Classification and Generation of Spanning Trees by Isomorphism

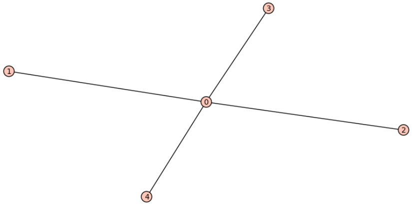
Drew Meier

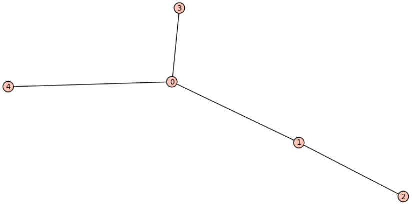
Department of Mathematics
Nebraska Wesleyan University

January 9, 2016

NEBRASKA
WESLEYAN
UNIVERSITY

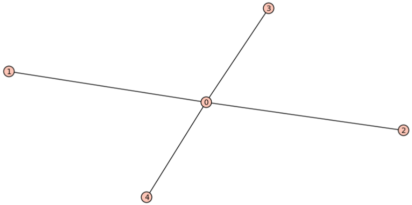
Two Spanning Trees

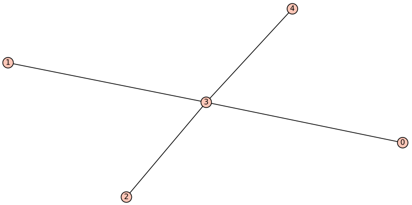




Isomorphisms

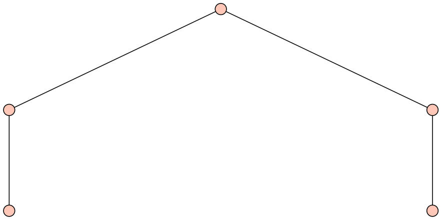
Two More Trees



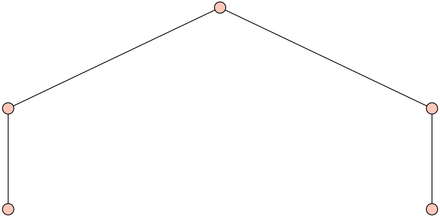


Rooting and Primary Representation of the Trees

New way to represent the tree

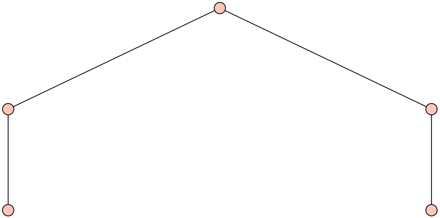


New way to represent the tree



[0

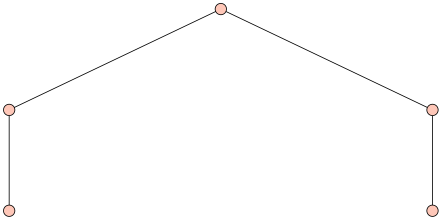
New way to represent the tree



$[0, 1$

Rooting and Primary Representation of the Trees

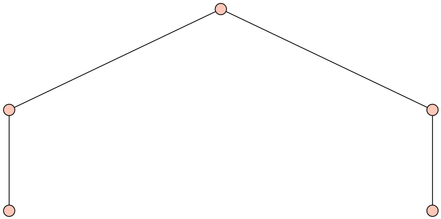
New way to represent the tree



$[0, 1, 2$

Rooting and Primary Representation of the Trees

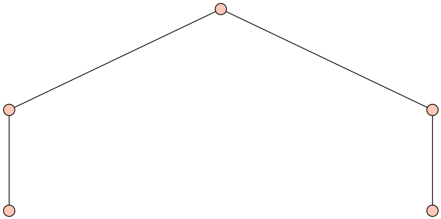
New way to represent the tree



$[0, 1, 2, 1$

Rooting and Primary Representation of the Trees

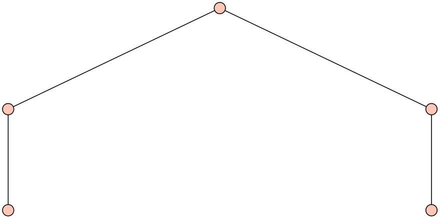
New way to represent the tree



$$[0, 1, 2, 1, 2]$$

Rooting and Primary Representation of the Trees

New way to represent the tree



$$[0, 1, 2, 1, 2]$$

- Goal is to find each lexicographically greatest integer sequence

The Succession Algorithm

- $L = [L_0, L_1, ..., L_{n-2}, L_{n-1}]$

The Succession Algorithm

- $L = [L_0, L_1, ..., L_{n-2}, L_{n-1}]$
- Set p to be the largest integer subscript such that $L_p \neq 1$

