\$\hat{\mathfrak{sl}}_n\$ crystals and cylindric partitions$^1$

Peter Tingley

Massachusetts Institute of Technology

Brown, March 1, 2011

$^1$Slides and notes available at www-math.mit.edu/\~{}ptingley/
Outline

1 Motivation and background
   - Crystals, Characters and Combinatorics
   - $\widehat{\mathfrak{sl}}_n$ and its crystals

2 Partiton and cylindric partition models
   - The Misra-Miwa-Hayashi realization
   - The multi-partition realization
   - Cylindric partitions
   - Two applications
   - Relationship with the Kyoto path model

3 Current work
   - Fayers’ crystals
   - Future directions
Example: $\mathfrak{sl}_3$

$\mathfrak{sl}_3$ is the Lie algebra consisting of $3 \times 3$ matrices with trace 0.

The Lie bracket is given by $[A, B] = AB - BA$.

The standard generators are:

- $E_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$,
- $F_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$,
- $E_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$,
- $F_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$.

Any representation of $\mathfrak{sl}_3$ decomposes (as a vector space) into the direct sum of the simultaneous eigenspaces for the diagonal matrices (weight spaces).
Example: $\mathfrak{sl}_3$

- $\mathfrak{sl}_3$ is the Lie algebra consisting of $3 \times 3$ matrices with trace 0.
Example: $\mathfrak{sl}_3$

- $\mathfrak{sl}_3$ is the Lie algebra consisting of $3 \times 3$ matrices with trace 0.
- The Lie bracket is given by $[A, B] = AB - BA$. 

Brown, March 1, 2011 3 / 15
**Example: \( \mathfrak{sl}_3 \)**

- \( \mathfrak{sl}_3 \) is the Lie algebra consisting of \( 3 \times 3 \) matrices with trace 0.
- The Lie bracket is given by \([A, B] = AB - BA\).
- The standard generators are:

\[
E_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},
\]

\[
E_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.
\]
Example: $\mathfrak{sl}_3$

- $\mathfrak{sl}_3$ is the Lie algebra consisting of $3 \times 3$ matrices with trace 0.
- The Lie bracket is given by $[A, B] = AB - BA$.
- The standard generators are:

  $E_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, \quad $F_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, \\
  $E_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$, \quad $F_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$.

- Any representation of $\mathfrak{sl}_3$ decomposes (as a vector space) into the direct sum of the simultaneous eigenspaces for the diagonal matrices (weight spaces).
The adjoint representation of \( \mathfrak{sl}_3 \)
The adjoint representation of $\mathfrak{sl}_3$

There are 6 one dimensional weight spaces and 1 two dimensional weight space.
The adjoint representation of $\mathfrak{sl}_3$

- There are 6 one dimensional weight spaces and 1 two dimensional weight space.

![Diagram showing the adjoint representation of $\mathfrak{sl}_3$ with 6 one dimensional weight spaces and 1 two dimensional weight space.]
The adjoint representation of $\mathfrak{sl}_3$

- There are 6 one dimensional weight spaces and 1 two dimensional weight space.
- The generators $F_1$ and $F_2$ act between weight spaces.
The adjoint representation of $\mathfrak{sl}_3$

- There are 6 one dimensional weight spaces and 1 two dimensional weight space.
- The generators $F_1$ and $F_2$ act between weight spaces.
The adjoint representation of $\mathfrak{sl}_3$

- There are 6 one dimensional weight spaces and 1 two dimensional weight space.
- The generators $F_1$ and $F_2$ act between weight spaces.
- There are 4 distinguished one dimensional spaces in the middle.
The adjoint representation of $\mathfrak{sl}_3$

- There are 6 one dimensional weight spaces and 1 two dimensional weight space.
- The generators $F_1$ and $F_2$ act between weight spaces.
- There are 4 distinguished one dimensional spaces in the middle.
- If we use $U_q(\mathfrak{sl}_3)$ and ‘rescale’ the operators, then “at $q = 0$”, they match up.
There are 6 one dimensional weight spaces and 1 two dimensional weight space.

The generators $F_1$ and $F_2$ act between weight spaces.

There are 4 distinguished one dimensional spaces in the middle.

If we use $U_q(\mathfrak{sl}_3)$ and ‘rescale’ the operators, then “at $q = 0$", they match up.
The adjoint representation of $\mathfrak{sl}_3$

- There are 6 one dimensional weight spaces and 1 two dimensional weight space.
- The generators $F_1$ and $F_2$ act between weight spaces.
- There are 4 distinguished one dimensional spaces in the middle.
- If we use $U_q(\mathfrak{sl}_3)$ and ‘rescale’ the operators, then “at $q = 0$”, they match up. You get a colored directed graph.
The adjoint representation of $\mathfrak{sl}_3$

- Often the vertices of the crystal graph can be parametrized by combinatorial objects.
The adjoint representation of $\mathfrak{sl}_3$

- Often the vertices of the crystal graph can be parametrized by combinatorial objects.
Often the vertices of the crystal graph can be parametrized by combinatorial objects.

Then the combinatorics gives information about representation theory, and vise-versa.
The adjoint representation of $\mathfrak{sl}_3$

- Often the vertices of the crystal graph can be parametrized by combinatorial objects.
- Then the combinatorics gives information about representation theory, and vice-versa.
- Here you see that the graded dimension of the representation is the generating function for semi-standard Young tableau.
The infinity crystal
There is a crystal $B_\lambda$ for each dominant integral weight $\lambda$. 
The infinity crystal

- There is a crystal $B_\lambda$ for each dominant integral weight $\lambda$.
- \( \{B_\lambda\} \) forms a directed system.
There is a crystal $B_\lambda$ for each dominant integral weight $\lambda$.

$\{B_\lambda\}$ forms a directed system.
There is a crystal $B_\lambda$ for each dominant integral weight $\lambda$.

{ $B_\lambda$ } forms a directed system.
There is a crystal $B_\lambda$ for each dominant integral weight $\lambda$. 

$\{B_\lambda\}$ forms a directed system. 

The limit of this system is $B_\infty$. 
The infinity crystal

\[ \bigcup_{\omega_1} B_{\omega_1} + 2 \omega_2 \]

In any realization, I want to understand these injections. They come from the fact that there is a canonical basis of $U_q(g)$ which descends to a basis of each $V(\lambda)$. 

Peter Tingley (MIT)

Brown, March 1, 2011 5 / 15
The infinity crystal
The infinity crystal

In any realization, I want to understand these injections. They come from the fact that there is a canonical basis of $\mathcal{U}_{\hat{\mathfrak{g}}}$ which descends to a basis of each $V(\lambda)$.
In any realization, I want to understand these injections.
In any realization, I want to understand these injections.
They come from the fact that there is a canonical basis of $U_q^-(\mathfrak{g})$ which descends to a basis of each $V(\lambda)$.
\( \hat{\mathfrak{sl}}_n \) and its crystals

### Definition 1:
\( \hat{\mathfrak{sl}}'_n \) is a central extension of the Lie algebra of polynomial loops in \( \mathfrak{sl}_n \) \( \otimes \mathbb{C}[t, t^{-1}] \oplus \mathbb{C} \) where \( \mathbb{C} \) is central.

### Definition 2:
\( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with Dynkin diagram

\[ \begin{array}{c}
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\cdot \\
\cdot \end{array} \]

\( \hat{\mathfrak{sl}}'_n \) is generated by \( \{ E_i, F_i \} \) \( 0 \leq i \leq n-1 \) subject to the relations that for each pair \( 0 \leq i < j \leq n-1 \), \( \{ E_i, F_i, E_j, F_j \} \) generate a copy of \( \{ \mathfrak{sl}_3 \) if \( |i - j| = 1 \mod (n) \) \( \mathfrak{sl}_2 \times \mathfrak{sl}_2 \) otherwise.

