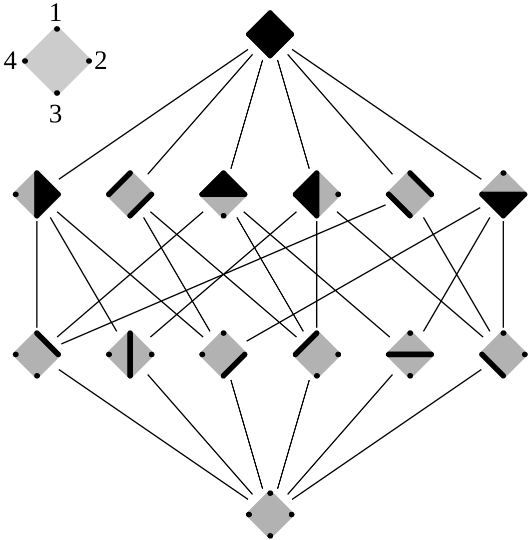
The algebra and geometry of sortable elements

Nathan Reading

NC State University

AMS Special Session on Algebraic Combinatorics
Fayetteville, AR, November 3–4, 2006

Noncrossing (NC) partitions (Kreweras, 1972)



Partitions of an n-cycle with noncrossing parts.

(Shown: $n = 4$, refinement order.)

NC partitions $\leftrightarrow$ certain elements of S_n.
Bijection: read parts clockwise as cycles.

W-NC partitions (Athanasiadis, Bessis, Brady, Reiner, Watt, $\sim$2000)

W: a finite Coxeter group with simple reflections S
Coxeter element: $c = s_1 \cdots s_{|S|}$ for $S = \{s_1, \ldots, s_{|S|}\}$
Factor c as a product of $|S|$ reflections $t_1 \cdots t_{|S|}$.
W-noncrossing partitions: elements of the form $t_1 \cdots t_i$
(as both $i \leq |S|$ and the factorization are allowed to vary).

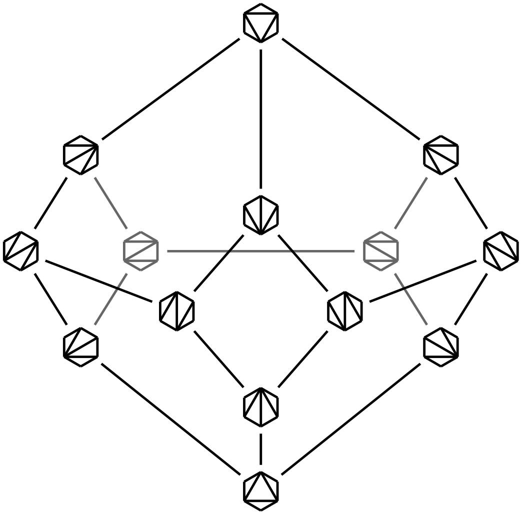
Example ($W = S_n$)

c is the product of the adjacent transpositions in any order.
This is always an n-cycle.
Reflections are (not-necessarily adjacent) transpositions.

Why do this?

1. Eilenberg-MacLane spaces (and more) for Artin groups (e.g. the braid group).
2. Interesting algebraic combinatorics.

Associahedron (Haiman, Lee, Milnor, Stasheff, 1963–1989)



Triangulations of a polygon. (Think: maximal collections of noncrossing diagonals.)

Edges connecting triangulations are "diagonal flips." This is a regular graph.

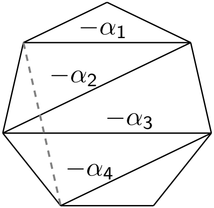
The associahedron is a simple convex polytope whose 1-skeleton is this graph.

Generalized associahedron (Fomin, Zelevinsky, 2003)

Positive roots ↔ reflections (in S_n: transpositions).
Simple roots ↔ simple reflections (in S_n: adjacent transp.).
Almost positive roots: Positive roots & negatives of simple roots.
Clusters: max'l sets of pairwise "compatible" almost positive roots.
Exchange graph: vertices are clusters, edges delete one root and replace it with a (unique) other root that makes a new cluster.

