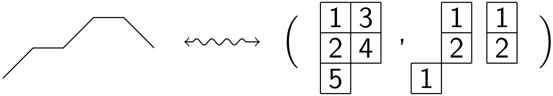
A Sundaram type bijection for $\mathrm{SO}(2k+1)$:

vacillating tableaux and pairs consisting of a standard Young tableau and an orthogonal Littlewood-Richardson tableau

Judith Jagenteufel

TECHNISCHE UNIVERSITÄT WIEN
Vienna | Austria

31st Conference on Formal Power Series and Algebraic Combinatorics (FPSAC), Ljubljana, July 2nd, 2019



Classical Schur-Weyl duality

$$V^{\otimes r} \cong \bigoplus_{\substack{\lambda \vdash r \\ \ell(\lambda) \leq n}} V^{\mathrm{GL}}(\lambda) \otimes S(\lambda)$$

as $\mathrm{GL}(V) \times \mathfrak{S}_r$ representations, $V = \mathbb{C}^n$.

- $\mathrm{GL}(V)$ acts diagonally and $\mathfrak{S}_r$ permutes tensor positions
- $V^{\mathrm{GL}}(\lambda)$, $S(\lambda)$ ... irreducible representation of $\mathrm{GL}(V)$, $\mathfrak{S}_r$

Robinson-Schensted

$$\{1, \ldots, n\}^r \leftrightarrow \bigcup_{\lambda} \Big(\mathsf{SSYT}(\lambda),\ \mathsf{SYT}(\lambda)\Big)$$

- $\mathsf{SSYT}(\lambda)$, $\mathsf{SYT}(\lambda)$ ... (semi)standard Young tableaux

Special orthogonal group

branching rule

$$V^{\mathrm{GL}}(\lambda)\downarrow^{\mathrm{GL}(V)}_{\mathrm{SO}(V)} \cong \bigoplus_{\mu} \mathrm{c}^{\mu}_{\lambda} V^{\mathrm{SO}}(\mu)$$

where $\mathrm{c}^{\mu}_{\lambda}$ are multiplicities counted by orthogonal LR tableaux

leads to

$$V^{\otimes r} \cong \bigoplus_{\substack{\mu \text{ a partition} \\ l(\mu)\leq n \\ \mu'_1+\mu'_2\leq n}} V^{\mathrm{SO}}(\mu)\otimes \bigoplus_{\substack{\lambda\vdash r \\ l(\lambda)\leq n}} \mathrm{c}^{\mu}_{\lambda}\, S(\lambda) = \bigoplus_{\substack{\mu \text{ a partition} \\ l(\mu)\leq n \\ \mu'_1+\mu'_2\leq n}} V^{\mathrm{SO}}(\mu)\otimes U(r,\mu)$$

as $\mathrm{SO}(n) \times \mathfrak{S}_r$ representations. $n = 2k+1$ thus n odd

- $V = \mathbb{C}^n$ ... vector representation of $\mathrm{SO}(n)$
- $V^{\mathrm{SO}}(\mu)$, $S(\lambda)$ ... irreducible representations of $\mathrm{SO}(n)$ and $\mathfrak{S}_r$

Our setting

$$U(r,\mu) = \Big(\bigoplus_{\lambda} \mathrm{c}^{\mu}_{\lambda}\, S(\lambda)\Big)$$

Our main result: a bijection between

vacillating tableaux $\leftrightarrow$ $\big($orthogonal LRT, SYT$\big)$

that preserves descents.

Standard Young tableaux

1	2	5	6	9
3	4	7	8	12
10	13			
11	16			
14				
15				
17				

Standard Young tableaux of shape λ ($\mathrm{SYT}(\lambda)$):

fillings of a Young diagram of shape λ with entries $\{1, 2, \ldots, |\lambda|\}$, increasing in rows and columns

