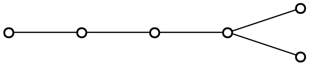
The Product Monomial Crystal

Joel Gibson
The University of Sydney
Supervisor: Dr. Oded Yacobi

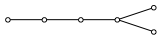
Presented at AustMS 2018, The University of Adelaide

December 5, 2018

MOTIVATION: NAKAJIMA QUIVER VARIETIES

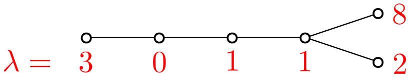


Motivation: Nakajima quiver varieties

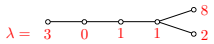


I've been studying a particular crystal called the *product monomial crystal*, which arises from a construction in geometric representation theory called a "Nakajima quiver variety". I'm just going to briefly sketch the underlying story here about how this crystal arises. Suppose we start off with a simple graph, for example the tree shown on the left.

MOTIVATION: NAKAJIMA QUIVER VARIETIES

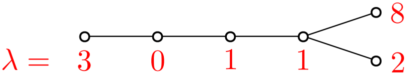


MOTIVATION: NAKAJIMA QUIVER VARIETIES



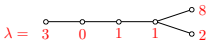
We also need to take some assignment of natural numbers to each node of the graph, which we call λ.

MOTIVATION: NAKAJIMA QUIVER VARIETIES



$\mathcal{M}(\lambda)$

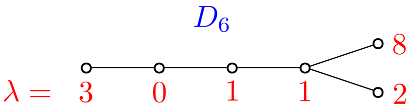
MOTIVATION: NAKAJIMA QUIVER VARIETIES



$\mathcal{M}(\lambda)$

Nakajima gave a construction to produce from this data an algebraic variety $\mathcal{M}(\lambda)$, called a "Nakajima quiver variety". This algebraic variety is smooth, and made up of many different connected components. The amazing thing about these varieties is that they naturally produce modules for Lie algebras.

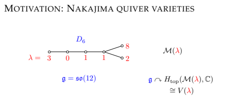
MOTIVATION: NAKAJIMA QUIVER VARIETIES



$\mathcal{M}(\lambda)$

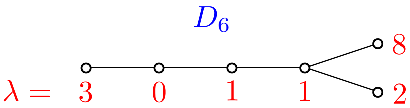
$\mathfrak{g} = \mathfrak{so}(12)$

$$\mathfrak{g} \curvearrowright H_{\text{top}}(\mathcal{M}(\lambda), \mathbb{C}) \cong V(\lambda)$$



The graph we started with is actually a Dynkin diagram of type D_6, and by Nakajima's results the top homology of the quiver variety will have a $\mathfrak{g}$-action, where $\mathfrak{g}$ is a Lie algebra associated to type D_6, for example $\mathfrak{so}(12)$. This top homology is finite-dimensional, and so we can ask how it decomposes in terms of irreducible highest-weight $\mathfrak{g}$-modules. In fact, it will be the *irreducible* $\mathfrak{g}$-module with highest-weight λ, where each number on the Dynkin diagram corresponds to a multiple of a fundamental weight. For example, here we would have $\lambda = 3\varpi_1 + 0\varpi_2 + \varpi_3 + \varpi_4 + 8\varpi_5 + 2\varpi_6$.

MOTIVATION: NAKAJIMA QUIVER VARIETIES



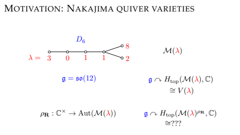
$\mathcal{M}(\lambda)$

$\mathfrak{g} = \mathfrak{so}(12)$

$$\mathfrak{g} \curvearrowright H_{\text{top}}(\mathcal{M}(\lambda), \mathbb{C}) \cong V(\lambda)$$

$\rho_{\mathbf{R}} : \mathbb{C}^\times \to \operatorname{Aut}(\mathcal{M}(\lambda))$

$$\mathfrak{g} \curvearrowright H_{\text{top}}(\mathcal{M}(\lambda)^{\rho_{\mathbf{R}}}, \mathbb{C}) \cong ???$$



The quiver variety $\mathcal{M}(\lambda)$ has a large family of different $\mathbb{C}^\times$-actions, which are parametrised somehow — here I've called the parameter $\mathbf{R}$. The top homology of the fixed-point subvariety of this action is again a $\mathfrak{g}$-module, but the decomposition of this module is mysterious. On one hand, if the action is trivial, we again get the irreducible module $V(\lambda)$. On the other hand, when the action is fairly generic, Nakajima showed that the module is a tensor product of the fundamental weights making up λ. This is also the "largest" that the module could possibly be.

The *product monomial crystal* is the crystal associated to this mysterious module.

Caution: I omitted many technicalities, so much of what I just said is technically untrue. But this is the "big picture".

SETUP

Fix some Lie-theoretic data:

1. $\mathfrak{g}$ a semisimple simply-laced complex Lie algebra $\mathfrak{g}$.
2. $\mathfrak{h} \subseteq \mathfrak{b} \subseteq \mathfrak{g}$ a choice of Cartan and Borel.

Setup

Fix some Lie-theoretic data:

1. $\mathfrak{g}$ a semisimple simply-laced complex Lie algebra $\mathfrak{g}$.
2. $\mathfrak{h} \subseteq \mathfrak{b} \subseteq \mathfrak{g}$ a choice of Cartan and Borel.

First I would like to talk a little about what crystals are, as "combinatorial shadows" of representations. Let us fix a complex semisimple Lie algebra $\mathfrak{g}$, containing a Borel subalgebra $\mathfrak{b}$ and a Cartan subalgebra $\mathfrak{h}$.

