Ambiguity in a certain context-free grammar

Eric Rowland
joint work with Bobbe Cooper and Doron Zeilberger

Mathematics Department
Tulane University, New Orleans

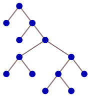
August 27, 2010

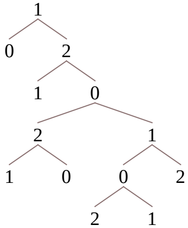
outline of the talk

the context-free grammar G

start symbols: $0, 1, 2$
formation rules: $0 \to 12$, $0 \to 21$, $1 \to 02$, $1 \to 20$, $2 \to 01$, $2 \to 10$

An n-leaf tree T parses a length-n word w on $\{0, 1, 2\}$ if T is a valid derivation tree for w under G.

For example, the tree  parses 0110212:



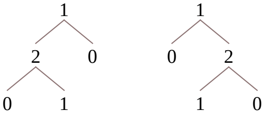
The set of possible derivation trees under G is the set of binary trees.

ambiguity

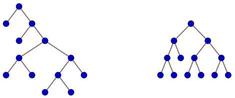
The grammar G is ambiguous;
there exist distinct trees that parse a common word.

The trees

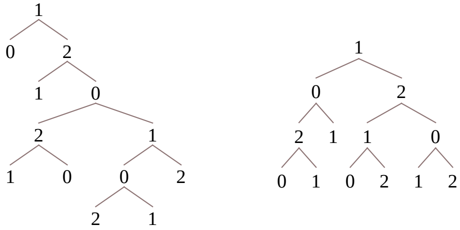
both parse 010:



The trees

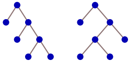


both parse 0110212:

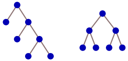


Theorem

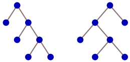
Let $n \geq 1$, and let T_1 and T_2 be n-leaf binary trees. Then T_1 and T_2 parse a common word under G.



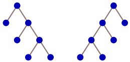
0010
0121



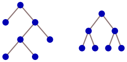
0112
0121



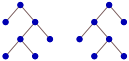
0010



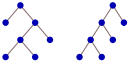
0112



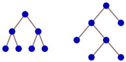
0121



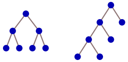
0010
0100



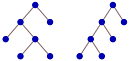
0100



0102



0102
0112



0100
0102

motivation

Theorem (Louis Kauffman, 1990)

The following are equivalent.

- *Every pair of n-leaf binary trees parses a common word under G.*
- *Every planar map is four-colorable.*

The four color theorem was proved in 1976 by Appel, Haken, and Koch and employed a case analysis carried out by machine.

Perhaps a word-theoretic proof of the four color theorem will be shorter than known proofs.

a word invariant

Proposition

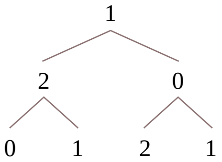
Let w be a word of length n on $\{0,1,2\}$ and T an n-leaf binary tree that parses w. Then for some permutation (i,j,k) of $(0,1,2)$,

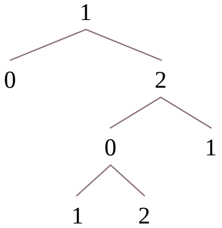
$$|w|_i \equiv |w|_j \not\equiv |w|_k \equiv |w| \mod 2.$$

Moreover, the root of T receives the label k when parsing w.

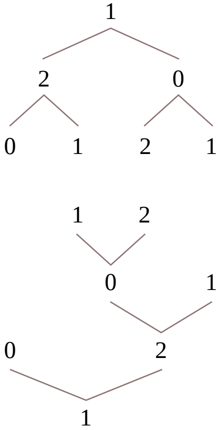
Proof. The congruence holds for words of length 1, and the formation rules $0 \to 12$, $0 \to 21$, $1 \to 02$, $1 \to 20$, $2 \to 01$, $2 \to 10$ preserve it because all four terms change parity with each rule application.

