Optimization of Hyperbolic Tree Layout

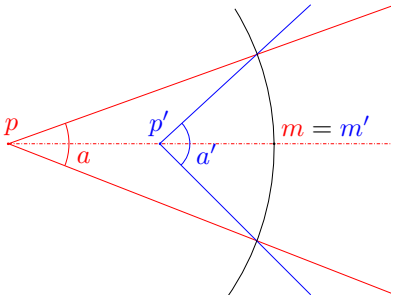
Jiří Zajíc
izajic@fit.vutbr.cz

Since massive expansion of computer science as a tool to process nearly any kind of information, there has always been a need to visualize the information in the nicest and friendliest way possible. Despite the fact that nowadays there are numerous known technologies and methods available in order to achieve this very essential goal, it can still be quite a challenge to effectively and efficiently display extremely large portions of data. Especially when interactive manipulation operations for navigating around are necessary.

Consider the relationship between different pieces of information of hierarchical nature. Useful and versatile way to represent such structure in a graphical form would be a tree. Tree with one root and number of nodes connected to each other by exactly one path. It is a fundamental concept that can be applied in computer science (binary search tree) as well as in many other fields, e.g., biology (evolutionary tree), project management (work breakdown structure), linguistics (phrase structure trees), mathematics (Von Neumann universe). Apparently, a big part of all the information in the world can be hierarchically structured.

It is difficult for user to extract information from large hierarchical structures, as the navigation of the structure is often a burden and content information is hidden inside individual nodes. Also, usually only small part of the monitor display is being used. In 1995, J. Lamping and R. Rao from Xerox Palo Alto Research Center came up with an idea of the Hyperbolic Browser. Browser, that can visualize any tree structure by transforming it into, so called, hypertree. Hypertree uses advantages of hyperbolic plane in non-Euclidean Geometry, providing truly unique displaying and browsing experience.

The idea of Hyperbolic Browser is more than seventeen years old, yet its utilization is seen quite rarely. One of the reasons is its computational complexity consisting of non-trivial mapping of the nodes onto hyperbolic plane, followed by projection on unit disk using the Poincaré model. Among others, three basic attributes have to be calculated for each node in order to map it onto hyperbolic plane: *vertex* coordinates, *middle point* to know the distance from its parent, and *angle of the wedge* the node has available for displaying its children.



Vertex, Middle Point, Wedge

When there is a truly user-friendly interface for browsing a hierarchical structure displayed in such manner — consider for example increasingly popular web environment where DOM nodes represent actual nodes in the tree — layout process can be computationally quite demanding. This presentation formally defines the tree that can be visualized in hyperbolic plane. And it also presents a formal grammar where nonterminals and terminals represent collapsed (unexplored) and expanded (explored) nodes, where deriving by given rules results in estimates of the node attribute values. Generated sentence unburdens the computational layout algorithm and speeds up the whole layout proces.