Setup

Fix some Lie-theoretic data:

1. $\mathfrak{g}$ a semisimple simply-laced complex Lie algebra $\mathfrak{g}$.
2. $\mathfrak{h} \subseteq \mathfrak{b} \subseteq \mathfrak{g}$ a choice of Cartan and Borel.

$$\mathfrak{sl}_3 \text{ example:} \quad \underbrace{\begin{pmatrix} * & & \\ & * & \\ & & * \end{pmatrix}}_{\mathfrak{h}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ & * & * \\ & & * \end{pmatrix}}_{\mathfrak{b}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ * & * & * \\ * & * & * \end{pmatrix}}_{\mathfrak{g}=\mathfrak{sl}_3}$$

Setup

Fix some Lie-theoretic data:

1. $\mathfrak{g}$ a semisimple simply-laced complex Lie algebra $\mathfrak{g}$.
2. $\mathfrak{h} \subseteq \mathfrak{b} \subseteq \mathfrak{g}$ a choice of Cartan and Borel.

$$\mathfrak{sl}_3 \text{ example:} \quad \underbrace{\begin{pmatrix} * & & \\ & * & \\ & & * \end{pmatrix}}_{\mathfrak{h}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ & * & * \\ & & * \end{pmatrix}}_{\mathfrak{b}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ * & * & * \\ * & * & * \end{pmatrix}}_{\mathfrak{g}=\mathfrak{sl}_3}$$

For example, if $\mathfrak{g} = \mathfrak{sl}_3$ the 3×3 matrices with zero trace, then a Cartan subalgebra is the diagonal matrices, and a compatible Borel subalgebra is the upper-triangular matrices.

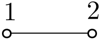
SETUP

Fix some Lie-theoretic data:

1. $\mathfrak{g}$ a semisimple simply-laced complex Lie algebra $\mathfrak{g}$.
2. $\mathfrak{h} \subseteq \mathfrak{b} \subseteq \mathfrak{g}$ a choice of Cartan and Borel.

$$\mathfrak{sl}_3 \text{ example:} \qquad \underbrace{\begin{pmatrix} * & & \\ & * & \\ & & * \end{pmatrix}}_{\mathfrak{h}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ & * & * \\ & & * \end{pmatrix}}_{\mathfrak{b}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ * & * & * \\ * & * & * \end{pmatrix}}_{\mathfrak{g}=\mathfrak{sl}_3}$$

Then, for free, get

1. A Dynkin diagram I, a simple graph. $I =$ 

2. A weight lattice P, and dominant weights P^+.

Setup

Fix some Lie-theoretic data:

1. $\mathfrak{g}$ a semisimple simply-laced complex Lie algebra $\mathfrak{g}$.
2. $\mathfrak{h} \subseteq \mathfrak{b} \subseteq \mathfrak{g}$ a choice of Cartan and Borel.

$$\mathfrak{sl}_3 \text{ example:} \quad \underbrace{\begin{pmatrix} * & & \\ & * & \\ & & * \end{pmatrix}}_{\mathfrak{h}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ & * & * \\ & & * \end{pmatrix}}_{\mathfrak{b}} \subseteq \underbrace{\begin{pmatrix} * & * & * \\ * & * & * \\ * & * & * \end{pmatrix}}_{\mathfrak{g}=\mathfrak{sl}_3}$$

Then, for free, get

1. A Dynkin diagram I, a simple graph. $I =$ 
2. A weight lattice P, and dominant weights P^+.

After fixing this data, we get for free a Dynkin diagram I, which will be a simple graph (as long as our original semisimple Lie algebra was simply-laced), a weight lattice, simple roots, dominant weights, and so on. The irreducible finite-dimensional representations of $\mathfrak{g}$ are in one-to-one correspondence with the dominant integral weights.

Characters of representations

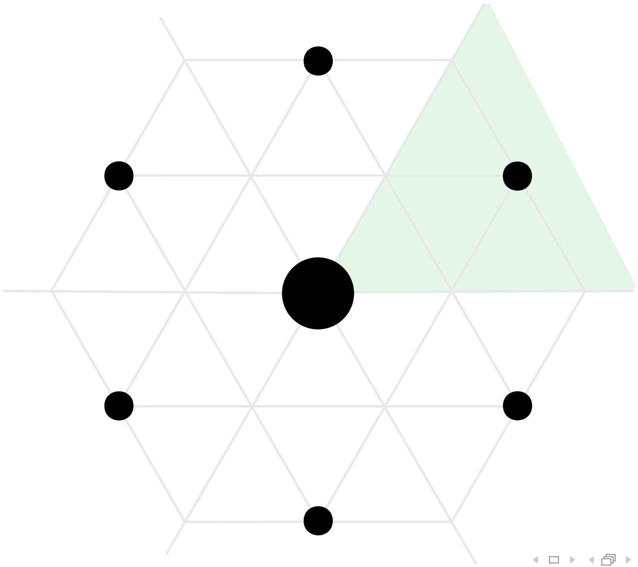
$\mathfrak{g} = \mathfrak{sl}_3$

Characters of representations

$\mathfrak{g} = \mathfrak{sl}_3$

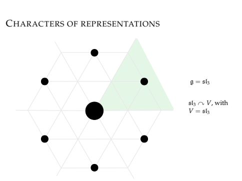
So, how can we motivate the definition of a crystal? On the screen is a picture of the weight lattice for $\mathfrak{sl}_3$, with the integral dominant weights being the lattice points in the green shaded area. For any $\mathfrak{sl}_3$ representation, we can plot its *character* on this lattice in the following way: we place a dot at each lattice point if the representation has a weight space there, with the size of the dot representing the dimension of the weight space.

Characters of representations



$\mathfrak{g} = \mathfrak{sl}_3$

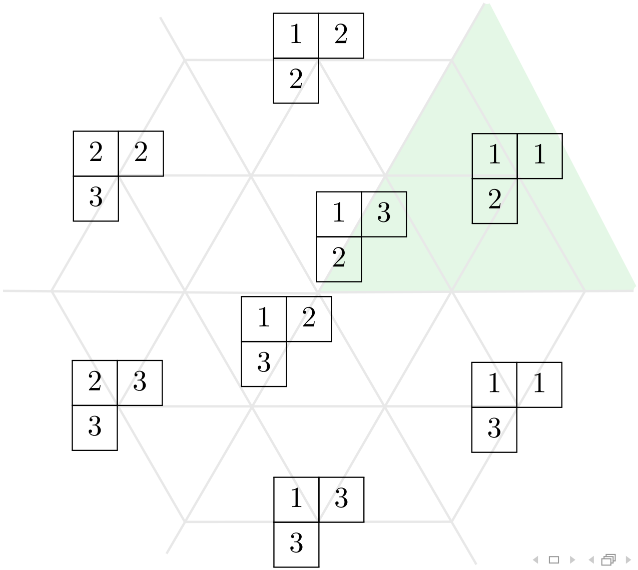
$\mathfrak{sl}_3 \curvearrowright V$, with
$V = \mathfrak{sl}_3$



For example, when we take the representation V to be the adjoint representation of $\mathfrak{sl}_3$, we get the following pattern of dots, where the zero-weight space is two-dimensional, and all others are one-dimensional.