Descents

d is a descent if $d + 1$ is in a row below *d*

1	2	5	6	9
3	4	7	8	12
10	13			
11	16			
14				
15				
17				

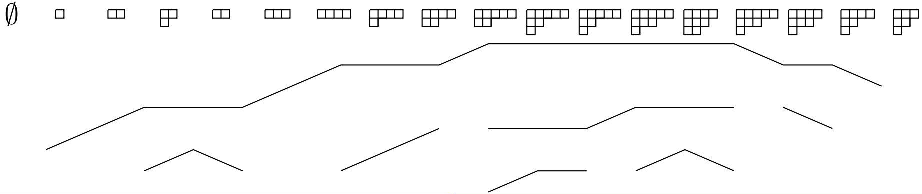
descent set $\{2, 6, 9, 10, 12, 13, 14, 16\}$

Vacillating tableaux - highest weight words

- *Vacillating tableau*: a sequence of partitions / Young diagrams $\emptyset = \lambda^0, \lambda^1, \ldots, \lambda^r$, at most k rows, *shape* λ^r
  - λ^i and λ^{i+1} differ in at most one cell
  - $\lambda^i = \lambda^{i+1}$ only if k^{th} row is non-empty
- *Highest weight word*: word w with letters in $\{0, \pm 1, \ldots, \pm k\}$, length r, *weight* $(\#1 - \#(-1), \ldots, \#k - \#(-k))$, such that for every prefix $w_1, \ldots, w_j$:
  - $\#i - \#(-i) \geq 0$
  - $\#i - \#(-i) \geq \#(i+1) - \#(-i-1)$
  - If the last position, $w_j = 0$ then $\#k - \#(-k) > 0$.

Example ($k = 3$)

$$(1,\ 1,\ 2,\ -2,\ 1,\ 1,\ 2,\ 2,\ 1,\ 3,\ 0,\ 2,\ 3,\ -3,\ -1,\ -2,\ -1)$$

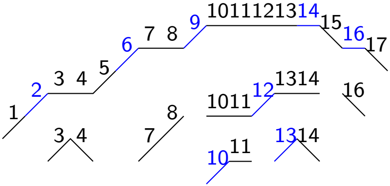


Descents of vacillating tableaux

Vacillating tableaux

a position i of w is a descent if there exists a path from w_i to w_{i+1} in the crystal graph:
$1 \to 2 \to \cdots \to k \to 0 \to -k \to \cdots \to -1$
and $w_i w_{i+1} \neq j(-j)$ if $\#j - \#(-j) = 0$ in $w_1, \ldots, w_{i-1}$.

Example



descent set $\{2, 6, 9, 10, 12, 13, 14, 16\}$

Quasi symmetric expansion

Recall the Frobenius character:

$$\operatorname{ch}\rho = \frac{1}{r!}\sum_{\pi\in\mathfrak{S}_r} \operatorname{Tr}\rho(\pi) p_{\lambda(\pi)}$$

where p_λ is a power sum symmetric function. We have:

$$\operatorname{ch} S(\lambda) = s_\lambda = \sum_{Q\in SYT(\lambda)} F_{\mathrm{Des}(Q)}$$

where s_λ is a Schur function and F_D is a fundamental quasi-symmetric function:

$$F_D = \sum_{\substack{i_1\leq i_2\leq\dots\leq i_r\\ j\in D\Rightarrow i_j<i_{j+1}}} x_{i_1}x_{i_2}\dots x_{i_r}.$$

Therefore, as our bijection is descent preserving, we obtain:

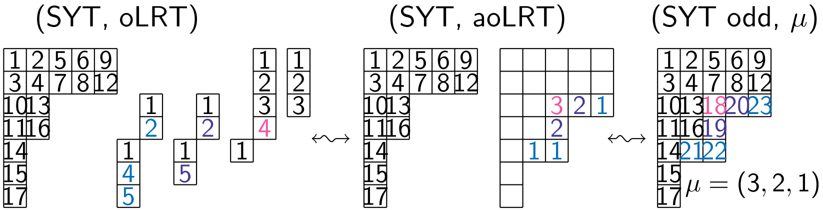
$$\operatorname{ch}\Big(\bigoplus_{\substack{\lambda\vdash r\\ l(\lambda)\leq n}} \mathrm{c}_\lambda^\mu\, S(\lambda)\Big) = \sum_{\substack{w \text{ vacillating tableau}\\ \text{of length } r \text{ and shape } \mu}} F_{\mathrm{Des}(w)}.$$

Main result: a bijection between

vacillating tableaux $\leftrightarrow$ $\big($orthogonal LRT, SYT$\big)$

that preserves descents.