For \( \hat{\mathfrak{sl}}_4 \):
\[ E_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad E_0 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \]
Definition 1:

\[ \hat{\mathfrak{sl}}_n \text{ is a central extension of the Lie algebra of polynomial loops in } \mathfrak{sl}_n \text{ } \hat{\mathfrak{sl}}_n \oplus \mathbb{C}[t, t^{-1}] \text{ where } \mathbb{C} \text{ is central.} \]

Definition 2:

\[ \hat{\mathfrak{sl}}_n \text{ (for } n \geq 3) \text{ is the Kac-Moody algebra with Dynkin diagram } \begin{array}{ccccccc} u & u & u & u & u & u & \ldots \end{array} \]

\[ \hat{\mathfrak{sl}}_n \text{ is generated by } \{\mathfrak{e}_i, \mathfrak{f}_i\}_{0 \leq i \leq n-1} \text{ subject to the relations that for each pair } 0 \leq i < j \leq n-1, \{\mathfrak{e}_i, \mathfrak{f}_i, \mathfrak{e}_j, \mathfrak{f}_j\} \text{ generate a copy of } \begin{cases} \mathfrak{sl}_3 & \text{if } |i-j| = 1 \text{ mod } (n) \\ \mathfrak{sl}_2 \times \mathfrak{sl}_2 & \text{otherwise.} \end{cases} \]

For \( \hat{\mathfrak{sl}}_4 \):

\[ \mathfrak{e}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathfrak{e}_0 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ t & 0 & 0 & 0 \end{pmatrix} \]
\( \hat{\mathfrak{sl}}_n \) and its crystals

- Definition 1: \( \hat{\mathfrak{sl}}_n \) is a central extension of the Lie algebra of polynomial loops in \( \mathfrak{sl}_n \)
Definition 1: $\hat{\mathfrak{sl}}_n$ is a central extension of the Lie algebra of polynomial loops in $\mathfrak{sl}_n$

$$\mathfrak{sl}_n \otimes \mathbb{C}[t, t^{-1}] \oplus \mathbb{C}C$$
Definition 1: $\widehat{\mathfrak{sl}}_n'$ is a central extension of the Lie algebra of polynomial loops in $\mathfrak{sl}_n$

\[ \mathfrak{sl}_n \otimes \mathbb{C}[t, t^{-1}] \oplus \mathbb{C} \]

where
Definition 1: $\hat{\mathfrak{sl}}_n'$ is a central extension of the Lie algebra of polynomial loops in $\mathfrak{sl}_n$

$$\mathfrak{sl}_n \otimes \mathbb{C}[t, t^{-1}] \oplus \mathbb{C}C$$

where

- $C$ is central.
Definition 1: $\hat{\mathfrak{sl}}'_n$ is a central extension of the Lie algebra of polynomial loops in $\mathfrak{sl}_n$

$$\mathfrak{sl}_n \otimes \mathbb{C}[t, t^{-1}] \oplus \mathbb{C}C$$

where

1. $C$ is central.
2. $[X \otimes t^a, Y \otimes t^b] = [X, Y] \otimes f(t)g(t) + tr(ad(X)ad(Y))\delta_{a+b,0}C$. 
**Definition 1:**
\( \hat{sl}_n \) is a central extension of the Lie algebra of polynomial loops in \( sl_n \) \( \otimes \mathbb{C} [t, t^{-1}] \oplus \mathbb{C} \delta_{ab} \).

**Definition 2:**
\( \hat{sl}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with Dynkin diagram

- \( \hat{sl}_n \) is generated by \( \{ E_i, F_i \} \) \( 0 \leq i \leq n-1 \) subject to the relations that for each pair \( 0 \leq i < j \leq n-1 \), \( \{ E_i, F_i, E_j, F_j \} \) generate a copy of \( \{ sl_3 \) if \( |i - j| \equiv 1 \pmod{n} \) \( sl_2 \times sl_2 \) otherwise.

For \( \hat{sl}_4 \):

\[
E_2 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix},
\]

\[
E_0 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
t_0 & 0 & 0 & 0
\end{pmatrix}
\]
Definition 1: \( \hat{\mathfrak{sl}}'_n \) is a central extension of the Lie algebra of polynomial loops in \( \mathfrak{sl}_n \otimes \mathbb{C}[t, t^{-1}] \oplus \mathbb{C} \) where \( 1 \) is central.

Definition 2: \( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with Dynkin diagram \( \cdots \). \( \hat{\mathfrak{sl}}'_n \) is generated by \( \{ E_i, F_i \} \) for \( 0 \leq i \leq n-1 \) subject to the relations that for each pair \( 0 \leq i < j \leq n-1 \), \( \{ E_i, F_i, E_j, F_j \} \) generate a copy of \( \mathfrak{sl}_3 \) if \( |i - j| = 1 \mod (n) \), \( \mathfrak{sl}_2 \times \mathfrak{sl}_2 \) otherwise.

For \( \hat{\mathfrak{sl}}_4 \):
\[
E_2 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix},
E_0 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
t & 0 & 0 & 0
\end{pmatrix}.
\]
Motivation and background

\( \hat{\mathfrak{sl}}_n \) and its crystals

Definition 1: \( \hat{\mathfrak{sl}}_n' \) is a central extension of the Lie algebra of polynomial loops in \( \mathfrak{sl}_n \) \( \otimes \mathbb{C} \left[ [t], [t^{-1}] \right] \oplus \mathbb{C} \) where \( \mathbb{C} \) is central.

Definition 2: \( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with dynkin diagram

\[ \begin{array}{c}
  \cdots \\
  \vdots \\
  \bullet - \bullet \\
  \bullet - \bullet \\
  \bullet - \bullet \\
 \end{array} \]

Fix \( n \geq 3 \). An (infinite) \( n \)-colored directed graph is an \( \hat{\mathfrak{sl}}_n \) crystal if, for each pair of colors \( c_i \) and \( c_j \), the graph consisting of all edges of those 2 colors is

\begin{align*}
\text{An } \mathfrak{sl}_3 \text{ crystal graph if } |i - j| = 1 \mod (n) \\
\text{An } \mathfrak{sl}_2 \times \mathfrak{sl}_2 \text{ crystal graph otherwise.}
\end{align*}
\(\hat{\mathfrak{sl}}_n\) and its crystals

- **Definition 2:** \(\hat{\mathfrak{sl}}_n\) (for \(n \geq 3\)) is the Kac-Moody algebra with dynkin diagram

\[
\begin{array}{c}
\bullet \\
\cdots \\
\bullet
\end{array}
\]

- \(\hat{\mathfrak{sl}}'_n\) is generated by \(\{E_i, F_i\}_{0 \leq i \leq n-1}\) subject to the relations that for each pair \(0 \leq i < j \leq n - 1\), \(\{E_i, F_i, E_j, F_j\}\) generate a copy of

\[
\begin{cases}
\mathfrak{sl}_3 & \text{if } |i - j| = 1 \text{ mod}(n) \\
\mathfrak{sl}_2 \times \mathfrak{sl}_2 & \text{otherwise}.
\end{cases}
\]
Motivation and background

\( \hat{\mathfrak{sl}}_n \) and its crystals

Definition 1: \( \hat{\mathfrak{sl}}'_n \) is a central extension of the Lie algebra of polynomial loops in \( \mathfrak{sl}_n \) \( \otimes \mathbb{C} \left[ t, t^{-1} \right] \oplus \mathbb{C} \) where \( \mathbb{C} \) is central.