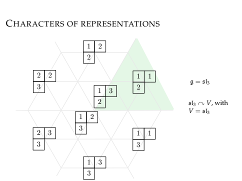
This data, being which weights appear in the representation with what multiplicity, is called the *character* of V, and determines V up to isomorphism. However, extracting the decomposition of V from its character is more or less a giant linear algebra exercise, roughly equivalent to decomposing a polynomial in terms of an interesting basis. So now we have two problems: finding this basis, and decomposing characters into this basis.

Characters of representations



$\mathfrak{g} = \mathfrak{sl}_3$

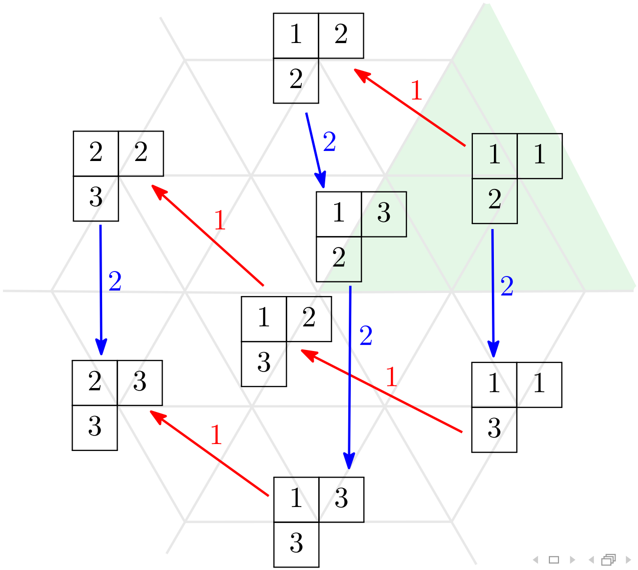
$\mathfrak{sl}_3 \curvearrowright V$, with $V = \mathfrak{sl}_3$



A first step towards solving these problems is to find some combinatorial set which gives the character as a weighted sum. A well-known example is the set of semistandard Young tableaux on a particular shape. Each of these tableaux have a certain weight, and together the weights of tableaux of this shape give the character for the adjoint representation. On the screen, I've placed each tableau over its weight, with the two tableaux in the centre having weight zero.

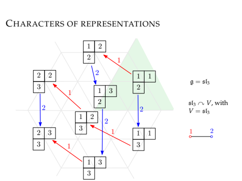
So at the moment we have a set, which is equipped with a weight function, where the character of the set is equal to the character of a $\mathfrak{g}$-module. This is *almost* a crystal, but we need one last crucial ingredient.

CHARACTERS OF REPRESENTATIONS



$\mathfrak{g} = \mathfrak{sl}_3$

$\mathfrak{sl}_3 \curvearrowright V$, with $V = \mathfrak{sl}_3$



Making this into a *crystal* means to make it into an edge-labelled graph, where edge labels correspond to nodes in the Dynkin diagram. Travelling along a 1-labelled edge will always subtract α_1 (the first simple root) from a weight, and similarly for the other simple roots. On the screen is a crystal for the adjoint representation of $\mathfrak{sl}_3$.

This directed graph is connected, which is the crystal analogue of the fact that the adjoint representation is irreducible. The "highest-weight element" is the element with no incoming edges, and the "lowest-weight element" is the element with no outgoing edges, so the crystal goes from high to low. The isomorphism class of the crystal is just the weight of the highest-weight element.

Decomposing a crystal means simply finding all of those elements which have no outgoing edges, and writing down their weights. This can be much easier than decomposing a polynomial in terms of a certain basis, especially when the rules for the edges are simple.

A $\mathfrak{g}$-CRYSTAL IS...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

A $\mathfrak{g}$-crystal is...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

The abstract notion of a $\mathfrak{g}$-crystal is a set, equipped with a weight function and some arrows, such that a long list of (fairly simple) axioms are satisfied. I think a good way to think of a $\mathfrak{g}$-crystal is as a *combinatorial shadow* of a representation. Where before you had a dimension n weight space, now you have n elements with that weight. The crystal arrows are morally replacing the f_i Chevalley generators. Knowing the decomposition of a crystal is just finding the highest-weight crystal elements, with no incoming arrows.

A $\mathfrak{g}$-CRYSTAL IS...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

$\mathfrak{g}$ crystals form a semisimple category, with simples indexed by dominant weights.

A $\mathfrak{g}$-crystal is...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

$\mathfrak{g}$ crystals form a semisimple category, with simples indexed by dominant weights.

Defining morphisms as morphisms of labelled graphs, the set of $\mathfrak{g}$-crystals forms a semisimple category, and the simple objects in this category are indexed by the same dominant weights as $\mathfrak{g}$-representations.

A $\mathfrak{g}$-CRYSTAL IS...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

$\mathfrak{g}$ crystals form a semisimple category, with simples indexed by dominant weights.

The category of crystals is *monoidal*: the underlying set of $C_1 \otimes C_2$ is $C_1 \times C_2$.

The decomposition numbers match those in $\mathfrak{g}$-mod:

$$[B(\nu) : B(\lambda) \otimes B(\mu)] = [V(\nu) : V(\lambda) \otimes V(\mu)]$$

A $\mathfrak{g}$-crystal is...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

$\mathfrak{g}$ crystals form a semisimple category, with simples indexed by dominant weights.

The category of crystals is *monoidal*: the underlying set of $C_1 \otimes C_2$ is $C_1 \times C_2$.