Strategy

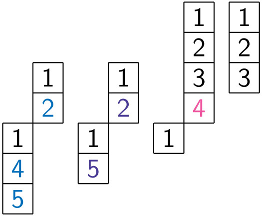


Orthogonal Littlewood-Richardson tableaux

$n = 2k + 1$, $\ell(\lambda) \leq n$, $\ell(\mu) \leq k$, $\mu \leq \lambda$
e.g. $n = 7$, $k = 3$, $\lambda = (5, 5, 2, 2, 1, 1, 1)$, $\mu = (3, 2, 1)$

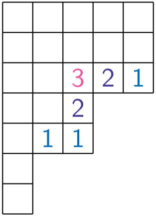
Kwon's LR tableaux

- k twocolumn skew-shape semistandard tableau with tail μ_i; one single column
- λ' determines the filling
- several conditions on size and filling

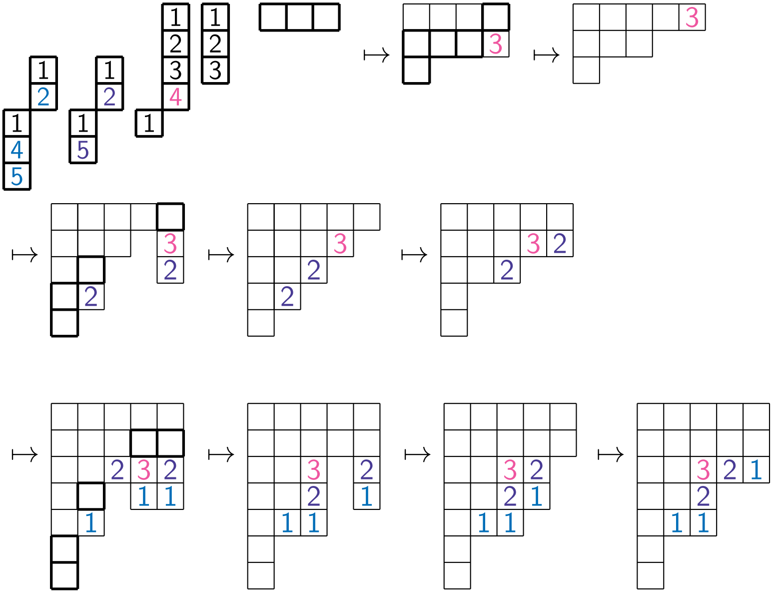


alternative LR tableaux

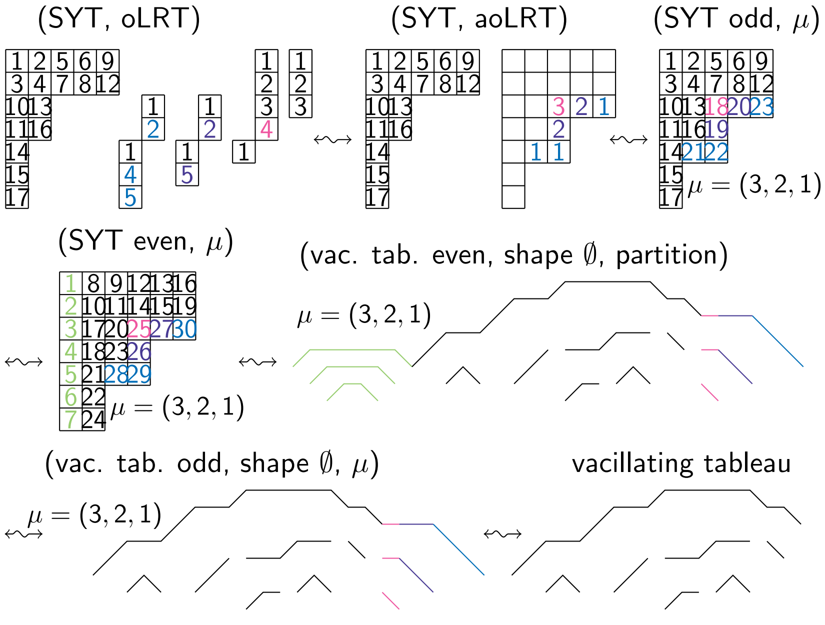
- reverse skew-shape semistandard tableaux, inner shape λ
- μ determines the filling, reading word is Yamanouchi
- technical condition



