Lossy compression of unordered rooted trees

Romain Azaïs, Jean-Baptiste Durand, Christophe Godin

▶ **To cite this version:**

Romain Azaïs, Jean-Baptiste Durand, Christophe Godin. Lossy compression of unordered rooted trees. DCC 2016 - Data Compression Conference, Mar 2016, Snowbird, Utah, United States. 10.1109/DCC.2016.73 . hal-01394707

HAL Id: hal-01394707
https://inria.hal.science/hal-01394707

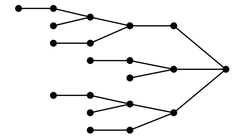
Submitted on 4 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

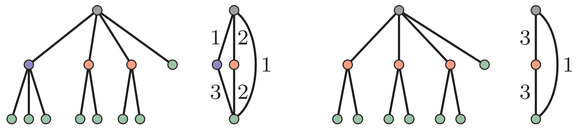
Lossy Compression of Unordered Trees

Romain Azaïs, Jean-Baptiste Durand and Christophe Godin



Objectives

DAG compression is a classical technique that consists in building a Directed Acyclic Graph (DAG) that represents an unordered tree without the redundancy of its subtrees.



Only trees with a high level of redundancy are efficiently compressed by this method. We introduce **a lossy compression method that consists in computing the DAG of a self-nested structure** that both presents the highest level of redundancy and **approximates the initial data to compress**.

Self-Nested Trees

Self-nested trees present remarkable compression properties because of the **systematic repetition of subtrees**. They are defined as trees such that all the subtrees of a given height are isomorphic. The DAG related to a self-nested tree τ is linear (there exists a path going through all its vertices) and has thus height$(\tau)+1$ nodes. In other words, **self-nested trees achieve maximum compression rates**. In addition, self-nested trees are very rare, while still being quite close to any unordered tree.

	$\deg \leq 2$	$\deg \leq 3$	$\deg \leq 4$
$h \leq 2$	0.88	$6.18.10^{-1}$	$3.52.10^{-1}$
$h \leq 3$	0.49	$3.38.10^{-2}$	$7.43.10^{-5}$
$h \leq 4$	0.07	$2.90.10^{-8}$	$4.16.10^{-23}$
$h \leq 5$	$3.36.10^{-4}$	$3.56.10^{-28}$	$1.66.10^{-100}$

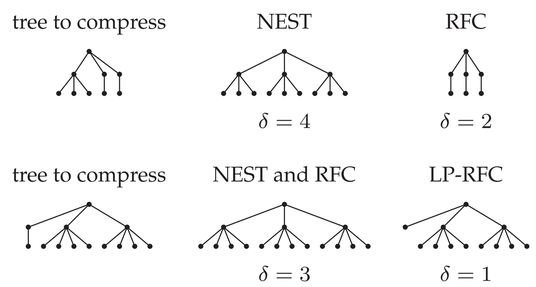
Table 1: Frequency of self-nested trees with given maximal height and ramification number with respect to unordered trees under the same constraint.

Edit Distance

This study requires to introduce a distance on the space of unordered trees in order to quantify the quality of a given self-nested estimate of a tree structure. We consider an editing distance defined from the following tree edit operations: **insertion and deletion of a leaf**. An edit script is a sequence of edit operations. The result of applying an edit script s to a tree τ is the tree τ^s obtained by applying the component edit operations to τ in the order they appear in the script. The cost of an edit script is only the number of edit operations. Finally, given two trees τ_1 and τ_2, the edit distance $\delta(\tau_1, \tau_2)$ is the **length of the minimum edit script** that transforms τ_1 into a tree that is isomorphic to τ_2.

Algorithm

The preexisting algorithm (NEST) only adds nodes to a tree τ to transform it into a self-nested structure. Instead of adding nodes, we propose to **replace some internal structures of τ by their centroid** (RFC). In particular, this allows us to delete some nodes and thus gives flexibility. These self-nested estimates may be computed in polynomial time. Nevertheless, this procedure can only modify subtrees and not delete them. That is why we introduce **a new algorithm that exploits local pruning** of τ (LP-RFC).



Example

The tree data to compress and its DAG version (top) and the solutions provided by the lossy compression algorithms NEST (middle) and LP-RFC (bottom). The NEST solution has too many nodes for looking like the initial tree (107 nodes added to obtain a self-nested tree). The LP-RFC solution is **visually close to the initial tree**, which is confirmed by the error rate of 35%. Both algorithms achieve a compression rate greater than 80%.