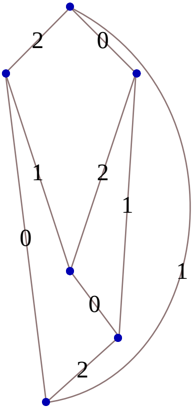
sketch of correspondence

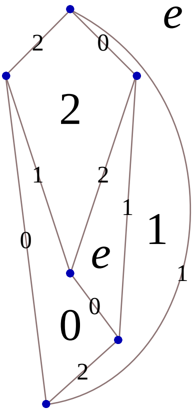




sketch of correspondence

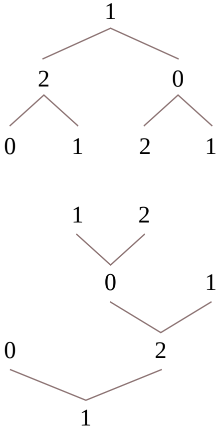


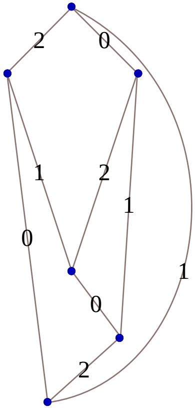


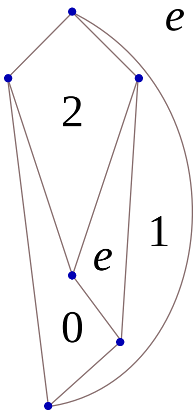


$\{e, 0, 1, 2\}$ is the Klein 4-group.

sketch of correspondence

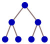
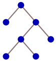






Let ParseWords(T_1, T_2) be the set of equivalence classes of words parsed by both trees T_1 and T_2.

We abuse notation slightly by writing a representative of each class.

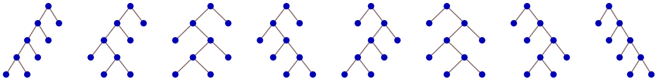
For example, ParseWords( , ) $= \{0121\}$.

The four color theorem is equivalent to the statement that for every pair of n-leaf binary trees T_1 and T_2 we have ParseWords(T_1, T_2) $\neq \{\}$.

outline of the talk

path trees

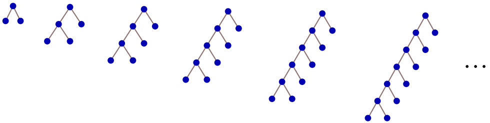
A path tree is a binary tree with at most two vertices on each level.



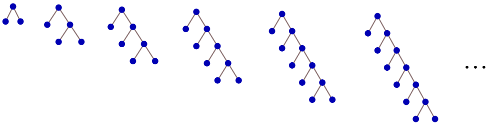
lll llr lrl lrr rll rlr rrl rrr

comb trees

Let LeftCombTree(n) be the n-leaf path tree corresponding to l^{n-2}.



Let RightCombTree(n) be the n-leaf path tree corresponding to r^{n-2}.



a pair of comb trees

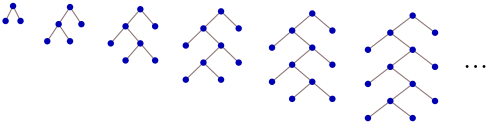
Theorem

ParseWords(LeftCombTree(n), RightCombTree(n)) $=$

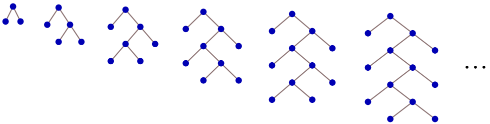
$$\begin{cases} \{01^{n-2}2\} & \textit{if } n \geq 2 \textit{ is even} \\ \{01^{n-2}0\} & \textit{if } n \geq 3 \textit{ is odd.} \end{cases}$$

crooked trees

Let LeftCrookedTree(n) be the path tree corresponding to $(lr)^{(n-2)/2}$.



Let RightCrookedTree(n) be the path tree corresponding to $(rl)^{(n-2)/2}$.

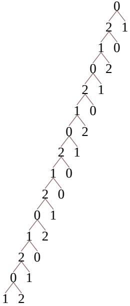


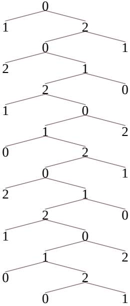
a comb tree and a crooked tree

Theorem

ParseWords(LeftCombTree(n), RightCrookedTree(n)) $=$

$$\begin{cases} \left\{ \operatorname{mod}(1-n,3)\,\left((012)^{n/6}\right)^R (012)^{(n-2)/6} \right\} & \text{if } n \geq 2 \text{ is even} \\ \left\{ \operatorname{mod}(1-n,3)\,\left((012)^{(n-3)/6}\right)^R (012)^{(n+1)/6} \right\} & \text{if } n \geq 3 \text{ is odd.} \end{cases}$$





a pair of crooked trees

The number of parse words is generally not constant.