Definition 2: \( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with Dynkin diagram

\[ \begin{array}{cccccccc}
& \bullet & \bullet & \cdots & \bullet & \bullet & \bullet & \\
& \bullet & & & & & & \\
\end{array} \]

\( \hat{\mathfrak{sl}}' \) is generated by \( \{E_i, F_i\}_{0 \leq i \leq n-1} \) subject to the relations that for each pair \( 0 \leq i < j \leq n-1 \), \( \{E_i, F_i, E_j, F_j\} \) generate a copy of

\[ \begin{cases} 
\mathfrak{sl}_3 & \text{if } |i - j| = 1 \mod(n) \\
\mathfrak{sl}_2 \times \mathfrak{sl}_2 & \text{otherwise.}
\end{cases} \]

For \( \hat{\mathfrak{sl}}_4 \):

\[ \begin{array}{cccc}
& \bullet & \bullet & \bullet & \bullet \\
& & & & \\
\end{array} \]
Definition 2: \( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with dynkin diagram

\[ \begin{array}{c}
\bullet \\
\bullet & \cdots & \bullet \\
\bullet \\
\end{array} \]

\( \hat{\mathfrak{sl}}_n \) is generated by \( \{E_i, F_i\}_{0 \leq i \leq n-1} \) subject to the relations that for each pair \( 0 \leq i < j \leq n - 1 \), \( \{E_i, F_i, E_j, F_j\} \) generate a copy of

\[ \begin{cases} 
\mathfrak{sl}_3 & \text{if } |i - j| = 1 \mod(n) \\
\mathfrak{sl}_2 \times \mathfrak{sl}_2 & \text{otherwise.}
\end{cases} \]

For \( \hat{\mathfrak{sl}}_4 \):

\[ E_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \]
\[ \hat{\mathfrak{sl}}_n \text{ and its crystals} \]

- Definition 2: \( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with Dynkin diagram

![Diagram](image_url)

- \( \hat{\mathfrak{sl}}'_n \) is generated by \( \{E_i, F_i\}_{0 \leq i \leq n-1} \) subject to the relations that for each pair \( 0 \leq i < j \leq n-1 \), \( \{E_i, F_i, E_j, F_j\} \) generate a copy of

\[
\begin{cases}
\mathfrak{sl}_3 & \text{if } |i - j| = 1 \mod(n) \\
\mathfrak{sl}_2 \times \mathfrak{sl}_2 & \text{otherwise}.
\end{cases}
\]

For \( \hat{\mathfrak{sl}}_4 \):

\[
E_2 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}, \quad E_0 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
t & 0 & 0 & 0
\end{pmatrix}
\]
Definition 1: \( \hat{\mathfrak{sl}}'_n \) is a central extension of the Lie algebra of polynomial loops in \( \mathfrak{sl}_n \) \( \otimes \mathbb{C} \left[ t, t^{-1} \right] \oplus \mathbb{C} \) where 1 is central.

Definition 2: \( \hat{\mathfrak{sl}}_n \) (for \( n \geq 3 \)) is the Kac-Moody algebra with Dynkin diagram \( \cdots \). \( \hat{\mathfrak{sl}}'_n \) is generated by \( \{ E_i, F_i \} \) \( 0 \leq i \leq n - 1 \) subject to the relations that for each pair \( 0 \leq i < j \leq n - 1 \), \( \{ E_i, F_i, E_j, F_j \} \) generate a copy of \( \mathfrak{sl}_3 \) if \( |i - j| = 1 \mod (n) \), \( \mathfrak{sl}_2 \times \mathfrak{sl}_2 \) otherwise.

For \( \hat{\mathfrak{sl}}_4 \): 
\[
E_2 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}, \quad E_0 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}.
\]
\[ \hat{\mathfrak{sl}}_n \] and its crystals

- Fix \( n \geq 3 \). An (infinite) \( n \)-colored directed graph is an \( \hat{\mathfrak{sl}}_n \) crystal if, for each pair of colors \( c_i \) and \( c_j \), the graph consisting of all edges of those 2 colors is
\( \hat{\mathfrak{sl}}_n \) and its crystals

- Fix \( n \geq 3 \). An (infinite) \( n \)-colored directed graph is an \( \hat{\mathfrak{sl}}_n \) crystal if, for each pair of colors \( c_i \) and \( c_j \), the graph consisting of all edges of those 2 colors is

\[
\begin{cases}
\text{An sl}_3 \text{ crystal graph if } |i - j| = 1 \mod(n) \\
\text{An sl}_2 \times \text{sl}_2 \text{ crystal graph otherwise.}
\end{cases}
\]
We define crystal operators on partitions. Here $(7, 6, 6, 6, 5, 3, 2)$. Color the boxes in the partition periodically with $n = 3$ colors. For instance, we now know that the $q$-character of $V_{\Lambda_0}$ is equal to the generating function of 3-regular partitions counted by size.
We define crystal operators on partitions.
We define crystal operators on partitions.
We define crystal operators on partitions. Here \((7, 6, 6, 6, 5, 3, 2)\).
We define crystal operators on partitions. Here \((7, 6, 6, 6, 5, 3, 2)\).

Color the boxes in the partition periodically with \(n = 3\) colors.
We define crystal operators on partitions. Here \((7, 6, 6, 6, 5, 3, 2)\).

Color the boxes in the partition periodically with \(n = 3\) colors.
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{\mathfrak{sl}}_3$.

- $F_2$ adds a $\bar{2}$ colored box.
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{\mathfrak{sl}}_3$

$F_2$ adds a $\bar{2}$ colored box.
F_2 adds a \( \bar{2} \) colored box.
- $F_2$ adds a $\bar{2}$ colored box.
$F_2$ adds a 2 colored box.
$F_2$ adds a $\bar{2}$ colored box.
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{sl}_3$

- $F_2$ adds a $\bar{2}$ colored box.
$F_2$ adds a $\bar{2}$ colored box.
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{sl}_3$

- $F_2$ adds a $\overline{2}$ colored box.
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{\mathfrak{sl}}_3$

- $F_2$ adds a $\bar{2}$ colored box.
- $E_2$ would send this partition to 0.
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{sl}_3$

Every connected component is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$. The subcrystal generated by the empty partition consists exactly of the 3-regular partitions, which are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).

For instance, we now know that the $q$-character of $V_{\Lambda_0}$ is equal to the generating function of 3-regular partitions counted by size.
Every connected is a copy of $B_{\Lambda_0}$. 
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$. 
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.

The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$.
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.

The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{\mathfrak{sl}}_3$

- Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.
- The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).

Not 3 regular
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{\mathfrak{sl}}_3$

- Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.
- The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.

The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).
The Misra-Miwa-Hayashi realization of $B_{\Lambda_0}$ for $\hat{sl}_3$

Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.

The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.

The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).
Every connected is a copy of $B_{\Lambda_0}$. In particular, the subcrystal generated by the empty partition is a model for $B_{\Lambda_0}$.

The subcrystal generated by the empty partition consists exactly of the 3-regular partitions are a single copy of $B_{\Lambda_0}$ (no 3 rows of same length).

For instance, we now know that the $q$-character of $V_{\Lambda_0}$ is equal to the generating function of 3-regular partitions counted by size.
The multi-partition realization (JMMO, FLOTW)
For $\ell = 4$, use 4-tuples of charged partitions.
The multi-partition realization (JMMO, FLOTW)

For \( \ell = 4 \), use 4-tuples of charged partitions.
The multi-partition realization (JMMO, FLOTW)

- For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition.
The multi-partition realization (JMMO, FLOTW)

For \( \ell = 4 \), use 4-tuples of charged partitions. The charge is a residue mod \( n = 3 \), which is the color of the vertex of the partition. We choose the multi-charge \((0, \bar{1}, \bar{1}, 2)\).
The multi-partition realization (JMMO, FLOTW)

For \( \ell = 4 \), use 4-tuples of charged partitions. The charge is a residue mod \( n = 3 \), which is the color of the vertex of the partition. We choose the multi-charge \((\bar{0}, \bar{1}, \bar{1}, \bar{2})\).
The multi-partition realization (JMMO, FLOTW)

For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge $(\mathbf{0}, \mathbf{1}, \mathbf{1}, \mathbf{2})$. 

