Uncrowding the 5-Vertex Model: RSK and Crystal Structures

Lisa Johnston, Evuilynn Nguyen, Anne Schilling

University of California, Davis

UC DAVIS UNIVERSITY OF CALIFORNIA

Symmetric Grothendieck Polynomials

Symmetric Grothendieck polynomials are K-theoretic analogues of Schur functions. They are generating functions over set-valued tableaux:

$$\mathfrak{G}_\lambda(\mathbf{z};\beta) := \sum_{T\in \mathsf{SVT}^n(\lambda)} \beta^{\mathsf{ex}(T)}\mathbf{z}^{\mathsf{wt}(T)}$$

Here $\mathsf{ex}(T)$ counts the number of extra entries in multicells.

When $\beta = 0$, we recover the Schur polynomial $s_\lambda(\mathbf{z})$.

Example: Let $\lambda = (2,1)$ and $n = 2$,

$$\mathsf{SVT}^2(\lambda) = \left\{ \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 2 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 2 \\ \cline{1-1}\end{array} \quad \begin{array}{|c|c|}\hline 1 & 1,2 \\ \hline 2 \\ \cline{1-1}\end{array} \right\}$$

$$\mathfrak{G}_{(2,1)}(z_1,z_2;\beta) = z_1z_2^2 + z_1^2z_2 + \beta z_1^2z_2^2$$

Schur expansion and uncrowding

A set-valued tableau may have more than one entry in a cell. Buch's uncrowding algorithm removes an excess entry from a multicell and RSK inserts it into the next row.

$$\begin{array}{|c|c|}\hline 1 & 1,2 \\ \hline 2 \\ \cline{1-1}\end{array} \xrightarrow{\text{uncrowd}} \begin{array}{|c|c|}\hline 1 & 1 \\ \hline 2 & 2 \\ \hline\end{array}$$

For this example, uncrowding gives the Schur expansion

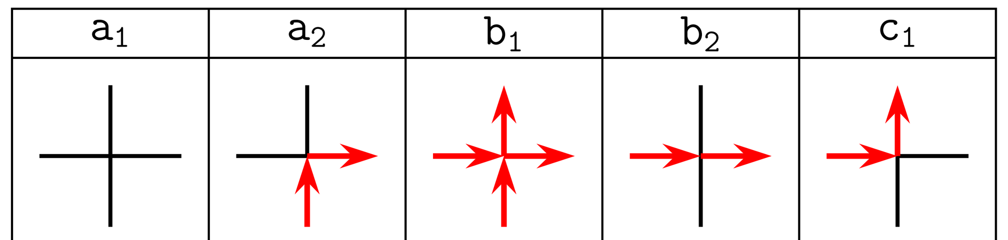
$$\mathfrak{G}_{(2,1)}(z_1,z_2;\beta) = s_{(2,1)}(z_1,z_2) + \beta s_{(2,2)}(z_1,z_2).$$

More generally, symmetric Grothendieck polynomials have a Schur expansion where the coefficient $C^\mu_\lambda \in \mathbb{Z}_{\geq 0}$ counts the possible "recording" tableaux obtained by uncrowding.

$$\mathfrak{G}_\lambda(\mathbf{z};\beta) = \sum_{\lambda\subseteq\mu} \beta^{|\mu|-|\lambda|}\, C^\mu_\lambda\, s_\mu(\mathbf{z})$$

The 5-vertex model

The Motegi and Sakai 5-vertex model realizes symmetric Grothendieck polynomials using admissible states. An admissible state is a configuration of arrows on the lattice using the five local vertex types shown below, with boundary conditions determined by λ.



a_1	a_2	b_1	b_2	c_1
1	$1+\beta z_i$	1	z_i	1

A decorated state is an admissible state together with a choice of either 1 or βz_i at each a_2 vertex.

The weight of a decorated state is computed by multiplying the local vertex weights. Since each a_2 (bump) vertex contributes a factor of $1+\beta z_i$, distributing the product of local weights amounts to choosing either 1 or βz_i from each a_2 factor:

× = choose 1 (trivial bump), ○ = choose βz_i (non-trivial bump)

A filled circle ● marks a b_2 vertex of weight z_i.