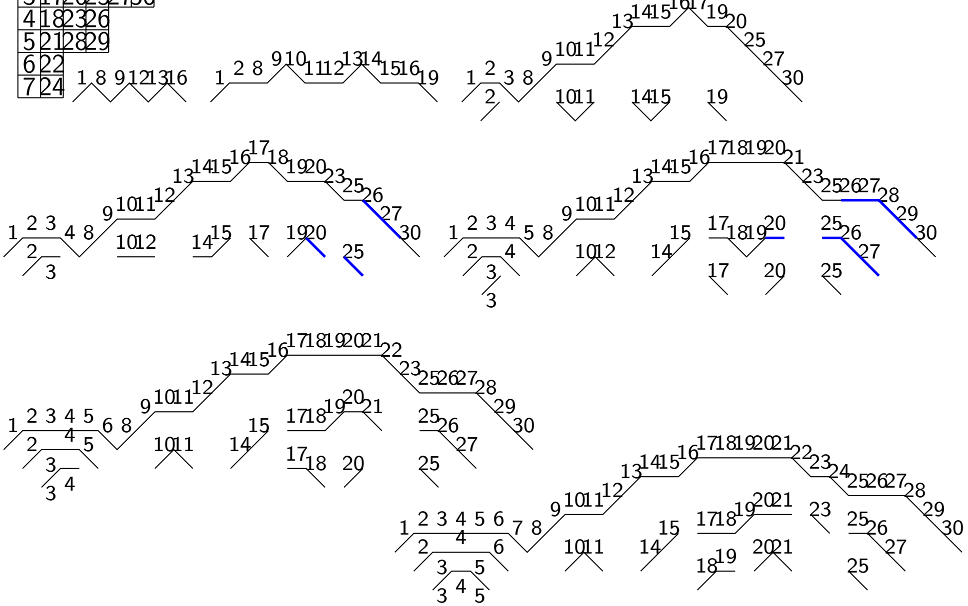
First part: manipulate the orthogonal LR tableaux



Strategy

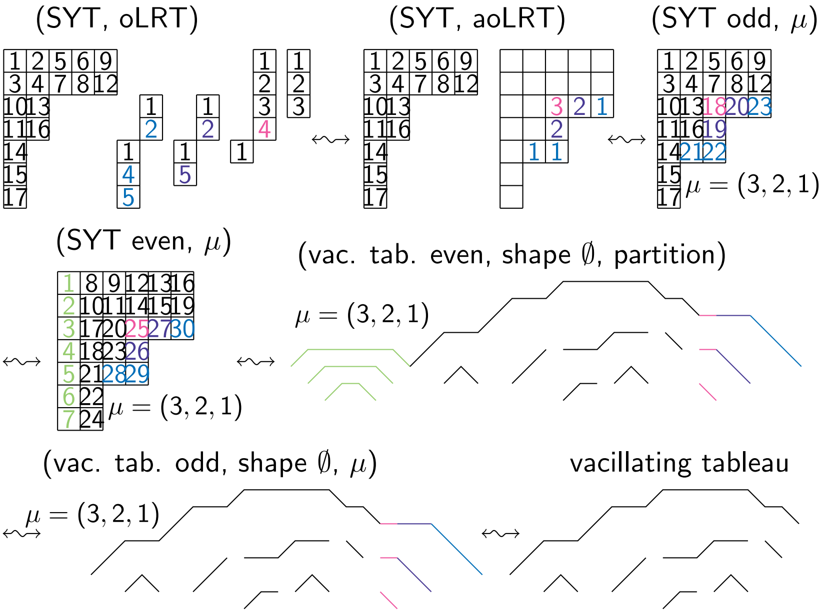

(SYT, oLRT)
(SYT, aoLRT)
(SYT odd, μ)
$\mu = (3, 2, 1)$
(SYT even, μ)
$\mu = (3, 2, 1)$
(vac. tab. even, shape $\emptyset$, partition)
$\mu = (3, 2, 1)$
(vac. tab. odd, shape $\emptyset$, μ)
$\mu = (3, 2, 1)$
vacillating tableau