\[
\begin{array}{cccc}
0 & 1 & 0 & 1 \\
2 & 0 & 2 & 1 \\
1 & 2 & 0 & 1 \\
0 & 1 & 2 & 2 \\
\end{array}
\]
For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge ($\bar{0}, \bar{1}, \bar{1}, \bar{2}$).
The multi-charge (\(\lambda^{(0)}\))

\[\begin{align*}
\lambda^{(0)} &= \begin{array}{cccc}
0 & 1 & 0 & 1 \\
2 & 0 & 1 & 1 \\
1 & 2 & 0 & 0 \\
0 & 1 & 2 & 0
\end{array} \\
\lambda^{(1)} &= \begin{array}{cccc}
1 & 2 & 1 & 0 \\
0 & 0 & 1 & 0 \\
2 & 0 & 2 & 1 \\
0 & 2 & 0 & 1
\end{array} \\
\lambda^{(2)} &= \begin{array}{cccc}
2 & 1 & 1 & 0 \\
0 & 0 & 2 & 1 \\
2 & 0 & 1 & 0 \\
0 & 1 & 0 & 1
\end{array} \\
\lambda^{(3)} &= \begin{array}{cccc}
2 & 1 & 2 & 0 \\
0 & 1 & 1 & 0 \\
2 & 0 & 0 & 0 \\
0 & 1 & 2 & 2
\end{array}
\end{align*}\]

- For \(\ell = 4\), use 4-tuples of charged partitions. The charge is a residue mod \(n = 3\), which is the color of the vertex of the partition. We choose the multi-charge \((\overline{0}, \overline{1}, \overline{1}, \overline{2})\).
The multi-partition realization (JMMO, FLOTW)

- For \( \ell = 4 \), use 4-tuples of charged partitions. The charge is a residue mod \( n = 3 \), which is the color of the vertex of the partition. We choose the multi-charge \((\bar{0}, \bar{1}, \bar{1}, \bar{2})\). These must satisfy a “shifted containment” condition.
The multi-charge model

- For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge $(\bar{0}, \bar{1}, \bar{1}, \bar{2})$. These must satisfy a "shifted containment" condition.
The multi-charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge $(\bar{0}, \bar{1}, \bar{1}, \bar{2})$. These must satisfy a “shifted containment" condition.

For $\ell = 4$, use 4-tuples of charged partitions.
For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge ($\tilde{0}, \tilde{1}, \tilde{1}, \tilde{2}$). These must satisfy a “shifted containment" condition.
For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge $(0, 1, 1, 2)$. These must satisfy a "shifted containment" condition.
The multi-partition realization (JMMO, FLOTW)

For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge ($\bar{0}, \bar{1}, \bar{1}, \bar{2}$). These must satisfy a “shifted containment" condition.
For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge $(\bar{0}, \bar{1}, \bar{1}, \bar{2})$. These must satisfy a “shifted containment” condition.
The multi-charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge $(0, \bar{1}, \bar{1}, 2)$. These must satisfy a “shifted containment" condition.
The multi-partition realization (JMMO, FLOTW)

- For $\ell = 4$, use 4-tuples of charged partitions. The charge is a residue mod $n = 3$, which is the color of the vertex of the partition. We choose the multi-charge ($\bar{0}, \bar{1}, \bar{1}, \bar{2}$). These must satisfy a “shifted containment" condition.
The multi-partition realization (JMMO, FLOTW)

\[ \lambda^{(0)} \]

\[ \lambda^{(1)} \]

\[ \lambda^{(2)} \]

\[ \lambda^{(3)} \]

- Again \( F_0 \) will add a box colored \( \bar{0} \).
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $0$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
The multi-partition realization (JMMO, FLOTW)

\[ \lambda^{(0)} \]
- Again \( F_0 \) will add a box colored \( \bar{0} \).
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".

\[ \lambda^{(1)} \]

\[ \lambda^{(2)} \]

\[ \lambda^{(3)} \]
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored 0.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
Again $F_0$ will add a box colored $\bar{0}$.

Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".

$\lambda^{(0)}$

$\lambda^{(1)}$

$\lambda^{(2)}$

$\lambda^{(3)}$
The multi-partition realization (JMMO, FLOTW)

\[ \lambda^{(0)} \]

- Again, \( F_0 \) will add a box colored \( \bar{0} \).

\[ \lambda^{(1)} \]

- Again, places you can add a box are labeled "(" and places you can remove a box are labeled ")".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
The multi-partition realization (JMMO, FLOTW)

<table>
<thead>
<tr>
<th>λ^(0)</th>
<th>λ^(1)</th>
<th>λ^(2)</th>
<th>λ^(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\tilde{0})</td>
<td>(\tilde{1})</td>
<td>(\tilde{0})</td>
<td>(\tilde{0})</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>(\bar{2})</td>
<td>(\bar{2})</td>
</tr>
<tr>
<td>(\bar{1})</td>
<td>(\bar{1})</td>
<td>(\bar{0})</td>
<td>(\bar{1})</td>
</tr>
<tr>
<td>(\bar{0})</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

- Again \(F_0\) will add a box colored \(\tilde{0}\).
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")."
The multi-partition realization (JMMO, FLOTW)

\[ \lambda^{(0)} \]

- Again, \( F_0 \) will add a box colored \( \bar{0} \).
- Again, places you can add a box are labeled "(" and places you can remove a box are labeled ")".

\[ \lambda^{(1)} \]

\[ \lambda^{(2)} \]

\[ \lambda^{(3)} \]
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(").
The multi-partition realization (JMMO, FLOTW)

- Again \( F_0 \) will add a box colored \( \bar{0} \).
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and \( F_0 \) adds the box corresponding to the first uncanceled "(".

\[
\begin{array}{c|c|c}
\lambda^{(0)} & 3 & 0 \\
\hline
-3 & 0 & 2 \\
\end{array}
\quad\begin{array}{c|c|c}
\lambda^{(1)} & 3 & 0 \\
\hline
-3 & 0 & 2 \\
\end{array}
\quad\begin{array}{c|c|c}
\lambda^{(2)} & 3 & 0 \\
\hline
-3 & 6 & 2 \\
\end{array}
\quad\begin{array}{c|c|c}
\lambda^{(3)} & 3 & 0 \\
\hline
& 6 & 2 \\
\end{array}
\]

\[
\begin{array}{c|c|c}
\bar{0} & 2 & 0 \\
\hline
2 & 0 & 2 \\
\end{array}
\quad\begin{array}{c|c|c}
\bar{0} & 2 & 0 \\
\hline
2 & 0 & 2 \\
\end{array}
\quad\begin{array}{c|c|c}
\bar{2} & 0 & 0 \\
\hline
\bar{2} & 0 & 0 \\
\end{array}
\quad\begin{array}{c|c|c}
\bar{2} & 0 & 0 \\
\hline
\bar{2} & 0 & 0 \\
\end{array}
\]
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\overline{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "("

The multi-partition realization \((\text{JMMO, FLOTW})\)

- Again \(F_0\) will add a box colored \(\bar{0}\).
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and \(F_0\) adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".

Partiton and cylindric partition models
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(").
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".

\[
\begin{array}{ccc}
\lambda^{(0)} & \lambda^{(1)} & \lambda^{(2)} & \lambda^{(3)} \\
3 & 0 & -3 & 3 & 0 & -3 & 3 & 0 & -3 & 6 & 3 & 0
\end{array}
\]
The multi-partition realization (JMMO, FLOTW)

- Again $F_0$ will add a box colored $\bar{0}$.
- Again places you can add a box are labeled "(" and places you can remove a box are labeled ")".
- The brackets are reordered appropriately, and $F_0$ adds the box corresponding to the first uncanceled "(".
The multi-partition realization (JMMO, FLOTW)

Our example is not 3-regular.
3-regular means no three differently colored rows have the same length.
The multi-partition realization (JMMO, FLOTW)

Every connected component is a copy of $B_\Lambda$. 

\[ \lambda^{(0)} \quad \lambda^{(1)} \quad \lambda^{(2)} \quad \lambda^{(3)} \]

\[ \begin{array}{cccc}
0 & 1 & 1 & 1 \\
2 & 0 & 2 & 1 \\
1 & 2 & 0 & 0 \\
0 & 1 & 0 & 2 \\
\end{array} \]

\[ \begin{array}{cccc}
3 & 0 & 0 & 0 \\
-3 & 0 & -3 & -3 \\
3 & 0 & 0 & 0 \\
3 & 0 & 0 & 0 \\
\end{array} \]
The multi-partition realization (JMMO, FLOTW)

Every connected component is a copy of $B_{\Lambda}$. 

$$\Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2$$
Every connected component is a copy of $B_\Lambda$.