Tableaux and lattices

There is a weight-preserving bijection between decorated states and set-valued tableaux,

$$\psi : \mathfrak{D}_\lambda \longrightarrow \mathsf{SVT}^n(\lambda).$$

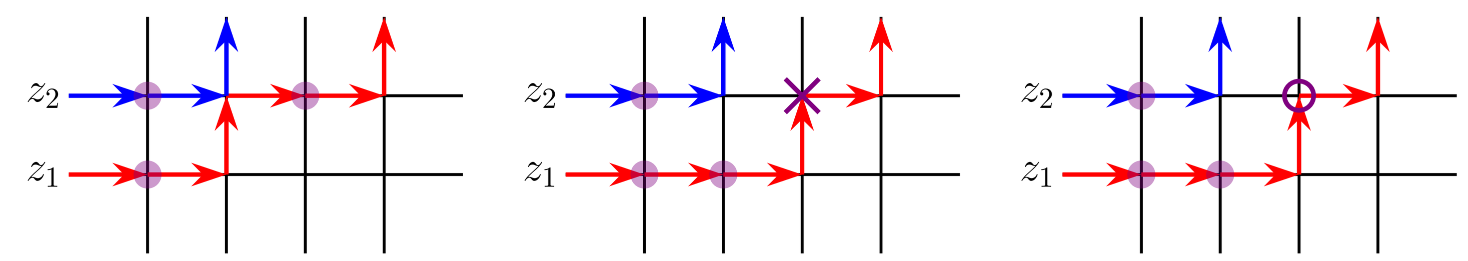
Consequently, the partition function of the 5-vertex model realizes the symmetric Grothendieck polynomials:

$$Z(\mathfrak{D}_\lambda;\mathbf{z};\beta) = \mathfrak{G}_\lambda(\mathbf{z};\beta).$$

We number lattice rows from bottom to top. The path entering from the left in lattice row i records row i of the tableau:

$$\text{path entering row } i \longleftrightarrow \text{tableau row } i.$$

For $\lambda = (2,1)$ and $n = 2$, the decorated states below map under ψ to the three set-valued tableaux in the first column.



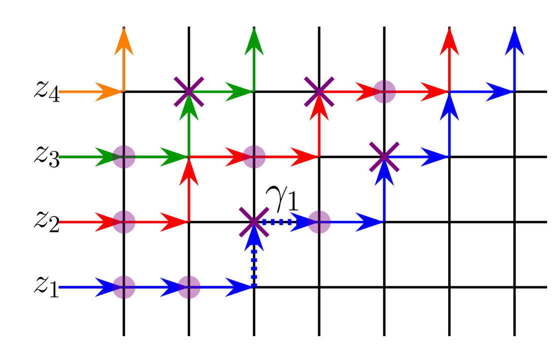
RSK and lattice insertion

RSK insertion can be performed directly on a decorated state by shifting a path segment until no non-trivial bump remains. To insert a letter u:

1. Locate the c_1 vertex in lattice row u on the first path.
2. Shift the path segment from this vertex to the next b_2 vertex or top arrow one column to the right.
3. Repeat on the next path whenever a new non-trivial bump appears.

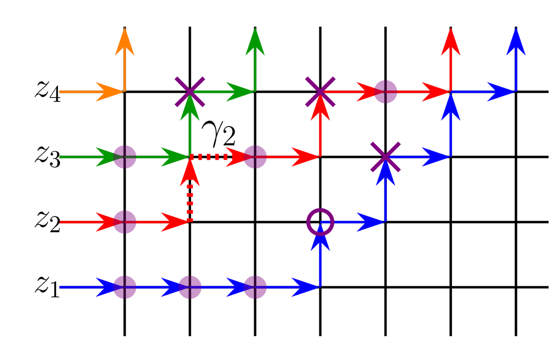
Below we insert the letter 1; the dotted segments show the shifted path segments γ_i.

①



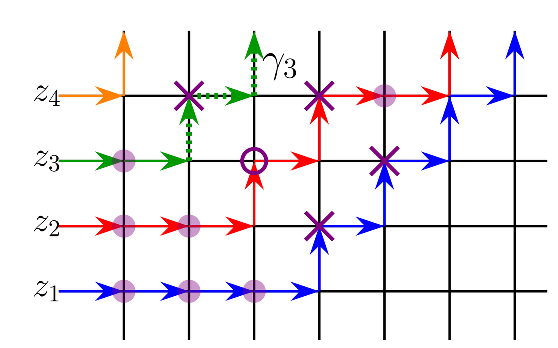
$$\begin{array}{|c|c|c|}\hline 1 & 1 & 2 \\ \hline 2 & 3 & 4 \\ \hline 3 \\ \cline{1-1}\end{array}$$

