Compact 3D Grid Drawings of Trees

Irene Parada, Kees Voorintholt, and Zeyu Zhang

TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

CONTRIBUTION

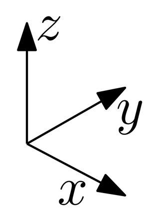
We show that perfect binary trees with $n-1$ vertices can be optimially embedded in 3D, that is, they admit a straight-line drawing on a $\sqrt[3]{n}$ by $\sqrt[3]{n}$ by $\sqrt[3]{n}$ grid without intersecting edges:

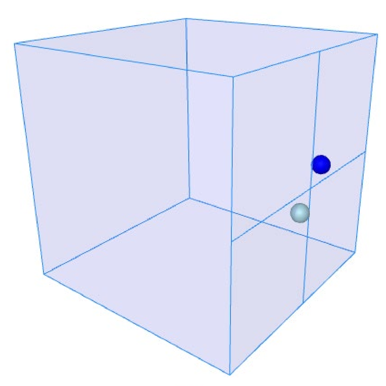
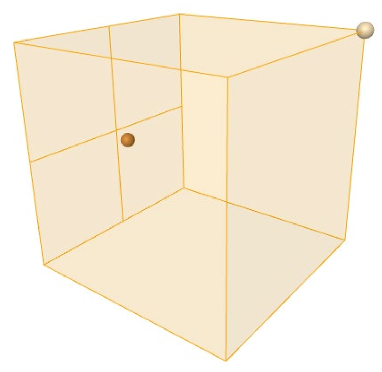
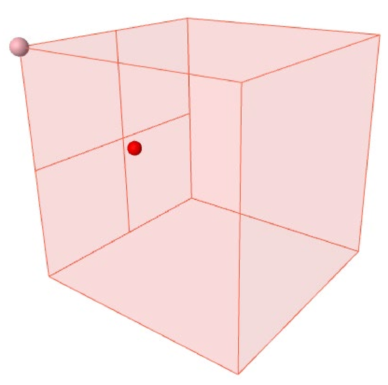
Theorem: *The perfect binary tree with height $k = 3x - 1$ for $x \in \mathbb{N}$ with $n = 2^{k+1} - 1$ vertices has a compact embedding on the $\sqrt[3]{n} \times \sqrt[3]{n} \times \sqrt[3]{n}$ grid.*

To show it, we adapt a recursive approach used by Akitaya *et al.* [1] for the 2D case: they recursively construct two types of compact embeddings of perfect binary trees on a square grid.

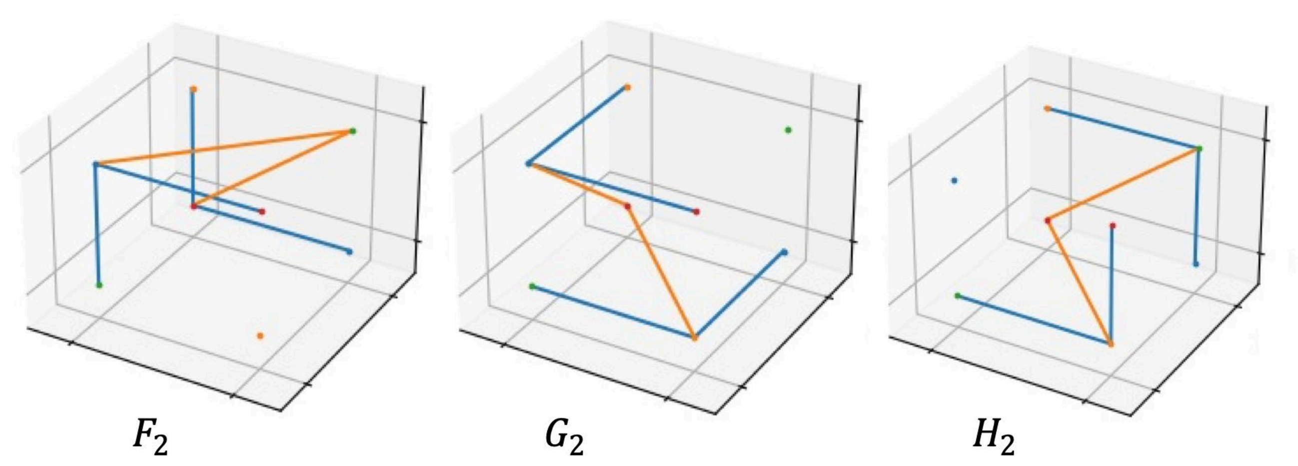
We recursively construct three blocks F_k, G_k, and H_k which are compact embeddings of a perfect binary tree T_k of height k.
The critical difference between the blocks is the placement of the root of T_k and the empty node.