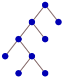
Theorem

For $n \geq 2$,

$$|\text{ParseWords}(\text{LeftCrookedTree}(n), \text{RightCrookedTree}(n))| = 2^{\lfloor n/2 \rfloor - 1}.$$

outline of the talk

Let $\mathrm{LeftTurnTree}(m, n)$ and $\mathrm{RightTurnTree}(m, n)$ be the $(m + n)$-leaf path trees corresponding to $l^m r^{n-2}$ and $r^m l^{n-2}$.

For example, $\mathrm{LeftTurnTree}(2, 3) =$ .

The next three theorems collectively enumerate parse words of $\mathrm{LeftTurnTree}(m, n)$ and $\mathrm{RightTurnTree}(k, m + n - k)$.

Theorem

For $m \geq 1$, $k \geq 1$, and $\max(2, k - m + 2) \leq n \leq k$,

$$|\mathrm{ParseWords}(\mathrm{LeftTurnTree}(m, n), \mathrm{RightTurnTree}(k, m + n - k))| = 1.$$

trees with one turn

Let

$$a(m,k) = |\text{ParseWords}(\text{LeftTurnTree}(m,k+1), \text{RightTurnTree}(k,m+1))|.$$

By symmetry, $a(m,k) = a(k,m)$.

Theorem

For $m \geq 1$, $k \geq 1$, and $n \geq k+2$,

$$|\text{ParseWords}(\text{LeftTurnTree}(m,n), \text{RightTurnTree}(k, m+n-k))| \\ = 2a(m,k).$$

recurrence satisfied by $a(m, k)$

Theorem

For $m \geq 1$ and $k \geq 1$,

$$a(m+3, k) - 2a(m+2, k) - a(m+1, k) + 2a(m, k) = 0.$$

No simple combinatorial proof known.

This recurrence can be written

$$(M-2)(M-1)(M+1)\, a(m, k) = 0.$$

For fixed k, the solution $a(m, k)$ is a linear combination of $2^m, 1, (-1)^m$.

main component of proof

Consider the labels of the internal vertices rather than of the leaves.

If the internal vertices of a path tree are labeled with v, the labeling can be extended to a parse word precisely when v is alternating — no two consecutive letters are equal.

Let A_m be the set of alternating words of the form $0v_2v_3\cdots v_{m-1}(1|2)$.
Let B_m be the set of alternating words of the form $0v_2v_3\cdots v_{m-1}(0|2)$.

Proposition

For $m \geq 2$, $|A_m| = (2^m + 2(-1)^m)/3$ and $|B_m| = (2^m - (-1)^m)/3$.

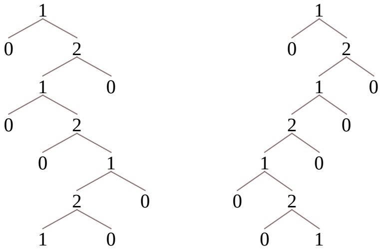
outline of the talk

trees sharing a bottom leaf

Proposition

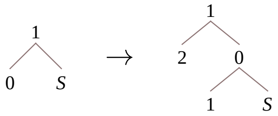
Let $n \geq 3$. If leaf i is a bottom leaf in two n-leaf path trees, then the trees both parse the word $0^{k-1}10^{n-k}$ for some $2 \leq k \leq n-1$.

Proof by example.

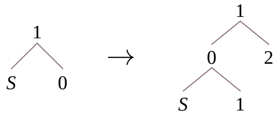


Suppose T'_1 and T'_2 are n-leaf binary trees parsing w. Fix $1 \leq i \leq n$. We may assume $w_i = 0$ and that the parent of leaf i in T'_2 receives 1.

Obtain T_2 by duplicating leaf i in T'_2 by performing the replacement



or



depending on whether leaf i is a left leaf or right leaf in T'_2.

Then T_2 parses the word obtained by replacing w_i by 21 or 12.

extending a pair of binary trees

Extend T_1' by attaching  to leaf i, obtaining T_1.
Clearly T_1 parses both of these words.

Moreover, every parse word of T_1 and T_2 arises uniquely in this way.

Proposition

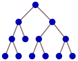
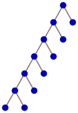
$$|\mathrm{ParseWords}(T_1, T_2)| = |\mathrm{ParseWords}(T_1', T_2')|.$$

In particular, T_1 and T_2 have a common parse word.

corollaries

Theorem

Let $n \geq 2$, and let T be an n-leaf binary tree. Let l be the level of leaf 1 in T. Then $|\text{ParseWords}(T, \text{LeftCombTree}(n))| = 2^{l-1}$.