The decomposition numbers match those in $\mathfrak{g}$-mod:

$$[B(\nu) : B(\lambda) \otimes B(\mu)] = [V(\nu) : V(\lambda) \otimes V(\mu)]$$

Perhaps the most exciting thing about crystals is the fact that they have a tensor product rule. Given any two $\mathfrak{g}$-crystals, their Cartesian product set can be equipped with an $\mathfrak{g}$-crystal structure, and the tensor product rule is associative. Amazingly, the decomposition numbers of crystals matches up with the decomposition numbers of representations! You can get the Littlewood-Richardson rule in about a page of working by just knowing the crystal structure on tableaux, and the tensor product rule.

A $\mathfrak{g}$-CRYSTAL IS...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

$\mathfrak{g}$ crystals form a semisimple category, with simples indexed by dominant weights.

The category of crystals is *monoidal*: the underlying set of $C_1 \otimes C_2$ is $C_1 \times C_2$.

The decomposition numbers match those in $\mathfrak{g}$-mod:

$$[B(\nu) : B(\lambda) \otimes B(\mu)] = [V(\nu) : V(\lambda) \otimes V(\mu)]$$

... but there is no functor $\mathfrak{g}$-mod $\to$ $\mathfrak{g}$-crystals.

A $\mathfrak{g}$-crystal is...

A $\mathfrak{g}$-crystal is a *combinatorial shadow* of a $\mathfrak{g}$-representation.

$\mathfrak{g}$ crystals form a semisimple category, with simples indexed by dominant weights.

The category of crystals is *monoidal*: the underlying set of $C_1 \otimes C_2$ is $C_1 \times C_2$.

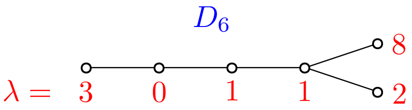
The decomposition numbers match those in $\mathfrak{g}$-mod:

$$[B(\nu) : B(\lambda) \otimes B(\mu)] = [V(\nu) : V(\lambda) \otimes V(\mu)]$$

... but there is no functor $\mathfrak{g}$-mod $\to$ $\mathfrak{g}$-crystals.

The "moral" of crystals is this: they give amazing combinatorial tools for dealing with problems in representation theory. However, it is important to understand that the correspondence between representations and crystals is not functorial, and there is no straightforward way to pass from a representation of $\mathfrak{g}$ to a $\mathfrak{g}$-crystal with the same decomposition.

Reminder: Nakajima quiver varieties



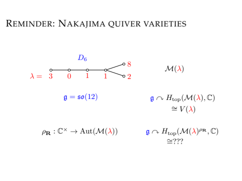
$\mathcal{M}(\lambda)$

$\mathfrak{g} = \mathfrak{so}(12)$

$\mathfrak{g} \curvearrowright H_{\text{top}}(\mathcal{M}(\lambda), \mathbb{C}) \cong V(\lambda)$

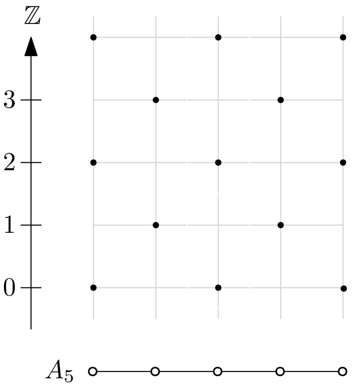
$\rho_{\mathbf{R}} : \mathbb{C}^{\times} \to \mathrm{Aut}(\mathcal{M}(\lambda))$

$\mathfrak{g} \curvearrowright H_{\text{top}}(\mathcal{M}(\lambda)^{\rho_{\mathbf{R}}}, \mathbb{C}) \cong ???$



Before I go any further, let me just relate crystals back to the Nakajima quiver varieties. Remember this setup we had, where I wanted to investigate that last representation which came out of the top homology of the fixed-point set under this action. By some miracle of geometric representation theory, the crystal for this representation is immediately explicit, and can be seen as irreducible components of the fixed-point set. It also has an explicit combinatorics, and so by investigating this crystal I can investigate that representation. The crystal that comes out is called the "product monomial crystal", and depends on both $\mathfrak{g}$ and the parameter $\mathbf{R}$.

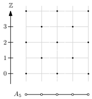
MONOMIAL CRYSTAL



Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$$

Monomial crystal

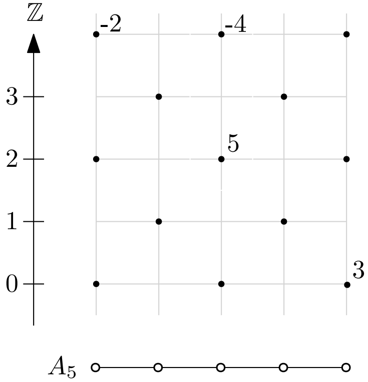


Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$

The product monomial crystal is defined as a subcrystal of a larger crystal, called the monomial crystal. I'll first define this larger crystal, which means I need to tell you what its elements are, what the weight of each element is, and how to calculate the f_i operator for each element. We need to fix a partition of the Dynkin diagram so that it becomes a bipartite graph — call the nodes on one side "odd", and on the other side "even". Next, we take $\mathbb{Z}$-many copies of the Dynkin diagram, and throw away half the points: the ones where the parities don't match. Call the set we are left with L.

MONOMIAL CRYSTAL

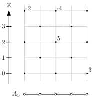


Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$$

The *monomial crystal* is the set $\mathbb{Z}L$.

Monomial crystal



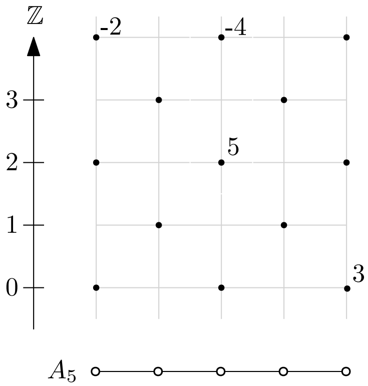
Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$

The *monomial crystal* is the set $\mathbb{Z}L$.