Example ($W = S_n$)

Simples: $\alpha_1, \ldots \alpha_{n-1}$, Positives: $\alpha_i + \cdots + \alpha_i$, $i \leq j$
Diagonals of $(n+2)$-gon ↔ "almost positive roots." $\binom{n+1}{2} - 1$.
"Compatible" = "noncrossing"
Exchanges = diagonal flips.



Why do this?

1. Cluster algebras of finite type.
2. Interesting polytopes.
3. Interesting algebraic combinatorics.

W-Catalan numbers (various researchers, 1980–present)

$$\mathrm{Cat}(W) := \prod_{i=1}^{n} \frac{e_i + h + 1}{e_i + 1}$$

Generalizes usual Catalan number ($W = S_n$).
h (Coxeter number) and e_i's (exponents) are fundamental numerical invariants of W.

$\mathrm{Cat}(W)$ counts:

1. W-noncrossing partitions
2. Clusters of almost positive roots for W
3. antichains in the root poset, positive regions in the Shi arrangement, B-stable ideals, conjugacy classes of elements of finite order in Lie groups.

W-Catalan numbers (various researchers, 1980–present)

$$\mathrm{Cat}(W) := \prod_{i=1}^{n} \frac{e_i + h + 1}{e_i + 1}$$

Generalizes usual Catalan number ($W = S_n$).
h (Coxeter number) and e_i's (exponents) are fundamental numerical invariants of W.

$\mathrm{Cat}(W)$ counts:

1. W-noncrossing partitions
2. Clusters of almost positive roots for W
3. antichains in the root poset, positive regions in the Shi arrangement, B-stable ideals, conjugacy classes of elements of finite order in Lie groups.
4. Sortable elements of W.

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ \ i{+}1)$ swaps the entries i and $i+1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4$, $\quad c = s_1 s_2 s_3$, $\quad \pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ i{+}1)$ swaps the entries i and $i + 1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4$, $c = s_1 s_2 s_3$, $\pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ i+1)$ swaps the entries i and $i+1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4, \quad c = s_1 s_2 s_3, \quad \pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132
2	s_2	$s_1 s_2$	4123

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ i{+}1)$ swaps the entries i and $i + 1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4$, $c = s_1 s_2 s_3$, $\pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132
2	s_2	$s_1 s_2$	4123
3	s_3	$s_1 s_2 s_3$	3124

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ i{+}1)$ swaps the entries i and $i+1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4, \quad c = s_1 s_2 s_3, \quad \pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132
2	s_2	$s_1 s_2$	4123
3	s_3	$s_1 s_2 s_3$	3124
4	s_1	$s_1 s_2 s_3 \vert$	3124

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i \; i{+}1)$ swaps the entries i and $i+1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4, \quad c = s_1 s_2 s_3, \quad \pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132
2	s_2	$s_1 s_2$	4123
3	s_3	$s_1 s_2 s_3$	3124
4	s_1	$s_1 s_2 s_3 \vert$	3124
5	s_2	$s_1 s_2 s_3 \vert s_2$	2134

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ i{+}1)$ swaps the entries i and $i+1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4, \quad c = s_1 s_2 s_3, \quad \pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132
2	s_2	$s_1 s_2$	4123
3	s_3	$s_1 s_2 s_3$	3124
4	s_1	$s_1 s_2 s_3 \vert$	3124
5	s_2	$s_1 s_2 s_3 \vert s_2$	2134
6	s_3	$s_1 s_2 s_3 \vert s_2$	2134

Sorting words in the example $W = S_n$

Multiplying a permutation π on the left by an adjacent transposition $s_i := (i\ i{+}1)$ swaps the entries i and $i+1$ in π.

Do this repeatedly, always putting entries into numerical order, and record the sequence of s_i's. Result: a reduced word for π.

Fix an order on the adjacent transpositions, and write a reduced word for π by trying the adjacent transpositions in that order, cyclically. Result: a sorting word for π. (C.f. "bubble sort.")