The “3-regular” multi-partitions form a single copy of $B_\Lambda$. 

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]
The multi-partition realization (JMMO, FLOTW)

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]

- Every connected component is a copy of \( B_\Lambda \).
- The “3-regular” multi-partitions form a single copy of \( B_\Lambda \).
- 3-regular means no three differently colored rows have the same length.
The multi-partition realization (JMMO, FLOTW)

Every connected component is a copy of \( B_\Lambda \).

The "3-regular" multi-partitions form a single copy of \( B_\Lambda \).

3-regular means no three differently colored rows have the same length.
The multi-partition realization (JMMO, FLOTW)

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]

- Every connected component is a copy of \( B_\Lambda \).
- The “3-regular" multi-partitions form a single copy of \( B_\Lambda \).
- 3-regular means no three differently colored rows have the same length.
The multi-partition realization (JMMO, FLOTW)

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]

- Every connected component is a copy of \( B_{\Lambda} \).
- The “3-regular" multi-partitions form a single copy of \( B_{\Lambda} \).
- 3-regular means no three differently colored rows have the same length.
The multi-partition realization (JMMO, FLOTW)

Every connected component is a copy of $B_\Lambda$.
The “3-regular” multi-partitions form a single copy of $B_\Lambda$.
3-regular means no three differently colored rows have the same length. Our example is not 3-regular.

$\Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2$
The multi-partition realization (JMMO, FLOTW)

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]
The multi-partition realization (JMMO, FLOTW)

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]

In fact, you can do this by just reading the brackets left to right.
In fact, you can do this by just reading the brackets left to right.

In this form, a different set of multi-partitions appears, called ‘Kleshchev multipartitions.’
In this form, a different set of multi-partitions appears, called ‘Kleshchev multipartitions.’

There are good reasons to use the JMMO rule.
The multi-partition realization (JMMO, FLOTW)

\[ \Lambda = \Lambda_0 + 2\Lambda_1 + \Lambda_2 \]

- In fact, you can do this by just reading the brackets left to right.
- In this form, a different set of multi-partitions appears, called ‘Kleshchev multipartitions.’
- There are good reasons to use the JMMO rule. For one, it is much harder to explicitly describe which tuples are Kleshchev multiupartitions.
Cylindric partitions and $B_\infty$

Consider $n = 3$, $\ell = 2$, and multi-charge $(\overline{0}, \overline{1})$. A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_{\infty}$

- An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a “cylindric partition”.
Cylindric partitions and $B_\infty$

- An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a “cylindric partition”.
- Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$. 
Cylindric partitions and $B_\infty$

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n=3$, $\ell=2$, and multi-charge $\left(\bar{\ell}^0, \bar{\ell}^1\right)$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_\Lambda'$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Partition and cylindric partition models

Cylindric partitions

Cylindric partitions and $B_\infty$

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition". Consider $n=3$, $\ell=2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_{\Lambda}$ if and only if it does not have three differently colored piles of the same height. The embedding $B_{\Lambda} \hookrightarrow B_{\Lambda}'$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_{\infty}$

A cylindric partition is in $B_{\Lambda}$ if and only if it does not have three differently colored piles of the same height. The embedding $B_{\Lambda} \hookrightarrow \to B_{\Lambda}'$ are given by "shifting." The imbedding into $B_{\infty}$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \rightarrow B_\Lambda'$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
A "multi-segment" $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\infty$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow \to B_{\Lambda}'$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow \rightarrow B_\Lambda'$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{\ell}, \ldots, \bar{1}, \bar{0})$ of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow \rightarrow B_\Lambda'$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_\Lambda'$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B$ if and only if it does not have three differently colored piles of the same height. The embedding $B \hookrightarrow \cdots$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
Cylindric partitions and $\mathcal{B}_\infty$

Cylindric partitions and $B_{\infty}$

A cylindric partition is in $\mathcal{B}_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $\mathcal{B}_\Lambda \hookrightarrow \mathcal{B}_{\Lambda'}$ are given by "shifting". The imbedding into $\mathcal{B}_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $\mathcal{B}_\infty$ crystal structure reads boxes in order of height.

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{0}, \bar{1}, \cdots)$

A cylindric partition is in $B_{\Lambda}$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_{\Lambda} \hookrightarrow \cdots \hookrightarrow B_{\Lambda}'$ are given by "shifting".

The imbedding into $B_{\infty}$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_{\infty}$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B\Lambda \hookrightarrow B\Lambda'$ are given by "shifting".

The imbedding into $B\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{0}, \bar{1}, \bar{2})$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.

\[ \begin{array}{ccc}
2 & 1 & 2 \\
1 & & 1 \\
0 & 2 & 0 \\
\end{array} \]
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{\lambda}_0, \bar{\lambda}_1, \ldots, \bar{\lambda}_\ell)$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{\lambda}_0, \bar{\lambda}_1)$. A cylindric partition is in $B_\infty$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
A "multi-segment" $(\overline{0}, \overline{1})$ implies that an $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three-dimensional picture called a "cylindric partition".

A cylindric partition is in $B\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B\Lambda \hookrightarrow \Lambda' \rightarrow B\Lambda'$ are given by "shifting". The embedding into $B\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B\infty$ crystal structure reads boxes in order of height.
A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting." The embedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
The embeding $B_{\Lambda} \hookrightarrow B_{\Lambda'}$ are given by “shifting".
Cylindric partitions and $B_\infty$

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \to B_{\Lambda'}$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

$B_{\Lambda_0 + \Lambda_1} \leftrightarrow B_{2\Lambda_0 + \Lambda_1}$
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.

$\bar{0}$ $\bar{1}$ $\leftrightarrow$ $B_{2\Lambda_0 + \Lambda_1}$
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow \to B_\Lambda'$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n=3$, $\ell=2$, and multi-charge $(\overline{0}, \overline{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment"

($\cdots$)

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_{\infty}$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_{\infty}$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_{\infty}$

A cylindric partition is in $B_{\Lambda}$ if and only if it does not have three differently colored piles of the same height. The embedding $B_{\Lambda} \hookrightarrow B_{\Lambda}'$ are given by "shifting." The imbedding into $B_{\infty}$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_{\infty}$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow \rightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_\Lambda'$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
Cylindric partitions and $B_\infty$

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.

For $f_{\bar{2}}$
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{\ell})$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition". Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{\ell})$. A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $(\overline{\lambda}_0, \overline{\lambda}_1)$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_{\infty}$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_{\infty}$ crystal structure reads boxes in order of height.

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".
Cylindric partitions and $B_\infty$

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition".

Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$. A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions. The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

- The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
Cylindric partitions and $B_\infty$

- The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.
Cylindric partitions and $B_{\infty}$

A cylindric partition is in $B_{\Lambda}$ if and only if it does not have three differently colored piles of the same height. The embedding $B_{\Lambda} \rightarrow B_{\Lambda}'$ are given by "shifting." The imbedding into $B_{\infty}$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

For $\bar{\ell} = (\bar{0}, \bar{1}, \bar{2})$.
Cylindric partitions and $B_\infty$
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $(\vec{0}, \vec{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow \cdots \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

A "multi-segment" $(\overline{0}, \overline{1}, \overline{2})$.

An $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three-dimensional picture called a "cylindric partition".

Consider $n=3$, $\ell=2$, and multi-charge $(\overline{0}, \overline{1})$.