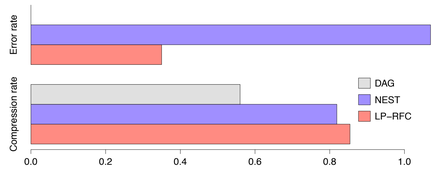
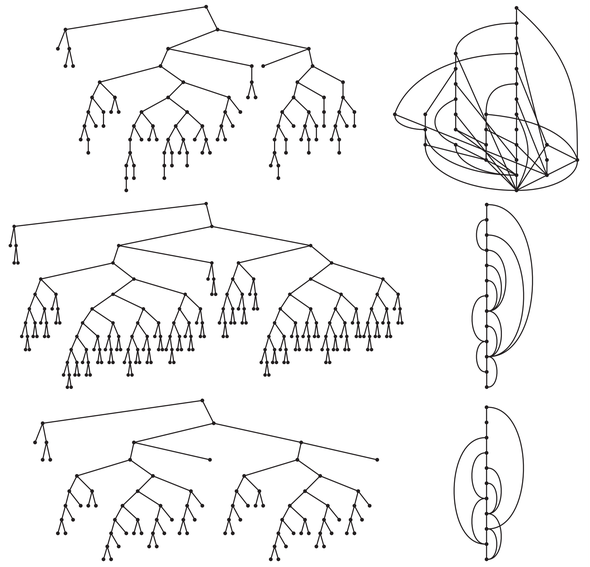


Figure 1: Error and compression rates on the example.



Simulation Study

Our simulations are performed on 500 small binary trees generated from a stochastic model. The three algorithms are equivalent in terms of compression rates. The key parameter is thus **the error rate that is much better for RFC and LP-RFC algorithms** than for NEST procedure (substantial gain of 20%). Local pruning is useful in RFC 24.2% of the time and makes the error decrease of 1.8% (7.3% when useful).

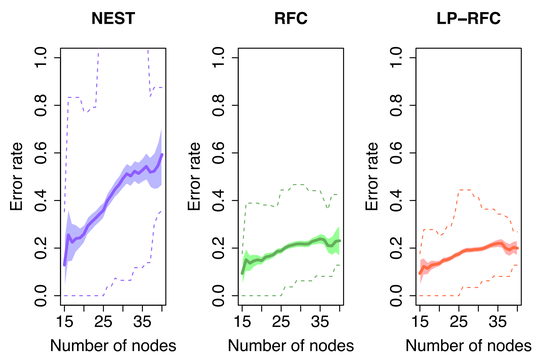


Figure 2: Error rates on the simulated dataset.

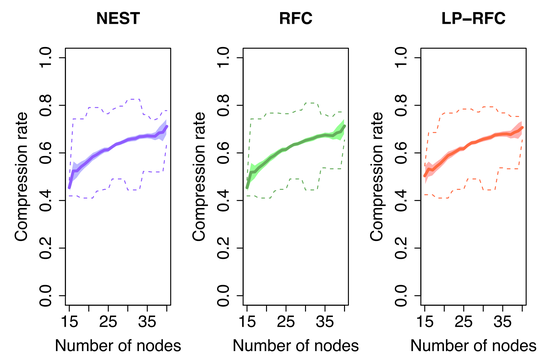


Figure 3: Compression rates on the simulated dataset.

Worst-Case Approximation Error

Among trees of height h and outdegree less than m, the tree that is the farthest to a self-nested tree may be identified. The editing distance to its best self-nested approximation is of order $0.25\, m^h$. The diameter of the space of unordered trees being of order m^h, it follows that **the largest area without any self-nested tree is of relative radius** 0.25. In addition, the error rate for this tree (and thus the worst error rate that must be expected from any lossy compression algorithm) is of order 0.33.

References

[1] Azaïs, Durand, and Godin. Lossy compression of trees via linear DAGs. *Preprint*, 2016.

[2] Godin and Ferraro. Quantifying the degree of self-nestedness of trees. Application to the structural analysis of plants. *IEEE TCBB*, 2010.

Contact Information

Romain Azaïs

Inria team BIGS – Institut Élie Cartan de Lorraine
Université de Lorraine in Nancy, France

romain.azais@inria.fr