The elements of the monomial crystal are the elements of the free $\mathbb{Z}$-module on L, which we think of as finite assignments of integers to the points of L, leaving the rest of the points assigned to 0. On the left, I've chosen four points and assigned them some integers, so the picture on the left is some random element of the monomial crystal.

Monomial crystal

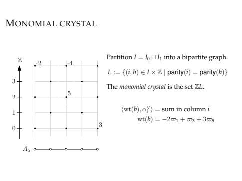


Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$$

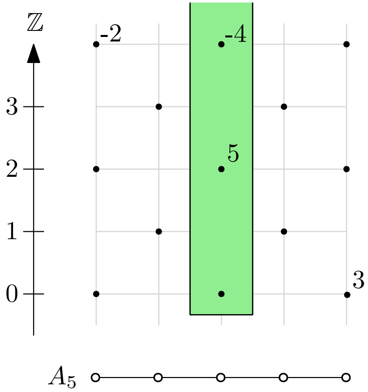
The *monomial crystal* is the set $\mathbb{Z}L$.

$$\langle \mathrm{wt}(b), \alpha_i^\vee \rangle = \text{sum in column } i$$
$$\mathrm{wt}(b) = -2\varpi_1 + \varpi_3 + 3\varpi_5$$



The weight of an element is given in terms of the fundamental weights, where the coefficient of the ith fundamental weight is the sum in column i. The f_i operator is more complicated, so I will show an example to compute f_3 of the element on the left.

Monomial crystal



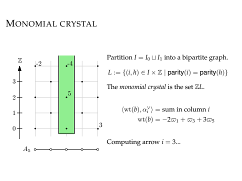
Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i,h) \in I \times \mathbb{Z} \mid \text{parity}(i) = \text{parity}(h)\}$$

The *monomial crystal* is the set $\mathbb{Z}L$.

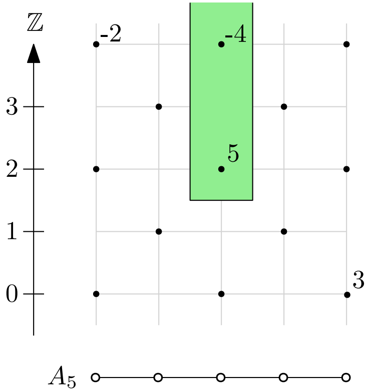
$$\langle \mathrm{wt}(b), \alpha_i^\vee \rangle = \text{sum in column } i$$
$$\mathrm{wt}(b) = -2\varpi_1 + \varpi_3 + 3\varpi_5$$

Computing arrow $i = 3$...



Firstly, we check if there is any upper half-column with a positive sum. The half-column on the left sums to 1, and so f_3 will act on our monomial.

MONOMIAL CRYSTAL



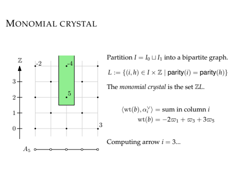
Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$$

The *monomial crystal* is the set $\mathbb{Z}L$.

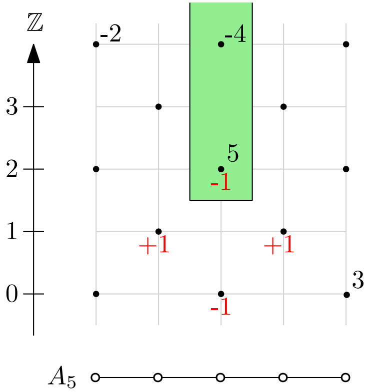
$$\langle \mathrm{wt}(b), \alpha_i^\vee \rangle = \text{sum in column } i$$
$$\mathrm{wt}(b) = -2\varpi_1 + \varpi_3 + 3\varpi_5$$

Computing arrow $i = 3$...



Next, we find the *smallest* upper half-column achieving the *maximum* half-column sum. (One way to justify this is that the f_i operator must be invertible, so it always needs to act on the "edge" of some piece of data.)

Monomial crystal



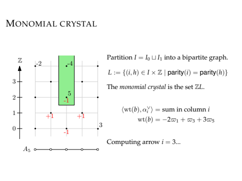
Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i, h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$$

The *monomial crystal* is the set $\mathbb{Z}L$.

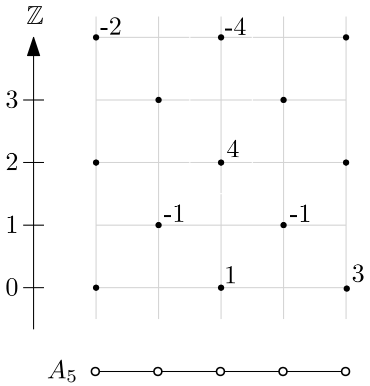
$$\langle \mathrm{wt}(b), \alpha_i^\vee \rangle = \text{sum in column } i$$
$$\mathrm{wt}(b) = -2\varpi_1 + \varpi_3 + 3\varpi_5$$

Computing arrow $i = 3$...



We then add a "gadget" of weight $-\alpha_i$ below that half-column, to obtain a new monomial.

MONOMIAL CRYSTAL



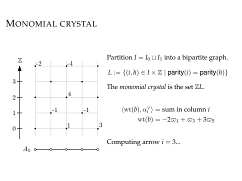
Partition $I = I_0 \sqcup I_1$ into a bipartite graph.

$$L := \{(i,h) \in I \times \mathbb{Z} \mid \mathsf{parity}(i) = \mathsf{parity}(h)\}$$

The *monomial crystal* is the set $\mathbb{Z}L$.

$$\langle \mathrm{wt}(b), \alpha_i^\vee \rangle = \text{sum in column } i$$
$$\mathrm{wt}(b) = -2\varpi_1 + \varpi_3 + 3\varpi_5$$

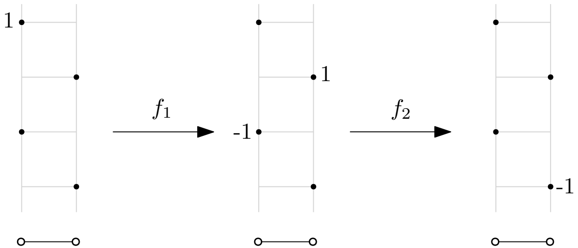
Computing arrow $i = 3$...