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_{\Lambda'}$ are given by "shifting".

The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height.
A “multi-segment"
Cylindric partitions and $B_\infty$

- The $B_\infty$ crystal structure reads boxes in order of height.
Cylindric partitions and $B_\infty$

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$: 

\[
\begin{array}{cccc}
\bar{2} & \bar{1} & 1 & 2 \\
0 & 0 & 1 & 1 & 2
\end{array}
\]
Cylindric partitions and $B_\infty$

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$.

\[
\begin{array}{cccc}
\bar{2} & \bar{1} & \bar{1} & \bar{2} \\
\bar{1} & \bar{0} & \bar{1} & \bar{1} \\
\bar{0} & \bar{0} & \bar{1} & \bar{2}
\end{array}
\]
Cylindric partitions and $B_\infty$

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$:
Cylindric partitions and $B_\infty$

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$: 

\[
\begin{array}{cccccc}
\bar{2} & \bar{1} & \bar{1} & 2 & 1 & 2 \\
\bar{0} & \bar{0} & 1 & 2 & 1 & \bar{2}
\end{array}
\]
Cylindric partitions and $B_{\infty}$

The $B_{\infty}$ crystal structure reads boxes in order of height. For $f_2$:...
Cylindric partitions and $B_\infty$

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$:

\[
\begin{array}{cccc}
2 & 1 & 2 & 1 \\
1 & 0 & 1 & 2 \\
0 & 0 & 1 & 2 \\
\end{array}
\]

\* The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$:
Cylindric partitions and $B_{\infty}$

The $B_{\infty}$ crystal structure reads boxes in order of height. For $f_2$: 

$$
\begin{array}{ccccccc}
\bar{2} & \bar{1} & \bar{1} & \bar{2} & \bar{1} & \bar{1} & \bar{2} \\
\bar{0} & \bar{0} & \bar{1} & \bar{1} & \bar{2} & \end{array}
$$
Cylindric partitions and $B_\infty$

A cylindric partition is in $B_\Lambda$ if and only if it does not have three differently colored piles of the same height.

The embedding $B_\Lambda \hookrightarrow B_\Lambda'$ are given by "shifting." The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$: 

- The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$: 

$\begin{array}{cccc}
\overline{2} & \overline{1} & \overline{1} & \overline{2} \\
\overline{1} & \overline{0} & \overline{1} & \overline{2} \\
\overline{0} & \overline{0} & \overline{1} & \overline{2}
\end{array}$
The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$: 

\[ \begin{array}{cccc} 
\bar{2} & \bar{1} & 2 \\
\bar{1} & \bar{0} & \bar{1} & 1 & 2 \\
\bar{0} & \bar{0} & \bar{1} & 1 & 2 
\end{array} \]
A "multi-segment" $\ell$-tuple of partitions satisfying the shifted containment conditions fits together into a three dimensional picture called a "cylindric partition". Consider $n = 3$, $\ell = 2$, and multi-charge $(\overline{0}, \overline{1})$. A cylindric partition is in $B_\infty$ if and only if it does not have three differently colored piles of the same height. The embedding $B_\Lambda \hookrightarrow B_\Lambda'$ are given by "shifting". The imbedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

- The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$:
Cylindric partitions and $B_\infty$

The $B_\infty$ crystal structure reads boxes in order of height. For $f_2$: 

\begin{align*}
\bar{2} & \quad \bar{1} & \quad \bar{1} & \quad \bar{2} & \quad \bar{1} & \quad 1 & \quad 2 & \quad 2 \\
\bar{1} & \quad \bar{0} & \quad \bar{1} & \quad \bar{1} & \quad 1 & \quad 1 & \quad 2 & \quad 2 \\
0 & \quad \bar{0} & \quad \bar{1} & \quad 1 & \quad 1 & \quad 2 & \quad 2 & \quad 2 \\
\end{align*}
Cylindric partitions and $B_\infty$

A "multi-segment" $(\bar{\ell}, \bar{\ell}, \bar{\ell})$-tuple of partitions satisfying the shifted containment conditions fits together into a three-dimensional picture called a "cylindric partition". Consider $n = 3$, $\ell = 2$, and multi-charge $(\bar{0}, \bar{1})$.

A cylindric partition is in $B_{\Lambda}$ if and only if it does not have three differently colored piles of the same height. The embedding $B_{\Lambda} \hookrightarrow B_{\Lambda}'$ are given by "shifting". The embedding into $B_\infty$ just records the vertical piles, not the arrangement into an $\ell$-tuple of partitions.

Cylindric partitions are only needed to describe the image of $B_{\Lambda}$. 
Application: generating functions/partition functions
Application: generating functions/partition functions
Application: generating functions/partition functions
The generating function for cylindric partitions on a given cylinder is a specialization of the Weyl character formula.
Application: generating functions/partition functions

- The generating function for cylindric partitions on a given cylinder is a specialization of the Weyl character formula. Since we want all cylindric partitions, not just $\ell$ regular ones, use Weyl character formula for $\hat{\mathfrak{gl}}_n$, not $\hat{\mathfrak{sl}}_n$. 

Corollary: 
$$\sum_{\pi \text{ on a given cylinder}} q^{\vert \pi \vert} = \dim q(W_\Lambda),$$ 
where $W_\Lambda$ is an irreducible representation of $\hat{\mathfrak{gl}}_n$ at level $\ell$.

(Calculated by A. Borodin in a different form).
The generating function for cylindric partitions on a given cylinder is a specialization of the Weyl character formula. Since we want all cylindric partitions, not just \( \ell \) regular ones, use Weyl character formula for \( \hat{\mathfrak{g}l}_n \), not \( \hat{\mathfrak{sl}}_n \).

**Corollary**

\[
\sum_{\pi \text{ on a given cylinder}} q^{|\pi|} = \dim_q(W_\Lambda), \text{ where } W_\Lambda \text{ is an irreducible representation of } \hat{\mathfrak{g}l}_n \text{ at level } \ell. \text{ (Calculated by A. Borodin in a different form).}
\]
(Borodin 2006) The partition function for cylindric plane partitions is given by:

\[ Z := \sum_{\text{cylindric partitions } \pi} q^{\|\pi\|} = \prod_{k \geq 1} \frac{1}{1 - q^{kN}} \prod_{i \in [1,N]} \frac{1}{1 - q^{(i-j)(N)+(k-1)N}}. \]

For any \( k \in \mathbb{Z} \), \( k(N) \) is the smallest non-negative integer congruent to \( k \) modulo \( N \).

1, \( N \) is the set of integers modulo \( N \).

\[ A[i] = \begin{cases} 1 & \text{if the boundary is sloping up and to the right at } i \\ 0 & \text{otherwise} \end{cases} \]
Borodin’s result

Theorem

\[(\text{Borodin 2006}) \text{ The partition function for cylindric plane partitions is given by:}\]

\[
Z := \sum_{\text{cylindric partitions } \pi} q^{|\pi|} = \prod_{k \geq 1} \frac{1}{1 - q^{kN}} \prod_{i \in \overline{1, N}} \frac{1}{1 - q^{(i-j)(N)+(k-1)N}}.
\]

\[
N = n + \ell
\]
Borodin’s result

**Theorem**

*(Borodin 2006)* The partition function for cylindric plane partitions is given by:

\[
Z := \sum_{\text{cylindric partitions } \pi} q^{|\pi|} = \prod_{k \geq 1} \frac{1}{1 - q^{kN}} \prod_{i \in \overline{1,N} : A[i] = 1} \frac{1}{1 - q^{(i-j)(N)+(k-1)N}}.
\]

- \( N = n + \ell \)
- For any \( k \in \mathbb{Z}, k(N) \) is the smallest non-negative integer congruent to \( k \) modulo \( N \).
**Borodin’s result**

**Theorem**

(Borodin 2006) The partition function for cylindric plane partitions is given by:

\[ Z := \sum_{\text{cylindric partitions } \pi} q^{\mid \pi \mid} = \prod_{k \geq 1} \frac{1}{1 - q^{kN}} \prod_{i \in \overline{1,N}} \frac{1}{1 - q^{(i-j)(N)+(k-1)N}}. \]

- \( N = n + \ell \)
- For any \( k \in \mathbb{Z} \), \( k(N) \) is the smallest non-negative integer congruent to \( k \) modulo \( N \).
- \( \overline{1, N} \) is the set of integers modulo \( N \).