Example ($W = S_4$, $c = s_1 s_2 s_3$, $\pi = 4231$)

Step	s_i tried	Sorting word	Permutation
0			4231
1	s_1	s_1	4132
2	s_2	$s_1 s_2$	4123
3	s_3	$s_1 s_2 s_3$	3124
4	s_1	$s_1 s_2 s_3 \vert$	3124
5	s_2	$s_1 s_2 s_3 \vert s_2$	2134
6	s_3	$s_1 s_2 s_3 \vert s_2$	2134
7	s_1	$s_1 s_2 s_3 \vert s_2 \vert s_1$	1234

Sortable elements (R., 2005)

In general, write a c-sorting word for $w \in W$ by trying the generators cyclically according to some order on the simple reflections S. (This order also defines a Coxeter element c.) Place a divider "|" every time a pass through S is completed.

A c-sorting word can be interpreted as a sequence of sets (sets of letters between dividers "|").

If the sequence is nested then w is c-sortable.

Example ($\pi = 4231$ with c-sorting word $s_1 s_2 s_3 s_4 | s_2 | s_1$)

π is not c-sortable because $\{s_1\} \not\subseteq \{s_2\}$.

Example ($W = S_3, \quad c = s_1 s_2$)

c-sortable:	$1, \; s_1, \; s_1 s_2, \; s_1 s_2 | s_1, \; s_2$
not c-sortable:	$s_2	s_1$

Example ($W = S_n, \quad c = s_1 s_2 \cdots s_n$)

c-sortables $\leftrightarrow$ "231-avoiding" or "stack-sortable" permutations. (C.f. Björner and Wachs, 1997.) For another c, "312-avoiding."

Results on sortable elements

1. For finite W, any c, bijection to W-noncrossing partitions. $w \mapsto$ reflections associated to "descents." (R., 2005)
2. For finite W, any c, bijection to vertices of the generalized associahedron—i.e. clusters. (R., 2005)
3. Thus, bijective explanation of why clusters and noncrossing partitions are equinumerous. (A different explanation: Athanasiadis, Brady, McCammond, Watt, 2005–2006)
4. Deep connection to the lattice theory of the weak order on W, specifically Cambrian lattices. (R., 2005)
5. Sortable elements lead to Cambrian fans, a novel (combinatorial) construction of the generalized associahedron. (R., Speyer, 2006)

Coxeter fan, Cambrian fan, cluster fan

The Coxeter arrangement: the set of reflecting hyperplanes of reflections of W. The hyperplanes cut space into simplicial cones (the Coxeter fan $\mathcal{F}$). Elements of $W \leftrightarrow$ maximal cones of $\mathcal{F}$.

$\pi_\downarrow^c$: the unique longest c-sortable element below w.
(In S_n "length" = "number of inversions.")
Define $x \equiv_c y$ if $\pi_\downarrow^c(x) = \pi_\downarrow^c(y)$.

The Cambrian fan $\mathcal{F}_c$: Maximal cones are unions (over $\equiv_c$-classes) of maximal cones of the Coxeter fan. (Why is $\mathcal{F}_c$ a fan? Because $\equiv_c$ is a lattice congruence of the weak order.)

The cluster fan: Each cluster defines a maximal cone—the positive linear span of the roots in the cluster.

Combinatorial isomorphism

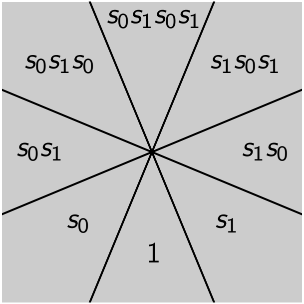
Theorem (R., Speyer, 2006)

The bijection c-sortables $\leftrightarrow$ clusters induces a combinatorial isomorphism between the Cambrian fan and the cluster fan.

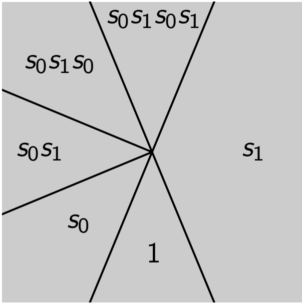
Consequences

1. Constructs the combinatorial backbone of cluster algebras of finite type in a new way. Some cluster algebra constructions are more natural in the Cambrian setting (e.g. "g-vectors").
2. Suggests a way to generalize the combinatorics of generalized associahedra to infinite Coxeter groups (work in progress with Speyer).