The e_i operator has a very similar definition in terms of lower half-columns. In practice, it is easier to remember which gadgets were added, rather than remembering the resulting labelling of points.

FUNDAMENTAL MONOMIAL CRYSTALS

The crystal generated by $(i, c) \in L$ is a *fundamental* crystal, written $B(i, c)$.



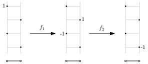
The basic crystal $B(1, c)$ in type A_2.

Theorem (Kashiwara)

The crystal $B(i, c)$ is isomorphic to $B(\varpi_i)$, the irreducible crystal of highest weight ϖ_i.

FUNDAMENTAL MONOMIAL CRYSTALS

The crystal generated by $(i, c) \in L$ is a *fundamental* crystal, written $B(i, c)$.



The basic crystal $B(1, c)$ in type A_2.

Theorem (Kashiwara)

The crystal $B(i, c)$ is isomorphic to $B(\varpi_i)$, the irreducible crystal of highest weight ϖ_i.

The whole monomial crystal is very large, and we don't consider it all at once. If we pick a single point (i, c) and assign it the value 1, then the crystal generated by this point is called the *basic* crystal at $B(i, c)$. Here I've shown an $\mathfrak{sl}_3$ example of a basic crystal, beginning with a point in the first column. It's easy to apply the f_1 and f_2 operators to verify that the crystal generated by this element has size 3, and is isomorphic to the first fundamental crystal of $\mathfrak{sl}_3$. A theorem of Kashiwara states that every subcrystal generated in this way will be isomorphic to a crystal of fundamental weight.

The product monomial crystal

Let $\mathbf{R} = \{(i_1, c_1), \ldots, (i_r, c_r)\}$ be a multiset.

- Each $B(i_k, c_k) \subseteq \mathbb{Z}L$ is a finite crystal isomorphic to $B(\varpi_{i_k})$.
- Let $B(\mathbf{R}) \subseteq \mathbb{Z}L$ be their sum:

$$B(\mathbf{R}) = \{b_1 + \cdots + b_r \mid b_k \in B(i_k, c_k)\}$$

- Redundancies may occur: $|B(\mathbf{R})| \leq |B(i_1, c_1)| \cdots |B(i_r, c_r)|$

> *Theorem (Kamnitzer, Tingley, Webster, Weekes, Yacobi)*
>
> $B(\mathbf{R})$ is a subcrystal of $\mathbb{Z}L$.

The crystal $B(\mathbf{R})$ is called the *product monomial crystal* associated to the data $\mathbf{R}$.

The product monomial crystal

Let $\mathbf{R} = \{(i_1, c_1), \dots, (i_r, c_r)\}$ be a multiset.

- Each $B(i_k, c_k) \subseteq \mathbb{Z}L$ is a finite crystal isomorphic to $B(\varpi_{i_k})$.
- Let $B(\mathbf{R}) \subseteq \mathbb{Z}L$ be their sum:

$$B(\mathbf{R}) = \{b_1 + \cdots + b_r \mid b_k \in B(i_k, c_k)\}$$

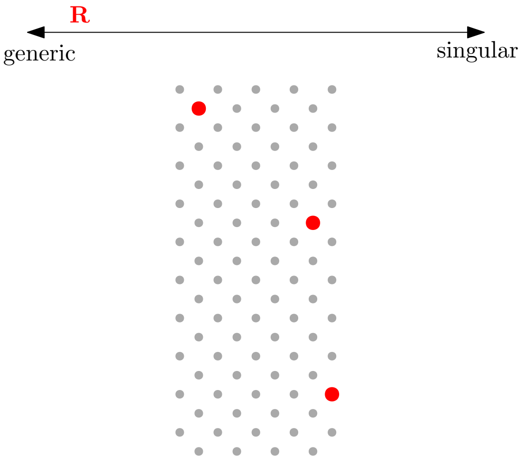
- Redundancies may occur: $|B(\mathbf{R})| \leq |B(i_1, c_1)| \cdots |B(i_r, c_r)|$

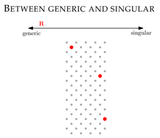
> *Theorem (Kamnitzer, Tingley, Webster, Weekes, Yacobi)*
>
> $B(\mathbf{R})$ is a subcrystal of $\mathbb{Z}L$.

The crystal $B(\mathbf{R})$ is called the *product monomial crystal* associated to the data $\mathbf{R}$.

Now we know what fundamental subcrystals are, we can define the product monomial crystal. It is parametrised by some multiset of points belonging to the lattice L. For each point, we generate the fundamental subcrystal associated to that point. We then take their sum, using the fact that they are all elements belonging to the monomial crystal $\mathbb{Z}L$. This is a strange thing to do, and is not a meaningful crystal operation. However, it turns out that the resulting set is always a subcrystal of the monomial crystal, finite because each of the fundamental crystals were.

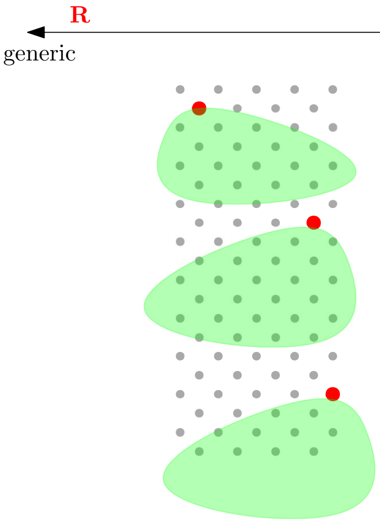
Between generic and singular



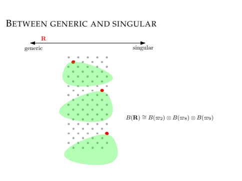


Depending on the parameter **R**, the product monomial crystal interpolates between generic and singular cases, and its behaviour in these extreme cases is easy to see. A "generic" **R**-parameter is when the points are far apart vertically.

