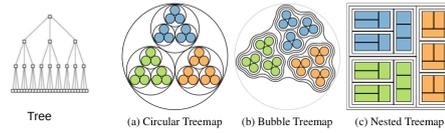


Bubble Treemaps for Uncertainty Visualization

Jochen Görtler,
Christoph Schulz,
Daniel Weiskopf

Presented by Haoran Yu

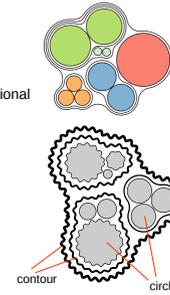
Recall: Treemaps



- Tree data has different visual encodings
- Each encoding has different requirements for space usage
- Each encoding has its advantage under certain specific visualization tasks
- Treemap uses the containment mark to encode parent-child relationship
 - Child nodes are nested inside the parent node

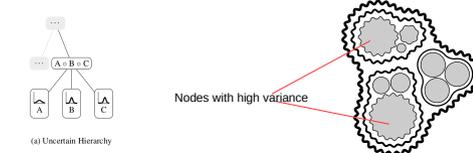
Bubble Treemaps

- Advantages:
 - More compact than other treemaps
 - Improve readability
 - Additional channels used to encode additional variables
- Additional channels:
 - Area for circles
 - Curvature for contours
 - Line thickness
- Terminology:
 - Contour = containment mark
 - Tree data is an e.g. of hierarchical data



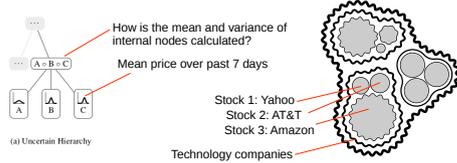
Visual Encoding

- Bubble Treemaps can visualize data with uncertainty:
 - Each node has 2 quantitative attributes: Mean and Variance
 - The mean of a node is encoded as the area channel
 - The variance is encoded as the curvature channel
 - The grouping of nodes can be encoded using color



Bubble Treemap Use Case

- S&P 500 Index
 - Financial data storing stock prices over a period of time
 - Data are grouped by industry or sector
 - Large variance of a stock price indicates company is unstable, either growing or declining



Algorithm for Bubble Treemap

1. Propagation of Mean and Variance to parent nodes
2. Create contours around subhierarchy (i.e. enclosed circles or child contours)
3. Create layout of nodes or contours by force-based circle packing

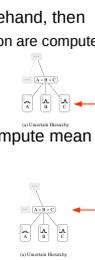


Propagation of Mean and Variance

- If mean and variance were not computed beforehand, then
 - for each leaf node i , the mean and standard deviation are computed as follows:

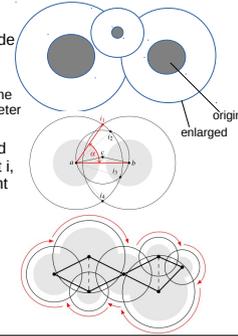
$$\mu_i = \int x p_i(x_i) dx_i, \quad \sigma_i = \sqrt{\int x^2 p_i(x_i) dx_i - \mu_i^2}$$
- For each parent node, use its child nodes to compute mean and standard deviation as follows:

$$\mu_{1\dots n} = \sum_i \mu_i, \quad \sigma_{1\dots n} = \sqrt{\sum_i \sigma_i^2}$$
 - where $1\dots n$ are child nodes of the parent
 - This is the additive property



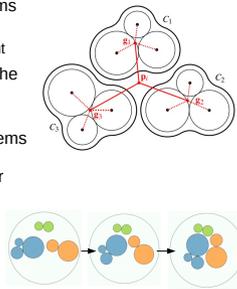
Create Contours

- For each child node, enlarge node by a width d
 - Where d is a combination of parameters $m + w + p$ (details in the paper), plus a smoothness parameter s .
- Intersect the enlarged nodes and find an optimal intersection point i , draw an arc segment at this point
- Use the arc segments at each intersection point to construct a complete contour.
- Recurse

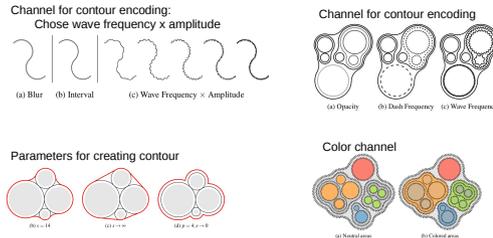


Pack and Create Layout

- For each hierarchy, find the items that are to be packed:
 - All the child nodes inside a parent
- Define a center point p , this is the point where children are to be packed around
- Use physics algorithm to pull items toward the center point until p becomes the center of mass for the packed items.
- Recurse



Design Choices



Summary

- Idiom: Bubble Treemaps
- What Data: Tree (node value or a group of values; parent-child relationships between nodes)
- What Derived: Mean and variance (if not already contained in data) for leaf nodes as well as for internal nodes
- How Encode: map mean value to area of circle; map parent-child relationship to containment; map variance to wave frequency x amplitude curvature for contours.
- Why Tasks: Detect abnormal data point; explore and observe pattern (such as finding the grouping with the largest mean)
- Scale: 3-6 levels of hierarchy, ~500 nodes

Bubble Treemap Critique

- Good:
 - Compactness and readability at the same time
 - Can encode both certain and uncertain data
 - Empty space between contour and nodes gives good perception of structure; labels can be drawn easily
 - Simple marks (circles and lines) are easy to interpret
 - Algorithm can draw smooth boundaries
 - Additional channels are free to use (color for nodes and line style for contours)
 - Encoding for uncertainty (contour waves) is easy to detect
- Bad:
 - Cannot correctly encode non-linearity data or data with conditional dependence
 - Effectiveness of treemaps decreases as hierarchy levels increase - counting the number of levels is difficult.
 - Incorrect parameters used for drawing contours produces bad layout
 - Treemap does not support user interaction, algorithm does not permit update quickly when data change
 - Physics-based packing algorithm produces different layout every time, cannot be used for treemap comparison
 - Nested structure does not guarantee congruent geometry for two identical sub-hierarchies, making pattern detection difficult.