1	8	9	12	13	16
2	10	11	14	15	19
3	17	20	25	27	30
4	18	23	26		
5	21	28	29		
6	22				
7	24				

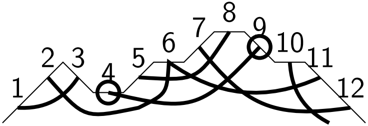


let w be word $(1, -1, \ldots, 1, -1)$ labeled by first row elements of Q; /* insert row 1 */
for $i = 2, 3, \ldots, n$ **do** /* insert row i */
$j := \lfloor i/2 \rfloor$; unmark everything **if** *i even* **then** change 0-entries of w into $j, -j, \ldots, j, -j$; /* init. j */
for *pairs of elements a, b in row i, start with the rightmost, go to left* **do**
$a_1 := a$, $b_1 := b$, $a_l := b_l := 0$ for $l = 2, 3, \ldots, j+1$, **if** *b is largest pos.* **then** insert b_1 with -1;
let p be rightmost pos. so far, $\tilde{p}$ be next pos. left of p with $w(\tilde{p}) \in \{0, \pm j\}$; /* b */
while $a_{j+1} < p$ or $w(p) \notin \{0, \pm j\}$ **do**
if *$p < b_l$, $p \neq a_l$, $w(p) = -l$ for an $l < j$, $a_{l+1} = 0$* **then**
if *p not marked, $b_{l+1} = 0$* **then** $w(p) := -l - 1$, $b_{l+1} := p$; /* b_{l+1} */
else if $p < a_l$, $p < b_{l+1}$ **then** $w(p) := -l - 1$, $a_{l+1} := p$; /* a_{l+1} */
if *i is even, $w(p) \in \{0, \pm j\}$* **then** /* i even */
if $b_j < p$, $w(\tilde{p}), w(p) = j, -j$ **then** for $l < j$ change $\pm l$ on l-level 0 between p and $\tilde{p}$ into $\pm(l+1)$, if $p < b_l$, $b_{l+1} = 0$ ignore b_l, if $p < a_l$, $a_{l+1} = 0$ ignore a_l; mark changed pos.; change $-j, j$ between p and $\tilde{p}$ into $0, 0$; /* adj. SP */
else if $a_j < p$, $w(\tilde{p}), w(p) = j, -j$ **then** $w(\tilde{p}), w(p) := 0, 0$; for $l < j$ mark $\pm l$ on l-level 0 between p and $\tilde{p}$, if $p < a_l$, $a_{l+1} = 0$ ignore a_l; /* mark it + connect */
else if *$p = a_j$, $w(\tilde{p}) = 0$ on j-level* 1 **then** $w(\tilde{p}), w(p) := j, 0$, $a_{j+1} := \tilde{p}$; /* $a_{j+1}1$ */
else if $p < a_j$, $w(p) = -j$, $a_{j+1} = 0$ **then** $w(p) := j$, $a_{j+1} := p$; /* $a_{j+1}2$ */
if $p < b_j$, $b_{j+1} = 0$ **then** $b_{j+1} := p$; /* b_{j+1} */
if *i is odd, $w(p) \in \{0, \pm j\}$* **then** /* i odd */
if *$b_{j+1} < p$, $w(p), w(\tilde{p}) = 0, 0$, p j-even position on j-level* 1 *if $b_j < p$ or* 2 *if $p < b_j$* **then** for $l < j$ change $\pm l$ on l-level 0 between p and $\tilde{p}$ into $\pm(l+1)$, if $p < b_l$, $b_{l+1} = 0$ ignore b_l, if $p < a_l$, $a_{l+1} = 0$ ignore a_l; mark changed pos.; /* adj. SP */
else if *$a_{j+1} < p < b_{j+1}$, $w(p) = j$ on j-level* 1 *for $p < a_j$ or* 0 *for $a_j < p$* **then**
$w(\tilde{p}), w(p) := 0, 0$; /* connect */
else if *$a_{j+1} < p < b_{j+1}$, $w(\tilde{p}), w(p) = 0, 0$, p j-even position on j-level* 2 *if $p < a_j$ or* 1 *if $a_j < p$* **then** $w(\tilde{p}), w(p) := -j, j$; for $l < j$ mark $\pm l$ on l-level 0 between p and $\tilde{p}$, if $p < a_l$, $a_{l+1} = 0$ ignore a_l; /* mark it + separate */
else if $p < b_j$, $p \neq a_j$, $w(p) = -j$, $a_{j+1} = 0$ **then**
if *p not marked, $b_{j+1} = 0$* **then** $w(p) := 0$, $b_{j+1} := p$; /* b_{j+1} */
else if $p < a_j$, $p < b_{j+1}$ **then** $w(p) := 0$, $a_{j+1} := p$; /* a_{j+1} */
if *$p = a_l$ on l-level* 0*, for an $l < j$, the l to the right is marked* **then** mark a_l;
if *p height V. in l for an $l < j$, ($p < a_l$ or p not marked), if $p < a_l$, $a_{l+1} = 0$ ignore a_l* **then**
$w(p) := l + 1$ **if** $a_{l+1} = 0$ **then** $b_{l+1} := 0$ **else** $a_{l+1} := 0$; /* height V. */
if *i is even*, $a_{j+1} \neq 0$ **then** $w(a_{j+1}), w(p) := 0, 0$, $a_{j+1} := 0$;
if *i is odd, $w(\tilde{p}) = 0$ on j-level* 0 **then** $w(\tilde{p}) := -j$, $b_{j+1} := 0$;
if *b is between p and the position to the left* **then** insert b_1 with -1; /* b */
else if *a is between those* **then** insert a_1 with -1; /* a */
let p be one position to the left in w, change $\tilde{p}$ according to it;
do one additional iteration of the inner for-loop with $a = b = 0$;
forget the labels of w, set $V = w$ and **return** V;