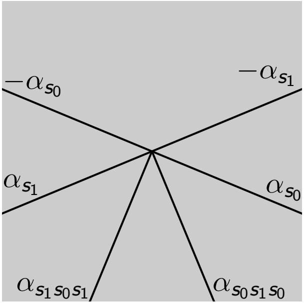
Example ($W = B_2$, $c = s_0 s_1$)



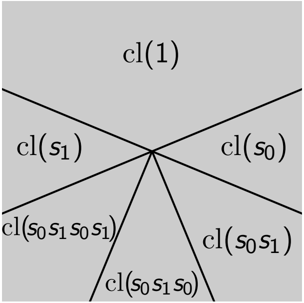
$\mathcal{F}$



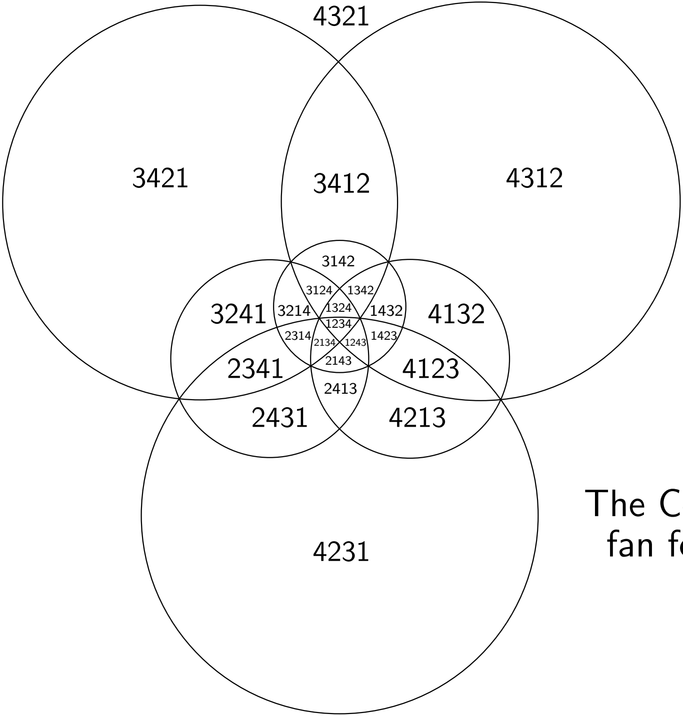
$\mathcal{F}_c$



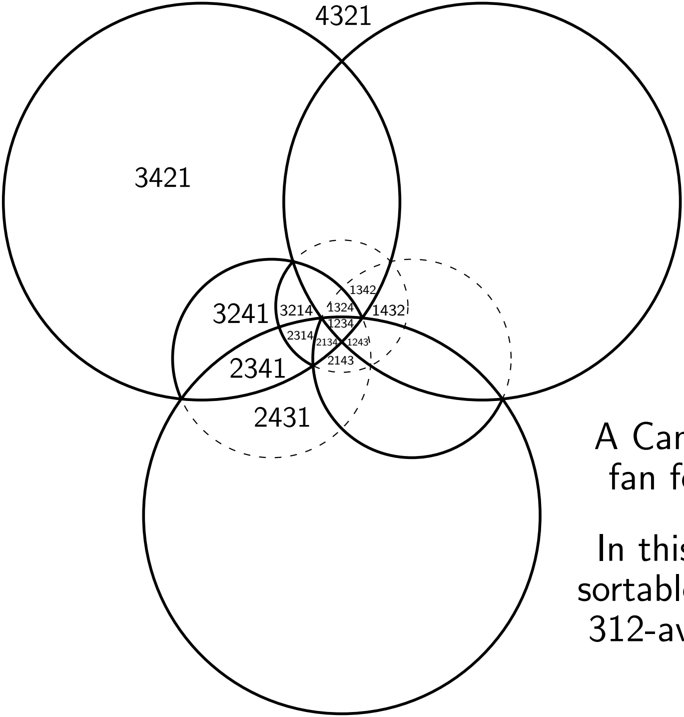
Cluster fan



The bijection



The Coxeter fan for S_4



A Cambrian fan for S_4

In this case, sortable means 312-avoiding.