For any \( k \in \mathbb{Z} \), \( k(N) \) is the smallest non-negative integer congruent to \( k \) modulo \( N \).

Theorem

\( Z := \sum_{\text{cylindric partitions } \pi} q^{\mid \pi \mid} = \prod_{k \geq 1} \frac{1}{1 - q^{kN}} \prod_{i \in \overline{1,N}} \frac{1}{1 - q^{(i-j)(N)+(k-1)N}}. \)

- \( N = n + \ell \)
- For any \( k \in \mathbb{Z} \), \( k(N) \) is the smallest non-negative integer congruent to \( k \mod N \).
- \( \overline{1, N} \) is the set of integers modulo \( N \).
Borodin’s result

Theorem

(Borodin 2006) The partition function for cylindric plane partitions is given by:

\[
Z := \sum_{\text{cylindric partitions } \pi} q^{\mid \pi \mid} = \prod_{k \geq 1} \frac{1}{1 - q^{kN}} \prod_{i \in \overline{1,N} : A[i] = 1} \frac{1}{1 - q^{(i-j)(N)+(k-1)N}}.
\]

- \( N = n + \ell \)
- For any \( k \in \mathbb{Z}, k(N) \) is the smallest non-negative integer congruent to \( k \) modulo \( N \).
- \( \overline{1,N} \) is the set of integers modulo \( N \).
- \( A[i] = \begin{cases} 1 & \text{if the boundary is sloping up and to the right at } i \\ 0 & \text{otherwise} \end{cases} \)
In fact, Borodin also calculated correlation functions for a system of random cylindric partitions. Although to do that, he needed to use "shift-mixing." It seems you don't get a determinental process otherwise (I'm quoting Borodin, and he says this hasn't been proven). But "shift mixing" is meaningful in representation theory as well. It means you are looking at a representation of $\hat{\mathfrak{sl}}_n \oplus C_l$, where $C_l$ is an infinite dimensional Clifford algebra. This is actually done quite often.

Question: what do Borodin's results mean representation theoretically?

Answer: They tell you something about expected behavior of randomly chosen basis vectors... but it is really a statistic on the combinatorial indexing set, I don't know what it means in any deeper sense.
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.
In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.

Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise.
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.
- Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.
- Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).
- But “shift mixing” is meaningful in representation theory as well.
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.
- Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).
- But “shift mixing” is meaningful in representation theory as well. It means you are looking at a representation of \( \hat{\mathfrak{sl}}_n \oplus Cl \), where \( Cl \) is an infinite dimensional Clifford algebra.
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.
- Although to do that, he needed to use “shift-mixing." It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).
- But “shift mixing" is meaningful in representation theory as well. It means you are looking at a representation of $\hat{\mathfrak{sl}}_n \oplus Cl$, where $Cl$ is an infinite dimensional Clifford algebra. This is actually done quite often.
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions
- Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).
- But “shift mixing” is meaningful in representation theory as well. It means you are looking at a representation of $\hat{\mathfrak{sl}}_n \oplus Cl$, where $Cl$ is an infinite dimensional Clifford algebra. This is actually done quite often.
- Question: what do Borodin’s results mean representation theoretically?
In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.

Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).

But “shift mixing" is meaningful in representation theory as well. It means you are looking at a representation of $\widehat{\mathfrak{sl}}_n \oplus Cl$, where $Cl$ is an infinite dimensional Clifford algebra. This is actually done quite often.

Question: what do Borodin’s results mean representation theoretically?

Answer: They tell you something about expected behavior of randomly chosen basis vectors...
Borodin’s result

- In fact, Borodin also calculated correlation functions for a system of random cylindric partitions.
- Although to do that, he needed to use “shift-mixing.” It seems you don’t get a determinental process otherwise (I’m quoting Borodin, and he says this hasn’t been proven).
- But “shift mixing” is meaningful in representation theory as well. It means you are looking at a representation of $\mathfrak{sl}_n \oplus Cl$, where $Cl$ is an infinite dimensional Clifford algebra. This is actually done quite often.
- Question: what do Borodin’s results mean representation theoretically?
- Answer: They tell you something about expected behavior of randomly chosen basis vectors...but it is really a statistic on the combinatorial indexing set, I don’t know what it means in any deeper sense.
Application: Level-rank duality
Application: Level-rank duality
Application: Level-rank duality
You can interpret a given cylinder as a level $\ell$ highest weight for $\hat{\mathfrak{sl}}_n$ or a level $n$ highest weight for $\hat{\mathfrak{sl}}_\ell$. Thus we observe:
Application: Level-rank duality

- You can interpret a given cylinder as a level $\ell$ highest weight for $\hat{\mathfrak{sl}}_n$ or a level $n$ highest weight for $\hat{\mathfrak{sl}}_\ell$. Thus we observe:

Theorem (originally due to I. Frenkel)

Let $W_\Lambda$ be an irreducible integrable level $\ell$ representation of $\hat{\mathfrak{gl}}_n$. There is a corresponding level $n$ irreducible integral representation $W_\Lambda'$ of $\hat{\mathfrak{gl}}_\ell$ so that $\dim_q(W_\Lambda) = \dim_q(W_\Lambda')$. 
You can interpret a given cylinder as a level $\ell$ highest weight for $\hat{\mathfrak{sl}}_n$ or a level $n$ highest weight for $\hat{\mathfrak{sl}}_\ell$. Thus we observe:

**Theorem (originally due to I. Frenkel)**

Let $W_\Lambda$ be an irreducible integrable level $\ell$ representation of $\hat{\mathfrak{sl}}_n$. 
You can interpret a given cylinder as a level $\ell$ highest weight for $\mathfrak{sl}_n$ or a level $n$ highest weight for $\mathfrak{sl}_\ell$. Thus we observe:

**Theorem (originally due to I. Frenkel)**

Let $W_\Lambda$ be an irreducible integrable level $\ell$ representation of $\hat{\mathfrak{sl}}_n$. There is a corresponding level $n$ irreducible integral representation $W_\Lambda'$ of $\hat{\mathfrak{sl}}_\ell$ so that

$$\dim_q(W_\Lambda) = \dim_q(W_\Lambda').$$
Application: Level-rank duality

- You can interpret a given cylinder as a level $\ell$ highest weight for $\hat{\mathfrak{sl}}_n$ or a level $n$ highest weight for $\hat{\mathfrak{sl}}_\ell$. Thus we observe:

**Theorem (originally due to I. Frenkel)**

Let $W_\Lambda$ be an irreducible integrable level $\ell$ representation of $\hat{\mathfrak{gl}}_n$. There is a corresponding level $n$ irreducible integral representation $W_{\Lambda'}$ of $\hat{\mathfrak{gl}}_\ell$ so that

$$\dim_q(W_\Lambda) = \dim_q(W_{\Lambda'}).$$
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model
Relation to the Kyoto path model

\[ \cdots \begin{array}{c} \bar{1} \ 2 \end{array} \otimes \begin{array}{c} 0 \ 1 \end{array} \otimes \begin{array}{c} 0 \ 2 \end{array} \otimes \begin{array}{c} 1 \ 2 \end{array} \otimes \begin{array}{c} 1 \ 2 \end{array} \otimes \begin{array}{c} 1 \ 1 \end{array} \otimes \begin{array}{c} 0 \ 1 \end{array} \]
Recent development: Berg/Fayers’ crystals
Recent development: Berg/Fayers’ crystals

- Define new operators $E_i$ and $F_i$ on the set of partitions.
Define new operators $E_i$ and $F_i$ on the set of partitions.
Recent development: Berg/Fayers’ crystals

- Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.
- For $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.