The Succession Algorithm

- $L = [L_0, L_1, ..., L_{n-2}, L_{n-1}]$
- Set p to be the largest integer subscript such that $L_p \neq 1$
- Set q to be the largest integer subscript such that $L_q = L_p - 1$

The Succession Algorithm

- $L = [L_0, L_1, ..., L_{n-2}, L_{n-1}]$
- Set p to be the largest integer subscript such that $L_p \neq 1$
- Set q to be the largest integer subscript such that $L_q = L_p - 1$
- The successor function $s(L)$ is given by

$$s_i = \begin{cases} L_i & 0 \leq i < p \\ L_{i-(p-q)} & p \leq i \leq n-1 \end{cases}$$

A Small Example

- $L = [0, 1, 2, 1, 2]$

A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$

A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$
- $p = 4$

A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$
- $p = 4$
- Find largest index q such that $L_q = L_p - 1$

A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$
- $p = 4$
- Find largest index q such that $L_q = L_p - 1$
- $q = 3$

A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$
- $p = 4$
- Find largest index q such that $L_q = L_p - 1$
- $q = 3$
- Apply successor function

A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$
- $p = 4$
- Find largest index q such that $L_q = L_p - 1$
- $q = 3$
- Apply successor function

$$s(L) = [0, 1, 2, 1$$

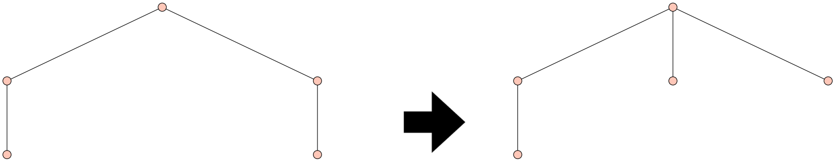
A Small Example

- $L = [0, 1, 2, 1, 2]$
- Find largest index p with $L_p \neq 1$
- $p = 4$
- Find largest index q such that $L_q = L_p - 1$
- $q = 3$
- Apply successor function

$$s(L) = [0, 1, 2, 1, 1]$$

Succession Algorithm and Failure Detection

Result



[0,1,2,1,2] → [0,1,2,1,1]

Succession Algorithm and Failure Detection

- There are cases where the succession algorithm fails
- $L = [0, 1, 2, 2..., 2]$
- Starting with the path guarantees there will always be a second 1 in the integer string
- If a different failure condition is met, a different set of operations on the string guarantee a new non-isomorphic tree to be generated

Burnside's Lemma

Lemma

The number of distinct labelings for a tree T on n vertices is given by

$$\frac{n!}{|Aut(T)|}$$

Cayley's Theorem

Theorem

The total number of labeled spanning trees in a complete graph on n vertices is given by

$$n^{n-2}$$

Burnside and Cayley

- Cayley's gives the total number of labeled trees on n vertices

Burnside and Cayley

- Cayley's gives the total number of labeled trees on *n* vertices
- Burnside's lemma gives the total number of labeled trees for each isomorphic class

Burnside and Cayley

- Cayley's gives the total number of labeled trees on n vertices
- Burnside's lemma gives the total number of labeled trees for each isomorphic class
- The sum of all labeled trees in each isomorphic class should yield the number given by Cayley's theorem

Example

http://austinmohr.com/home/?page_id=1422

Future Work

- Subgraph Isomorphism Problem
- Number of bipartite graphs in polynomial time

References

- McKay, Brendan D., et al. "Constant Time Generation of Free Trees." SIAM Journal on Computing 15 (1986): 540-48. Print.

Thanks for listening!