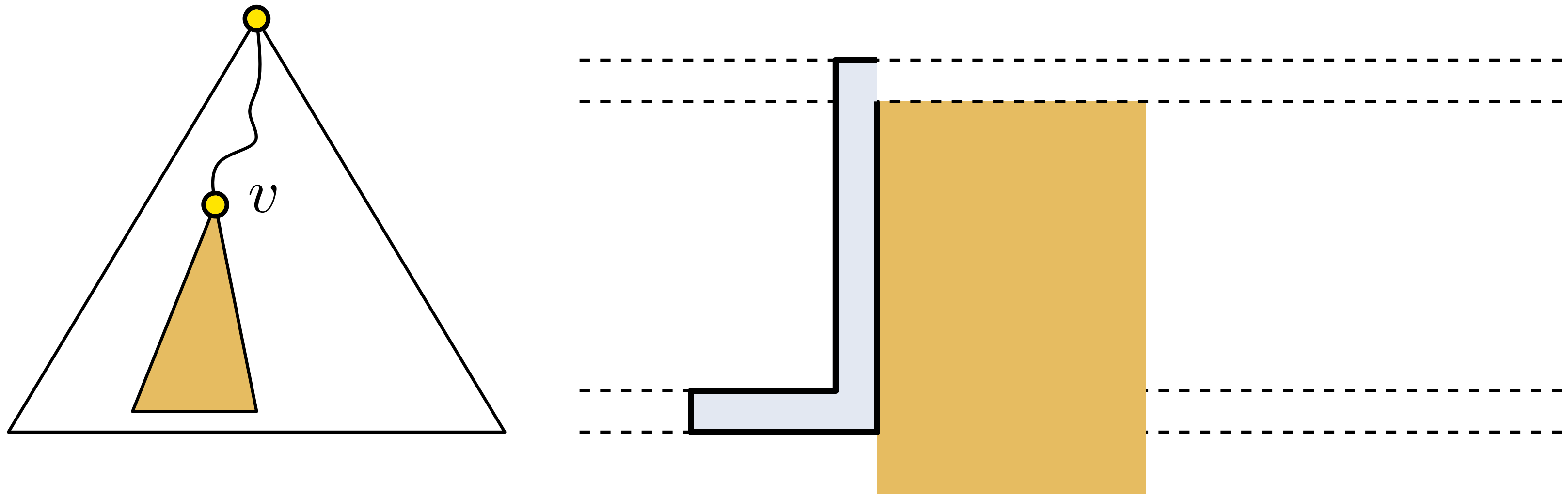
Compute polygons in left-to-right DFS order of T_1.



- fix foot, leg, and bridge strip

Layout Algorithm

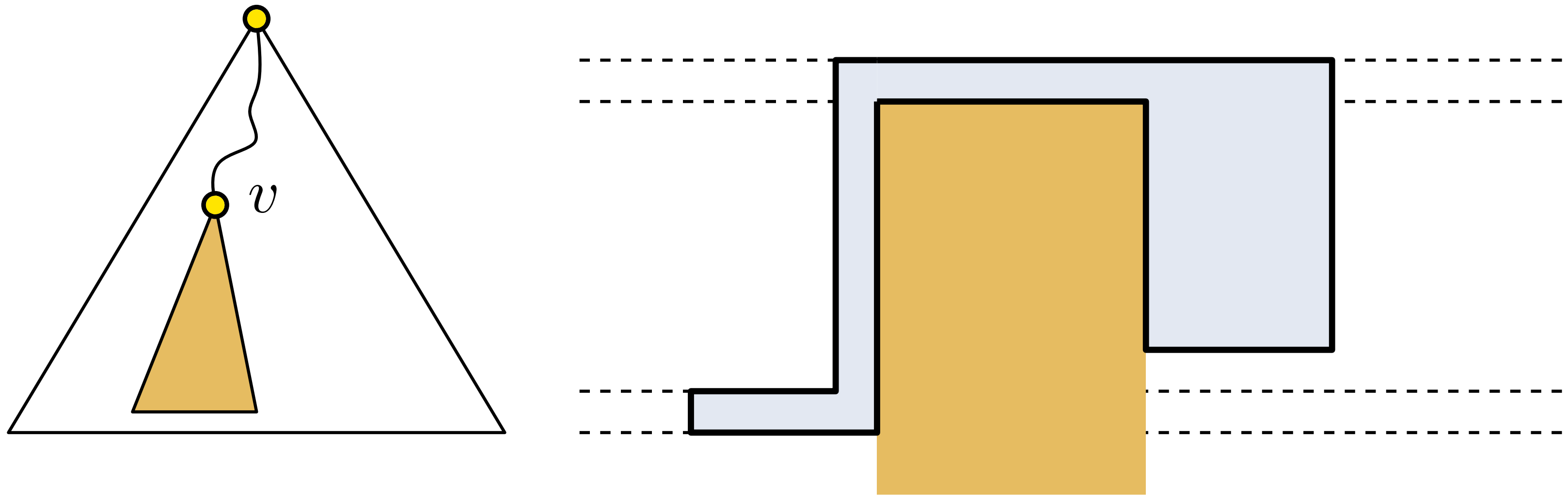
Compute polygons in left-to-right DFS order of T_1.