②



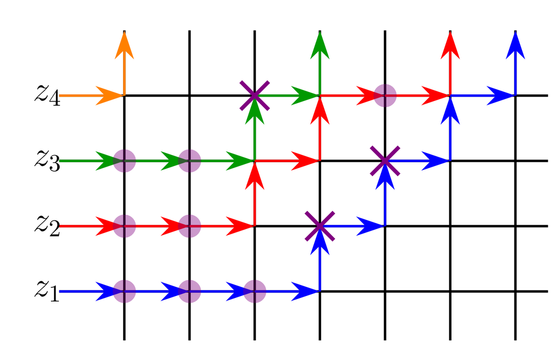
$$\begin{array}{|c|c|c|}\hline 1 & 1 & 1,2 \\ \hline 2 & 3 & 4 \\ \hline 3 \\ \cline{1-1}\end{array}$$

③



$$\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 2 & 2,3 & 4 \\ \hline 3 \\ \cline{1-1}\end{array}$$

④



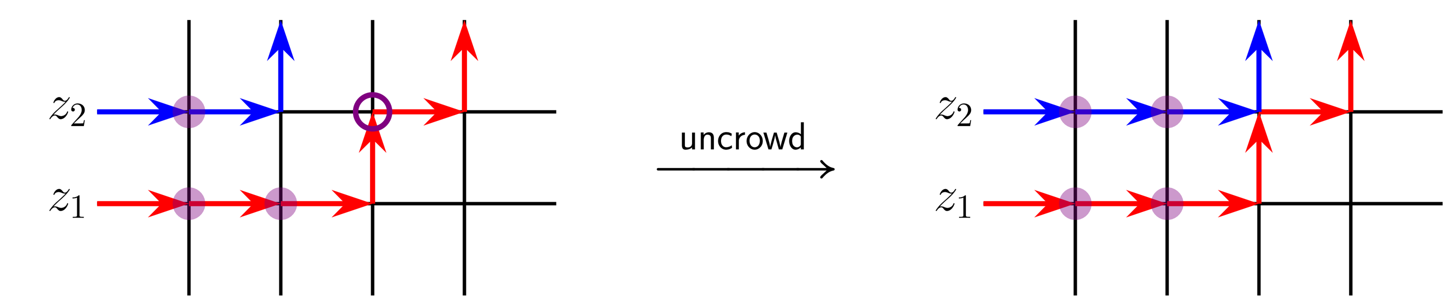
$$\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 2 & 2 & 4 \\ \hline 3 & 3 \\ \cline{1-2}\end{array}$$

Lattice uncrowding

A non-trivial bump is the lattice version of a crowded entry in a set-valued tableau. Lattice uncrowding removes these bumps by repeatedly applying lattice insertion moves.

The algorithm processes paths from top to bottom:

1. Find the next non-trivial bump on the current path.
2. Replace it by a trivial bump.
3. Insert into the path above using the lattice insertion move.
4. Repeat until no non-trivial bumps remain.



Under the bijection ψ, this agrees with Buch's uncrowding algorithm on set-valued tableaux.

Crystal operators on the lattice

Crystal operators can also be defined directly on decorated states. For each i, the f_i operator only looks at two adjacent rows i and $i+1$.

1. Place brackets on certain vertices in rows i and $i+1$.
2. Pair brackets using a lattice version of the signature rule.
3. Shift one path segment at the unpaired bracket to define the f_i operator.

This is the lattice analogue of the usual crystal operator on set-valued tableaux.

Main takeaway

The bijection ψ intertwines the tableau and lattice algorithms: applying ψ before or after the operation gives the same result.

This holds for:

(1) RSK insertion (2) uncrowding (3) crystal operators

tableau algorithms	$\Longleftrightarrow$	local moves on decorated states

Thus these combinatorial structures can be studied directly in the 5-vertex model.

References

[1] A. S. Buch, A Littlewood–Richardson rule for the K-theory of Grassmannians.

[2] L. Johnston, E. Nguyen, and A. Schilling, Uncrowding the 5-vertex model: RSK and crystal structures, arXiv preprint, submitted.

[3] A. Lascoux and M.-P. Schützenberger, Structure de Hopf de l'anneau de Grothendieck.

[4] K. Motegi and K. Sakai, Vertex models, TASEP, and Grothendieck polynomials.