- for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Recent development: Berg/Fayers’ crystals

- Define new operators $E_i$ and $F_i$ on the set of partitions.
- For $i = 2$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.

for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Define new operators $E_i$ and $F_i$ on the set of partitions.

for $i = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Define new operators $E_\bar{i}$ and $F_\bar{i}$ on the set of partitions.

for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.

- for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Recent development: Berg/Fayers’ crystals

- Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.
- For $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Recent development: Berg/Fayers’ crystals

- Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.
- for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Recent development: Berg/Fayers’ crystals

- Define new operators $E_i$ and $F_i$ on the set of partitions.
- For $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.

for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
Recent development: Berg/Fayers’ crystals

Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.

- for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.

- $F_{\bar{2}}$ adds the box corresponding to the first uncanceled $\circ$. 

CAUTION: other components are not all crystals.

A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
Define new operators $E_i$ and $F_i$ on the set of partitions.

- for $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.
- $F_{\bar{2}}$ adds the box corresponding to the first uncanceled $\triangleright$. 
Define new operators $E_i$ and $F_i$ on the set of partitions.

for $i = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.

$F_{\bar{2}}$ adds the box corresponding to the first uncanceled $\circ$. 
Current work  Fayers’ crystals

Recent development: Berg/Fayers’ crystals

Define new operators $E_{\bar{i}}$ and $F_{\bar{i}}$ on the set of partitions.

For $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left.

$F_{\bar{2}}$ adds the box corresponding to the first uncanceled $\ddash$.

The component generated by the empty partition is a copy of $B_{\Lambda_0}$.

**CAUTION:** other components are not all crystals.

A partition is in $\mathcal{B}_{\Lambda_0}$ if and only if there are no illegal hooks.

One can actually read the boxes according to ANY slope (in a certain range).

The same result is true, although definition of "illegal hook" is a bit more complicated.

This gives uncountably many realizations of $B_{\Lambda_0}$.
Recent development: Berg/Fayers’ crystals

The component generated by the empty partition is a copy of $B(\Lambda_0)$. 
Recent development: Berg/Fayers’ crystals

- The component generated by the empty partition is a copy of $B(\Lambda_0)$.
- CAUTION: other components are not all crystals.
Recent development: Berg/Fayers’ crystals

- The component generated by the empty partition is a copy of $B(\Lambda_0)$.
- **CAUTION:** other components are not all crystals.
- A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
Recent development: Berg/Fayers’ crystals

- The component generated by the empty partition is a copy of $B(\Lambda_0)$.
- **CAUTION**: other components are not all crystals.
- A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
Recent development: Berg/Fayers’ crystals

- The component generated by the empty partition is a copy of $B(\Lambda_0)$.
- **CAUTION**: other components are not all crystals.
- A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
Recent development: Berg/Fayers’ crystals

- The component generated by the empty partition is a copy of $B(\Lambda_0)$.
- CAUTION: other components are not all crystals.
- A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
Recent development: Berg/Fayers’ crystals

- The component generated by the empty partition is a copy of $B(\Lambda_0)$.
- CAUTION: other components are not all crystals.
- A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
The component generated by the empty partition is a copy of $B(\Lambda_0)$.

CAUTION: other components are not all crystals.

A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks.
Recent development: Berg/Fayers’ crystals

Define new operators $E = \bar{2}$ and $F = \bar{1}$ on the set of partitions. For $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left. $F\bar{2}$ adds the box corresponding to the first uncanceled $\bar{3}$. The component generated by the empty partition is a copy of $B(\Lambda_0)$. Caution: other components are not all crystals. A partition is in $B(\Lambda_0)$ if and only if there are no illegal hooks. One can actually read the boxes according to ANY slope (in a certain range). The same result is true, although definition of “illegal hook” is a bit more complicated. This gives uncountably many realizations of $B\Lambda_0$. 

Peter Tingley (MIT)
Recent developments: Berg/Fayers’ crystals

One can actually read the boxes according to ANY slope (in a certain range)
Recent development: Berg/Fayers’ crystals

One can actually read the boxes according to ANY slope (in a certain range)
Fayers’ crystals

Recent development: Berg/Fayers’ crystals

- One can actually read the boxes according to ANY slope (in a certain range)
Define new operators $E\bar{i}$ and $F\bar{i}$ on the set of partitions. For $\bar{i} = \bar{2}$, construct a string of brackets as before, but ordered lexicographically by height, then right to left. $F\bar{2}$ adds the box corresponding to the first uncanceled $\dashv$.

The component generated by the empty partition is a copy of $\mathcal{B}(\Lambda_0)$. CAUTION: other components are not all crystals.

A partition is in $\mathcal{B}(\Lambda_0)$ if and only if there are no illegal hooks.

One can actually read the boxes according to ANY slope (in a certain range). The same result is true, although definition of "illegal hook" is a bit more complicated. This gives uncountably many realizations of $\mathcal{B}(\Lambda_0)$.
Recent development: Berg/Fayers’ crystals

- One can actually read the boxes according to ANY slope (in a certain range)
Recent development: Berg/Fayers’ crystals

- One can actually read the boxes according to ANY slope (in a certain range)
- The same result is true, although definition of "illegal hook" is a bit more complicated.
Recent development: Berg/Fayers’ crystals

- One can actually read the boxes according to ANY slope (in a certain range)
- The same result is true, although definition of "illegal hook" is a bit more complicated.
- This gives uncountably many realizations of $B_{\Lambda_0}$.
I established a connection between one case of Fayers' crystal and a case of Nakajima's monomial crystal. Nakajima's monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers' crystals exist.

The rest of the picture works at higher level. Can Fayers' rule be extended beyond level 1? Current work with Steven Sam is going to answer at least some of this. We can show that the 'slope' in Fayers model comes from a choice of $\mathbb{C}^*$ action on Nakajima's quiver varieties. This should work at higher levels, and in fact in more general quiver varieties. Maybe we'll even find some new combinatorics.
Future directions

- I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal
I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal.

Nakajima’s monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers’ crystals exist.
I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal.

Nakajima’s monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers’ crystals exist.

The rest of the picture works at higher level. Can Fayers’ rule be extended beyond level 1?
Future directions

- I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal.

- Nakajima’s monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers’ crystals exist.

- The rest of the picture works at higher level. Can Fayers’ rule be extended beyond level 1?

- Current work with Steven Sam is going to answer at least some of this.
I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal.

Nakajima’s monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers’ crystals exist.

The rest of the picture works at higher level. Can Fayers’ rule be extended beyond level 1?

Current work with Steven Sam is going to answer at least some of this. We can show that the ‘slope’ in Fayers model comes from a choice of $\mathbb{C}^*$ action on Nakajima’s quiver varieties.
I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal

Nakajima’s monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers’ crystals exist.

The rest of the picture works at higher level. Can Fayers’ rule be extended beyond level 1?

Current work with Steven Sam is going to answer at least some of this. We can show that the ‘slope’ in Fayers model comes from a choice of $\mathbb{C}^*$ action on Nakajima’s quiver varieties. This should work at higher levels, and in fact in more general quiver varieties.
I established a connection between one case of Fayers’ crystal and a case of Nakajima’s monomial crystal.

Nakajima’s monomial crystal comes from deep algebraic and geometric structures (q-characters; quiver varieties). Perhaps these can be used to understand why Fayers’ crystals exist.

The rest of the picture works at higher level. Can Fayers’ rule be extended beyond level 1?

Current work with Steven Sam is going to answer at least some of this. We can show that the ‘slope’ in Fayers model comes from a choice of \( \mathbb{C}^* \) action on Nakajima’s quiver varieties. This should work at higher levels, and in fact in more general quiver varieties. Maybe we’ll even find some new combinatorics.