- fix foot, leg, and bridge strip
- compute child polygons

Layout Algorithm

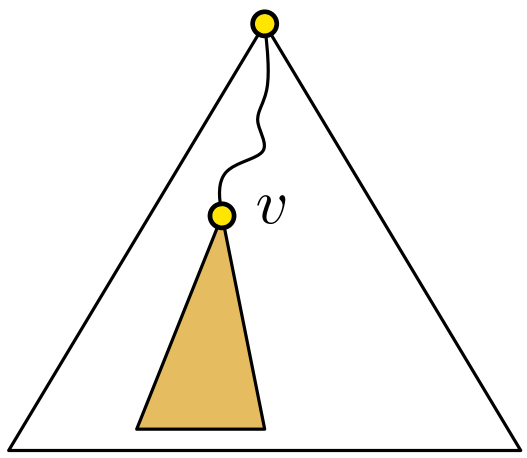
Compute polygons in left-to-right DFS order of T_1.

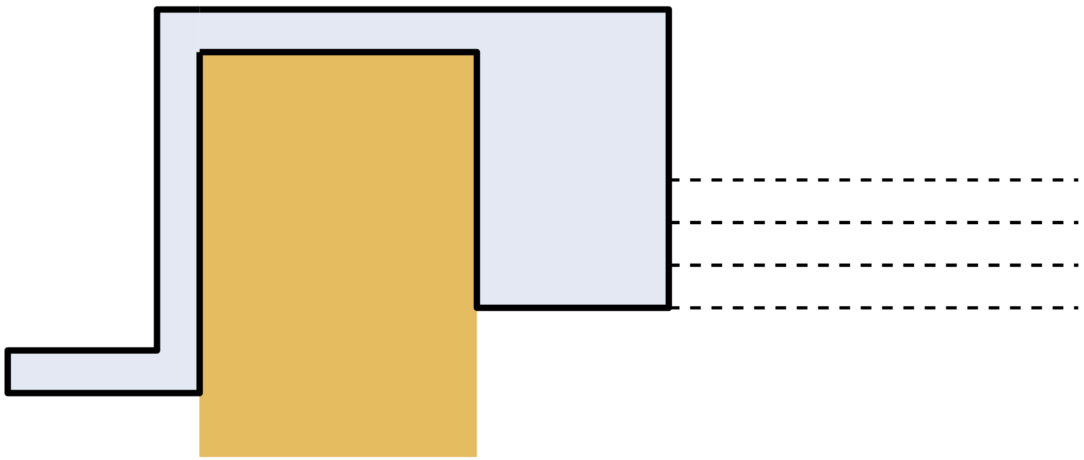


- fix foot, leg, and bridge strip
- compute child polygons
- fix bridge and body

Layout Algorithm

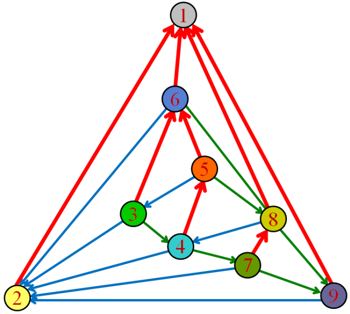
Compute polygons in left-to-right DFS order of T_1.

