# Polytopes

Henrique Hepp

May 7, 2014

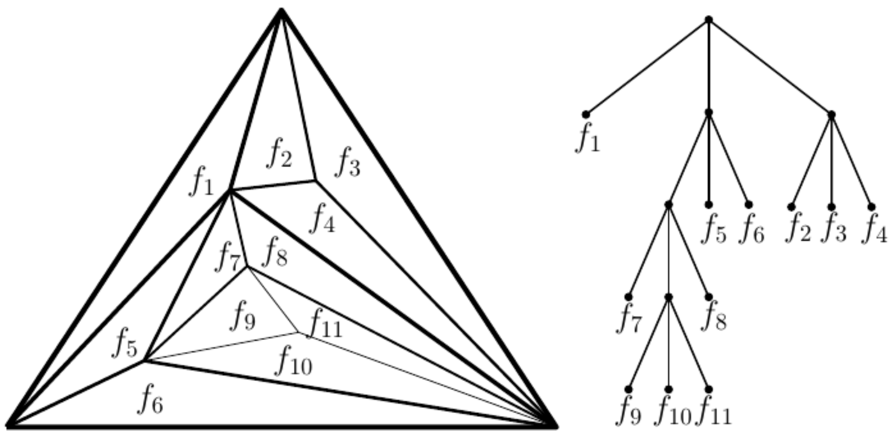
Algorithm based on the article "Embedding Stacked Polytopes on a Polynomial-Size Grid".

The algorithm can be divided in:

1. Receive the entry.
2. Heavy caterpillar decomposition.
3. Balance the tree.
4. Get the coefficients.
5. Lift the graph to a polytope.

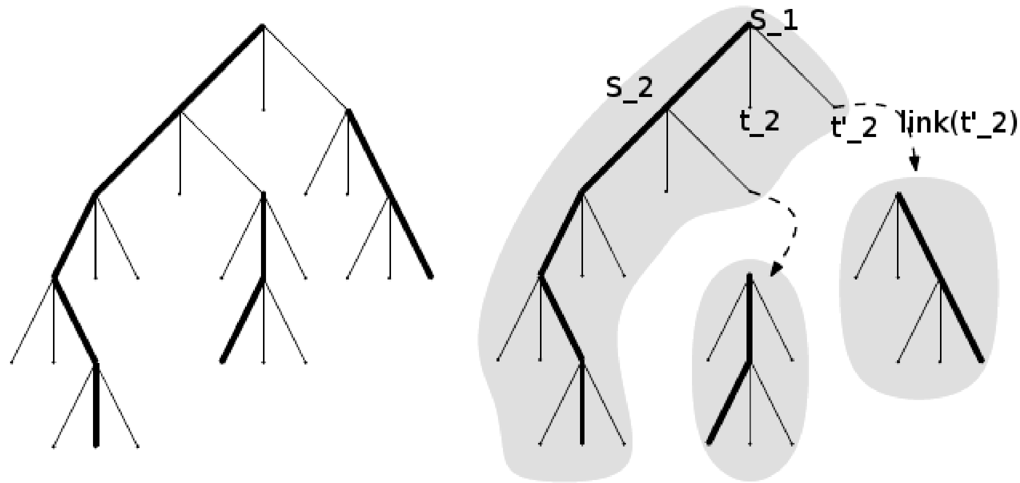
## 1 The Entry of the algorithm

It is given to the algorithm a 3 connected planar graph and a tree representation of the graph, as showed below.




## 2 Heavy caterpillar decomposition

The tree representation is decomposed in *heavy paths*. Resulting sub-trees called *caterpillars*. When a node lies on a heavy path it is called a *spine node*, otherwise is a *tree node*. The spine nodes are labelled by $s_1$ (root), $s_2$, ..., $s_i$,...,$s_\perp$. The children from $s_i$ are $s_{i+1}$, $t_{i+1}$ and $t'_{i+1}$. In $t$ are stored a pointer link($t$) to a other caterpillar.




# 3 Balance the tree

**function** BALANCE(C)
    **Input:** A caterpillar $C$ from the heavy caterpillar decomposition of $\mathcal{T}(G)$.
    **Output:** Weights for the nodes of $\mathcal{T}(G)$.
    **for all** $t_i$, $t'_i$ in $C$ **do**
        BALANCE(link($t_i$))
        BALANCE(link($t'_i$))
        **if** $w(t_i) > w(t'_i)$ **then**
            relabel $t_i \leftrightarrow t'_i$
        **end if**
        $w(t_i) = w(t'_i)$
        add $(w(t_i) - w(t'_i))$ to the weight of $s_\perp$ in link $(t_i)$
        add $(w(t'_i) - w(t_i))$ to the weight of $\text{link}^{-1}(C)$
    **end for**
**end function**

# 4 Get the coefficients

If $v_i$ is the vertex stacked on the face $v_j v_k v_l$ then:

$$\alpha_{ijkl} = \frac{1}{w(t_u)}$$

Where $w(t_u)$ is the weight of the subface $v_j v_k v_l$ obtained by the function BALANCE.

The weights are now updated with the coordinate values. First the coordinates from the vertices are rounded down. $r_i = (\lfloor x_i \rfloor, \lfloor y_i \rfloor)$.

The weights are:

$$\dot{w}_{ij} = \sum_{\{i,j,k,l\} \in S} \lfloor Y \alpha_{ijkl} \rfloor w_{ij}^{kl}$$

The coefficient $\alpha_{ijkl}$ is scaled by multiplying it with $Y = 4n^2$, $n$ is the number of vertices.

$$w_{ij}^{kl} = [i,k,l][j,k,l]$$

$$[i,j,k] = \det \begin{pmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \\ 1 & 1 & 1 \end{pmatrix}$$

# 5 The lifting