OVERVIEW OF THE BLOCKS



	Formulas	Sketch
F_k	Root node: $(2\cdot 2^{\frac{k-2}{3}}, 2^{\frac{k-2}{3}}+1, 2^{\frac{k-2}{3}}+1)$ Empty node: $(2\cdot 2^{\frac{k-2}{3}}, 2^{\frac{k-2}{3}}, 2^{\frac{k-2}{3}})$	
G_k	Root node: $(1, 2^{\frac{k-2}{3}}+1, 2^{\frac{k-2}{3}})$ Empty node: $(2\cdot 2^{\frac{k-2}{3}}, 2\cdot 2^{\frac{k-2}{3}}, 2\cdot 2^{\frac{k-2}{3}})$	
H_k	Root node: $(1, 2^{\frac{k-2}{3}}+1, 2^{\frac{k-2}{3}})$ Empty node: $(1, 1, 2\cdot 2^{\frac{k-2}{3}})$	

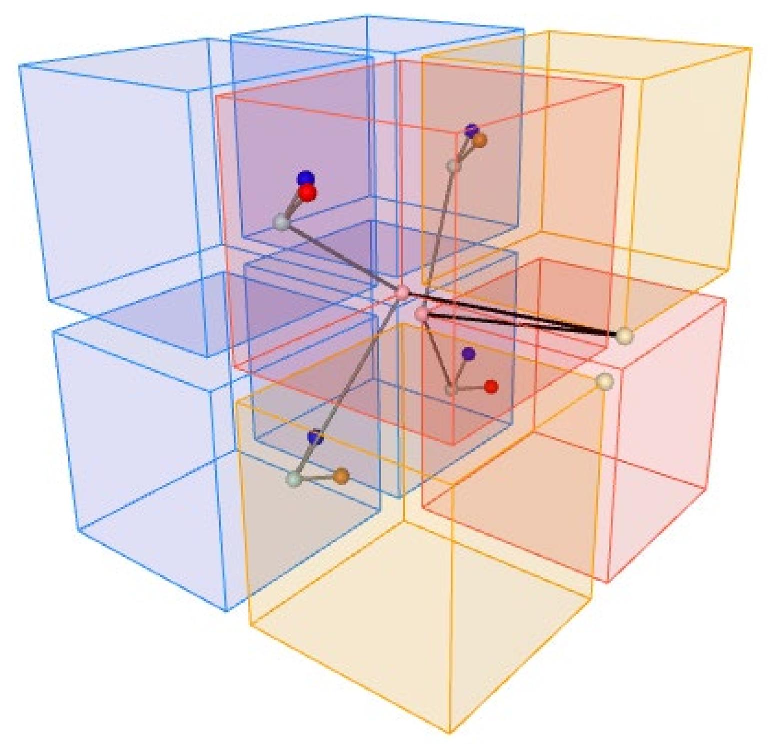
TRIVIAL BLOCKS



Each non-trivial block representing T_k can be recursively constructed from eight smaller sub-blocks representing T_{k-3}. Some sub-blocks need to be rotated around an axis.
The sub-blocks are connected by non-intersecting edges using the root and empty nodes to complete the drawing of T_k.

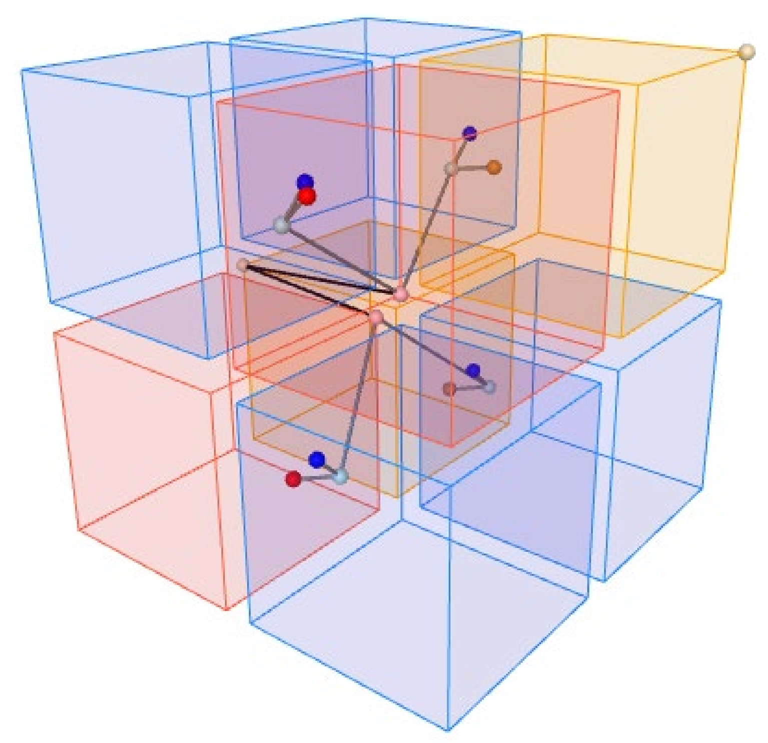
The eight sub-blocks are listed from bottom to top (-z to +z), from front to back (-y to +y), and from left to right (-x to +x).