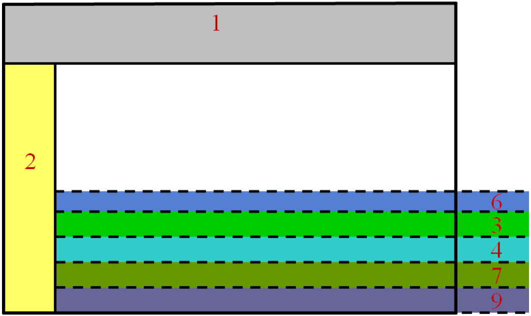




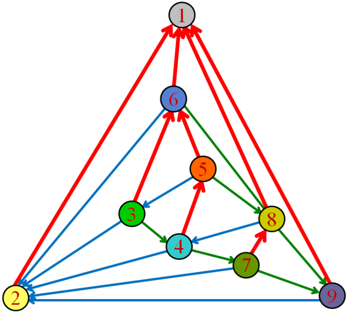
- fix foot, leg, and bridge strip
- compute child polygons
- fix bridge and body
- fix foot strips for children in T_2

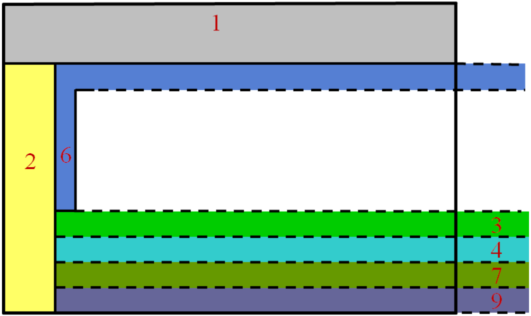
Example



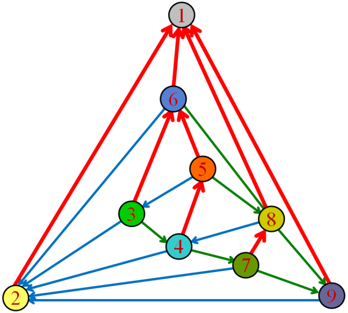


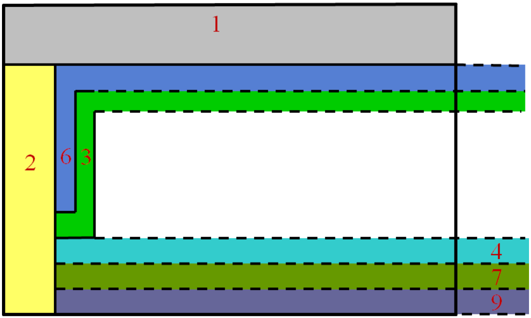
Example



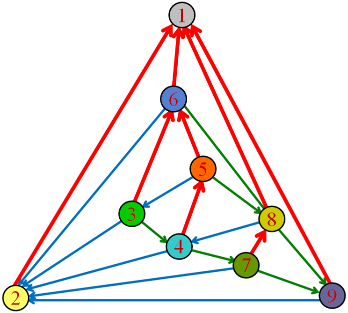


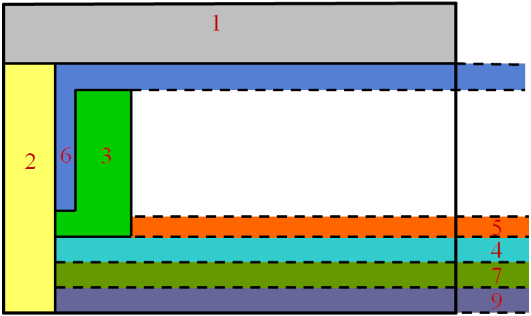
Example



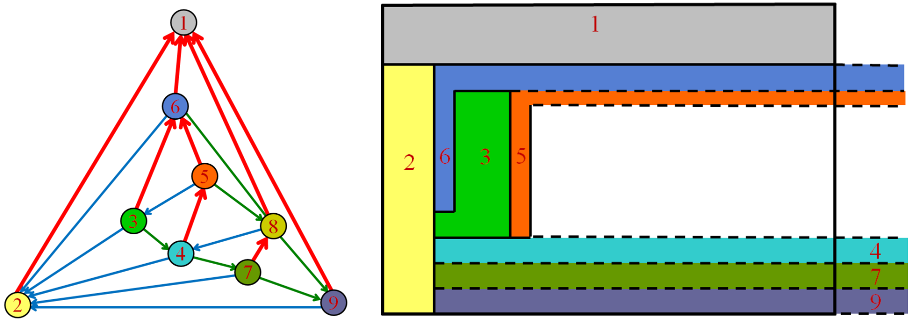


Example