BETWEEN GENERIC AND SINGULAR

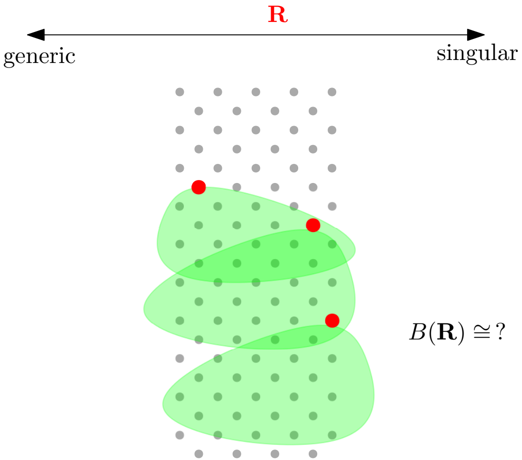


$$B(\mathbf{R}) \cong B(\varpi_2) \otimes B(\varpi_8) \otimes B(\varpi_9)$$

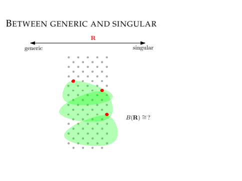


The subcrystal generated by the top red dot can only have coefficients lying in some bounded area below the red dot, which I've pictured here using the green blob. You can see that given any labelling of the lattice, it is clear which red dot each label must belong to, and so there is no redundancy, and we get the full tensor product.

BETWEEN GENERIC AND SINGULAR

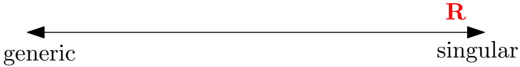


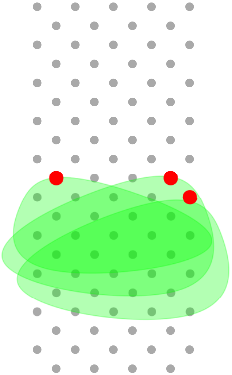
$B(\mathbf{R}) \cong ?$



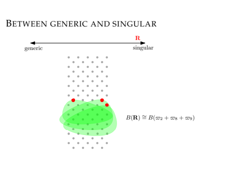
As we push the red dots closer together, we may get some redundancies where the fundamental crystals overlap. This is the difficult case.

BETWEEN GENERIC AND SINGULAR





$$B(\mathbf{R}) \cong B(\varpi_2 + \varpi_8 + \varpi_9)$$



Finally, when the red dots are pushed even closer together, we get maximum redundancy. In this case it's easy to explicitly find the single highest-weight element of the crystal, and conclude that the crystal must be irreducible, and isomorphic to the sum of fundamental weights.

There is a natural question: can we describe $B(\mathbf{R})$ for any parameter multiset $\mathbf{R}$?

My contributions

Natural question: can we describe $B(\mathbf{R})$ for arbitrary $\mathbf{R}$?

> *Theorem (G, 2018)*
>
> In any simply-laced type, there is a Demazure-type formula giving the character of $B(\mathbf{R})$. This formula consists of Demazure operators π_i, and multiplications by the fundamental weights ϖ_i.

The character formula is proved using a novel method for analysing $B(\mathbf{R})$ through *Demazure truncations*.

Natural question: can we describe $B(\mathbf{R})$ for arbitrary $\mathbf{R}$?

Theorem (G, 2018)

In any simply-laced type, there is a Demazure-type formula giving the character of $B(\mathbf{R})$. This formula consists of Demazure operators π_i, and multiplications by the fundamental weights ϖ_i.

The character formula is proved using a novel method for analysing $B(\mathbf{R})$ through *Demazure truncations*.

Given any multiset $\mathbf{R}$, there is a Demazure-type formula for the character of the crystal $B(\mathbf{R})$. How this formula is proven is by defining a family of smaller subsets of product monomial crystals, which are closed under the e_i operators, but not necessarily the f_i operators. (Following arrows backwards is always fine, but following them forwards may land you outside the subset).

These smaller subsets are related by either extension-of-i-strings, which acts as the Demazure operator π_i on characters, or by including another element of the parameter multiset $\mathbf{R}$, which acts as multiplication by e^{ϖ_i} on characters. A path from the trivial crystal $\{1\}$ to $B(\mathbf{R})$ then gives a character formula.

I don't have time to go into the proof now, but I will discuss some interesting results in type A. (Everything so far holds in any simply-laced type.) In type A, I can use this character formula along with known results about generalised Schur modules to show that the product monomial crystal always decomposes in the same way as a generalised Schur module, which gives amonst other things a new combinatorial formula for the characters of these modules.

Schur functors

λ a partition, $\mathbb{S}_\lambda : \mathsf{Vect}_\mathbb{C} \to \mathsf{Vect}_\mathbb{C}$ a "Schur functor".

$\mathbb{S}_\lambda(V)$ is the image of d_λ:

$$d_\lambda : \mathrm{Alt}^{\mathrm{cols}\,\lambda}(V) \xrightarrow{\text{comult}} V^{\otimes\lambda} \xrightarrow{\text{mult}} \mathrm{Sym}^{\mathrm{rows}\,\lambda}(V)$$

Schur functors

λ a partition, $\mathbb{S}_\lambda : \mathsf{Vect}_\mathbb{C} \to \mathsf{Vect}_\mathbb{C}$ a "Schur functor".

$\mathbb{S}_\lambda(V)$ is the image of d_λ:

$$d_\lambda : \mathrm{Alt}^{\mathrm{cols}\,\lambda}(V) \xrightarrow{\mathsf{comult}} V^{\otimes\lambda} \xrightarrow{\mathsf{mult}} \mathrm{Sym}^{\mathrm{rows}\,\lambda}(V)$$

Let's switch gears for a second and talk about some classical constructions in type A. There are some natural endofunctors of Vect called *Schur functors*. They are parametrised by partitions, and are roughly "antisymmetrise along rows, while symmetrising along columns". Given a vector space V, for each column of λ we form an exterior power, and for each row of λ we form a symmetric power. Then we make a map d_λ from the tensor product of exterior powers into the tensor product of symmetric powers, by fully comultiplying on the exterior side, using the shape of λ, then fully multiplying on the symmetric side.