F_k BLOCK



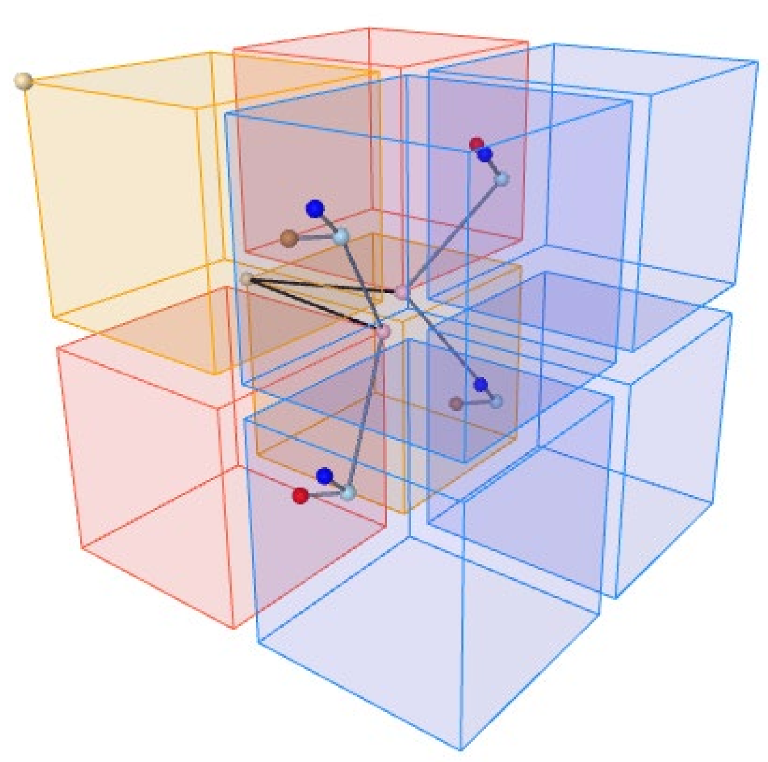
Sub-blocks: F_{k-3}, G_{k-3}, F_{k-3}, H_{k-3}, F_{k-3}, H_{k-3} (rot. $\pi\,|_x$), F_{k-3}, and G_{k-3} (rot. $\pi\,|_x$).

G_k BLOCK



Sub-blocks: H_{k-3} (rot. $\pi\,|_x$), F_{k-3} (rot. $\pi\,|_z$), G_{k-3} (rot. $\pi\,|_z$), F_{k-3} (rot. $\pi\,|_z$), F_{k-3}, H_{k-3} (rot. $\pi\,|_x$), F_{k-3}, and G_{k-3}.

H_k BLOCK



Sub-blocks: H_{k-3} (rot. $\pi\,|_z$), F_{k-3} (rot. $\pi\,|_z$), G_{k-3} (rot. $\pi\,|_z$), F_{k-3} (rot. $\pi\,|_z$), G_{k-3} (rot. $\pi\,|_z$), F_{k-3} (rot. $\pi\,|_x$), H_{k-3} (rot. $\pi\,|_x$ and rot. $\pi\,|_z$), and F_{k-3} (rot. $\pi\,|_z$).

References

[1] Akitaya H.A., Löffler M., Parada I. How to Fit a Tree in a Box. In: Proc. 26th International Symposium on Graph Drawing and Network Visualization (GD 2018). pp. 361–367. LNCS 11282, Springer (2018). https://doi.org/10.1007/978-3-030-04414-5 26

Scan the QR codes for interactive 3D drawings