Strategy


(SYT, oLRT)
(SYT, aoLRT)
(SYT odd, μ)
$\mu = (3,2,1)$
(SYT even, μ)
$\mu = (3,2,1)$
(vac. tab. even, shape $\emptyset$, partition)
$\mu = (3,2,1)$
(vac. tab. odd, shape $\emptyset$, μ)
$\mu = (3,2,1)$
vacillating tableau

An alternative algorithm for $n = 3$



$\rightsquigarrow$

1	2	4	5	7	10
3	6	8	12		
9	11				

Provides further properties, only conjectured for general odd n

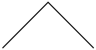
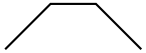
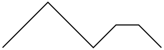
- Concatenation
- Insertion algorithm

Conjectures

Conjecture (proven for $n = 3$, tableaux of even length)

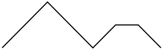
Concatenation of two standard Young tableaux with same-parity row lengths corresponds to concatenation of the associated paths.

Concatenating $\begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & 4 \\ \hline\end{array}$ with $\begin{array}{|c|}\hline 1 \\ \hline 2 \\ \hline 3 \\ \hline\end{array}$ yields $\begin{array}{|c|c|c|}\hline 1 & 2 & 5 \\ \hline 3 & 4 & 6 \\ \hline 7 \\ \cline{1-1}\end{array}$.

Concatenating  with  yields .

Conjecture

Reversal of the vacillating tableau corresponds to evacuation of the standard Young tableau.

Reversal of  yields .

Evacuation of $\begin{array}{|c|c|c|}\hline 1 & 2 & 5 \\ \hline 3 & 4 & 6 \\ \hline 7 \\ \cline{1-1}\end{array}$ yields $\begin{array}{|c|c|c|}\hline 1 & 4 & 5 \\ \hline 2 & 6 & 7 \\ \hline 3 \\ \cline{1-1}\end{array}$.

Thank you!