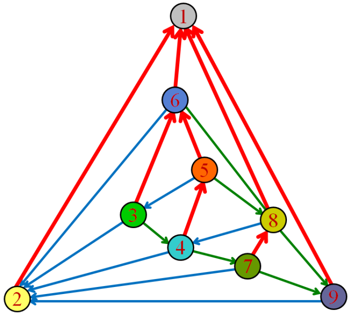


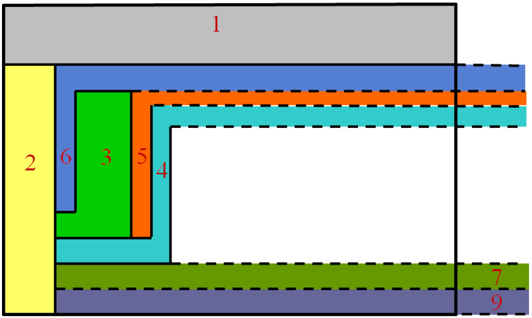


Example

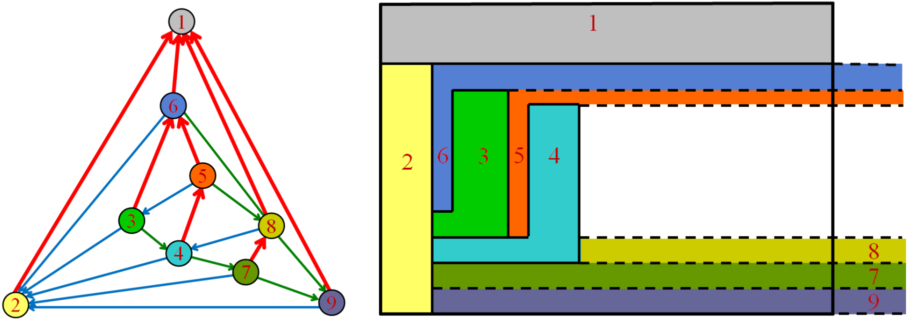


Example

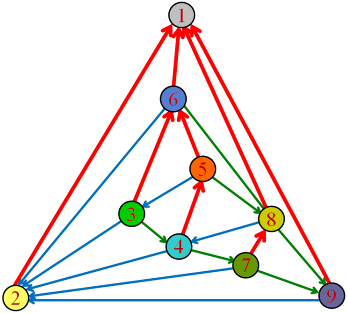


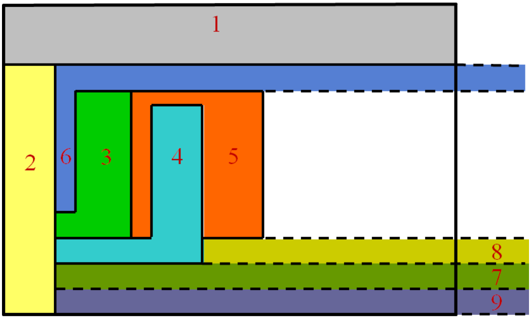


Example

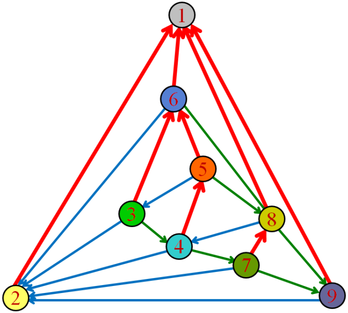


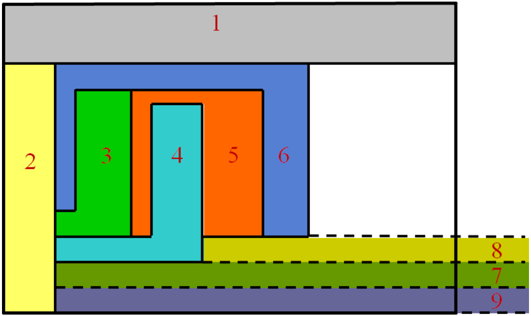
Example



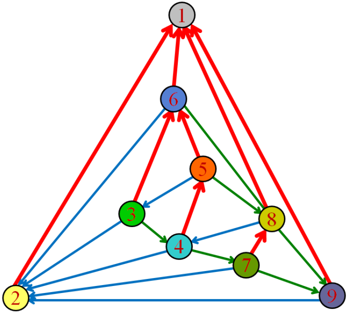


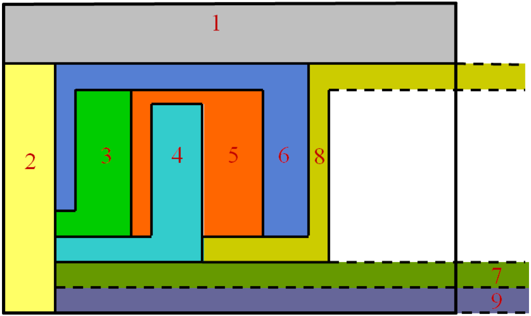
Example



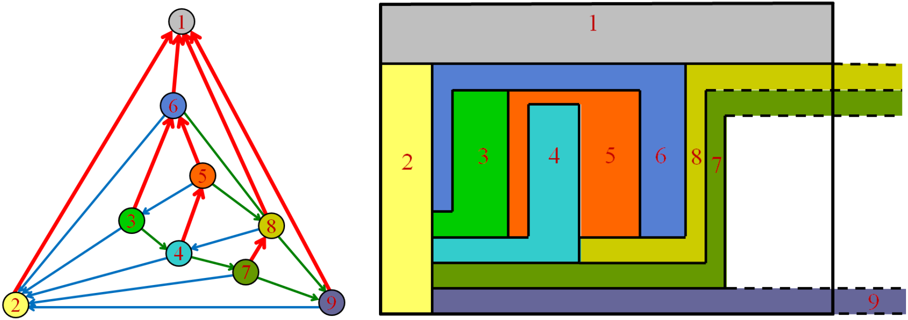


Example