For example, ParseWords( , ) $= \left\{ \begin{array}{l} 0100120, \\ 0102102, \\ 0111021, \\ 0112012 \end{array} \right\}$.

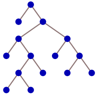
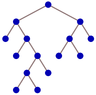
Theorem

Let $n \geq 4$. Let T_1 be an n-leaf binary tree and T_2 an n-leaf left turn tree. Then T_1 and T_2 have a common parse word.

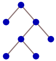
outline of the talk

decomposable pairs

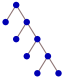
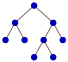
If two trees have a common branch system in the same position, we can decompose the pair into two smaller pairs. For example,

$T_1 =$  $T_2 =$ 

share the branch system

$S =$ ,

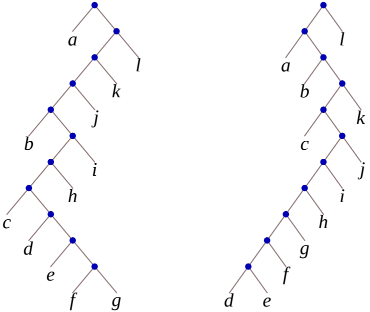
which we remove to obtain the 5-leaf trees

 .

We can find a parse word for the original pair from a parse word for this smaller pair and any valid labeling of S.

decomposable pairs — general trees

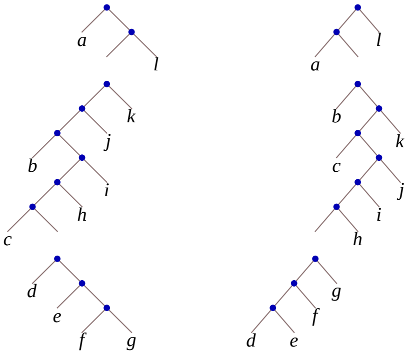
Consider the pair



More generally, we only require dangling subtrees S_1 and S_2 with the same set of leaves.

decomposable pairs — general trees

Breaking the trees at levels 2 and 8 as

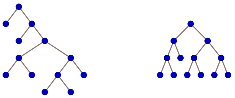


produces the same partition $\{\{a, l\}, \{b, c, h, i, j, k\}, \{d, e, f, g\}\}$ of the leaves in both trees.

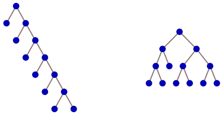
mutual crookedness

A pair of n-leaf trees T_1 and T_2 is mutually crooked if it cannot be obtained by duplicating some leaf i in a pair of $(n-1)$-leaf trees.

For example,



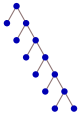
are mutually crooked, while the following are not.

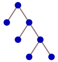


weak mutual crookedness

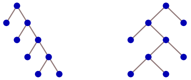
A pair of n-leaf trees T_1 and T_2 is weakly mutually crooked if it cannot be obtained by triplicating some leaf i in a pair of $(n-2)$-leaf trees.

Each tree in the pair



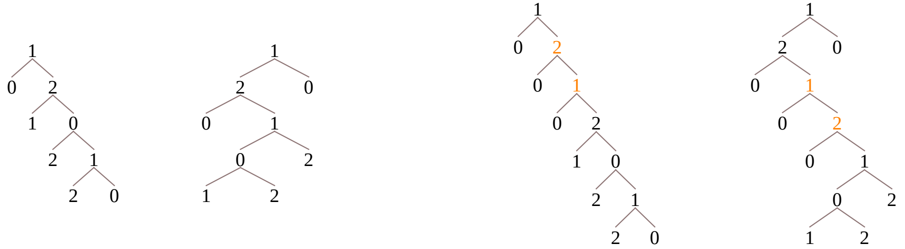
contains the right comb  in leaves 1–3.

Shortening this comb by two leaves produces the pair



which parses 01220.

We can re-insert the two leaves and obtain a parse word for the original pair by alternating the internal vertex labels.



Theorem

A pair of trees that is not weakly mutually crooked is reducible.

mutual crookedness

However, it appears that something stronger is true.

Conjecture

A pair of trees that is not mutually crooked is reducible.

The two consecutive leaves conjecturally receive the same label.
But there is no obvious relationship between the parse words.

summary

- The parse words of simple parameterized families can often be determined/enumerated.
- The number of parse words of LeftTurnTree(m, n) and RightTurnTree$(k, m + n - k)$ is given by a simple recurrence of order 3.
- To prove the "four color theorem for path trees" it suffices to consider indecomposable, weakly mutually crooked pairs of path trees that do not share a bottom leaf!