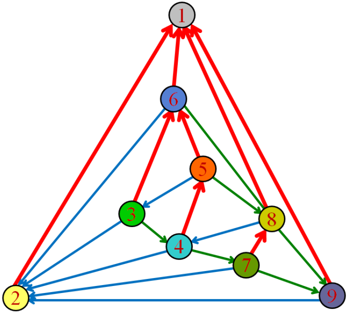


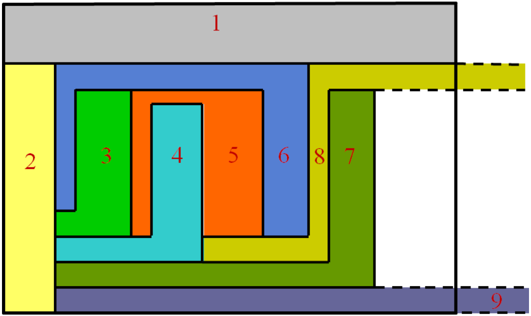


Example

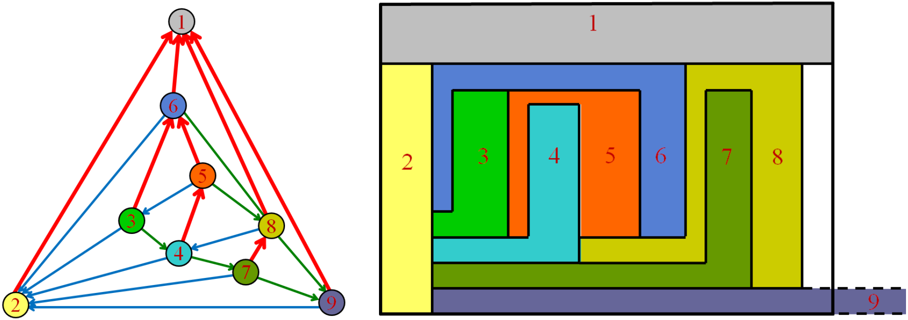


Example

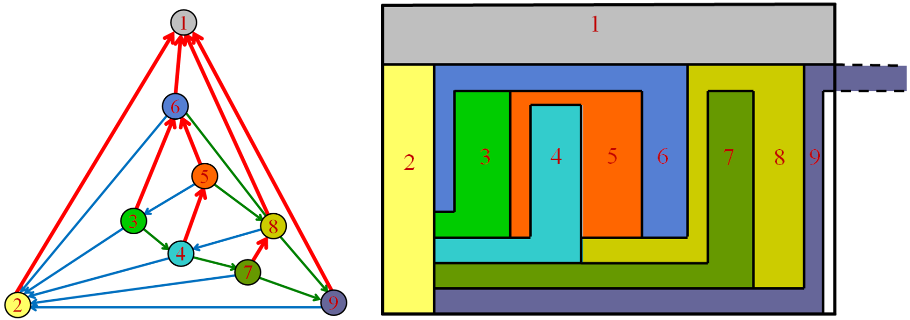




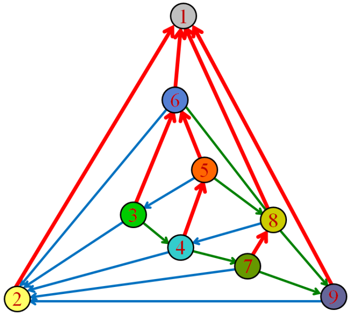
Example

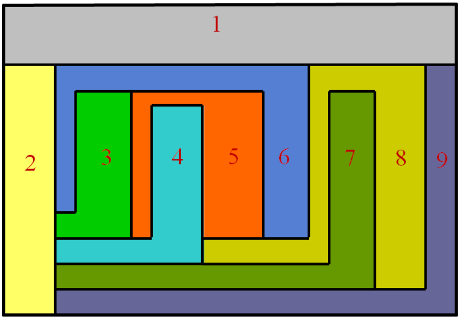


Example



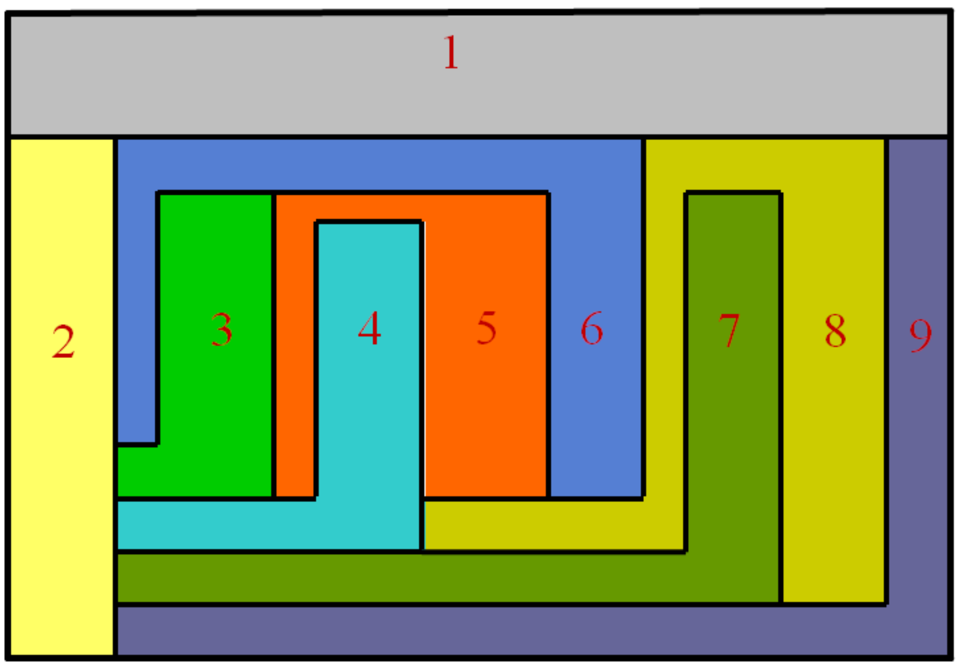
Example





Summary

- every vertex-weighted planar triangulated graph can be drawn as a rectilinear contact representation with 10-gons
- running time of the algorithm is $O(n)$
- actually 8-sided polygons are always sufficient, but a constructive algorithm is still missing



Lust auf mehr?

Regelmäßige Master-Vorlesungen:

- Algorithmische Kartografie (erstmals SS 2013)
- Algorithmische Geometrie (im SS, nicht 2013)
- Algorithmen zur Visualisierung von Graphen (im WS)

Themengebiete für praktische und theoretische Abschlussarbeiten:

- Geovisualisierung
- Graphenzeichnen
- Algorithmische Geometrie
- ...

Kontakt:

Martin Nöllenburg `noellenburg@kit.edu`