SCHUR FUNCTORS

λ a partition, $\mathbb{S}_\lambda : \mathsf{Vect}_\mathbb{C} \to \mathsf{Vect}_\mathbb{C}$ a "Schur functor".

$\mathbb{S}_\lambda(V)$ is the image of d_λ:

$$d_\lambda : \mathrm{Alt}^{\mathrm{cols}\ \lambda}(V) \xrightarrow{\mathsf{comult}} V^{\otimes\lambda} \xrightarrow{\mathsf{mult}} \mathrm{Sym}^{\mathrm{rows}\ \lambda}(V)$$

For $\lambda = (3,1)$,

$$d_\lambda : \bigwedge^2(V) \otimes \bigwedge^1(V) \otimes \bigwedge^1(V) \to S^3(V) \otimes S^1(V)$$

$$(v_1 \wedge v_2) \otimes v_3 \otimes v_4 \mapsto \begin{array}{|c|c|c|}\hline v_1 & v_3 & v_4 \\ \hline v_2 & & \\ \hline \end{array} - \begin{array}{|c|c|c|}\hline v_2 & v_3 & v_4 \\ \hline v_1 & & \\ \hline \end{array} \mapsto v_1v_3v_4 \otimes v_2 - v_2v_3v_4 \otimes v_1$$

Schur functors

λ a partition, $\mathbb{S}_\lambda : \mathsf{Vect}_\mathbb{C} \to \mathsf{Vect}_\mathbb{C}$ a "Schur functor".

$\mathbb{S}_\lambda(V)$ is the image of d_λ:

$$d_\lambda : \mathrm{Alt}^{\mathrm{cols}\,\lambda}(V) \xrightarrow{\mathsf{comult}} V^{\otimes\lambda} \xrightarrow{\mathsf{mult}} \mathrm{Sym}^{\mathrm{rows}\,\lambda}(V)$$

For $\lambda = (3,1)$,

$$d_\lambda : \bigwedge^2(V) \otimes \bigwedge^1(V) \otimes \bigwedge^1(V) \to S^3(V) \otimes S^1(V)$$

$$(v_1 \wedge v_2) \otimes v_3 \otimes v_4 \mapsto \begin{array}{|c|c|c|}\hline v_1 & v_3 & v_4 \\ \hline v_2 \\ \cline{1-1}\end{array} - \begin{array}{|c|c|c|}\hline v_2 & v_3 & v_4 \\ \hline v_1 \\ \cline{1-1}\end{array} \mapsto v_1v_3v_4 \otimes v_2 - v_2v_3v_4 \otimes v_1$$

Here is an example of the map d_λ for the partition $(3,1)$. We take a vector in the tensor product of exterior powers, and fully comultiply — in this case, we just antisymmetrise the first pair. Then we think of pure tensors in the middle as tableaux, and multiply along rows into the tensor product of symmetric powers.

(Generalised) Schur modules

By functorality, $G \curvearrowright V \implies G \curvearrowright \mathbb{S}_\lambda(V)$

When $G = \mathrm{GL}_n(\mathbb{C})$, the $\mathbb{S}_\lambda(\mathbb{C}^n)$ is called the *Schur module* for λ.

(Generalised) Schur modules

By functorality, $G \curvearrowright V \implies G \curvearrowright \mathbb{S}_\lambda(V)$

When $G = \mathrm{GL}_n(\mathbb{C})$, the $\mathbb{S}_\lambda(\mathbb{C}^n)$ is called the *Schur module* for λ.

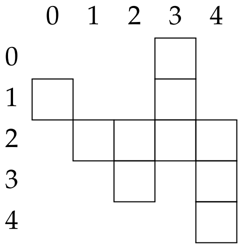
If V was originally a representation for some group G, then $\mathbb{S}_\lambda(V)$ is also a G-representation, just by sending every automorphism over the functor. So for the group GL_n, we can send the basic representation through the Schur functor, to get what is called a *Schur module*. It is well-known that as λ varies over partitions with at most n rows, the Schur modules vary over all irreducible GL_n representations.

(GENERALISED) SCHUR MODULES

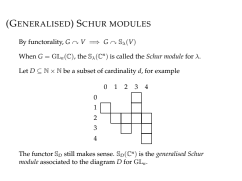
By functorality, $G \curvearrowright V \implies G \curvearrowright \mathbb{S}_\lambda(V)$

When $G = \mathrm{GL}_n(\mathbb{C})$, the $\mathbb{S}_\lambda(\mathbb{C}^n)$ is called the *Schur module* for λ.

Let $D \subseteq \mathbb{N} \times \mathbb{N}$ be a subset of cardinality d, for example



The functor $\mathbb{S}_D$ still makes sense. $\mathbb{S}_D(\mathbb{C}^n)$ is the *generalised Schur module* associated to the diagram D for GL_n.



What is less well-known is that this whole definition still makes sense when we replace the partition λ by an arbitrary diagram, meaning a finite subset of lattice points in the plane. In this case, we call the functor applied to the vector representation a "generalised Schur module". It's quite easy to see that the generalised Schur module is invariant under column permutations of D, and its isomorphism class is invariant under row permutations of D. Some of these might be more familiar, for example when D is a skew shape, then you get the expected decomposition out the end, corresponding to what you might expect from skew Schur functions in the symmetric algebra.

Crystal of generalised Schur modules

$\mathbb{S}_D(\mathbb{C}^n)$ is an $\mathfrak{sl}_n$-module: what is its crystal?

- GL_n-character of $\mathbb{S}_D(\mathbb{C}^n)$: Magyar, Reiner, Shimozono (1990s).

> *Theorem (G, 2018)*
>
> In type A, the crystal $B(\mathbf{R})$ is the crystal of a generalised Schur module, for a diagram D depending on $\mathbf{R}$. Conversely, this gives the crystal of every generalised Schur module for a column-convex diagram.

2019-01-17

Crystal of generalised Schur modules

$\mathbb{S}_D(\mathbb{C}^n)$ is an $\mathfrak{sl}_n$-module: what is its crystal?

- GL_n-character of $\mathbb{S}_D(\mathbb{C}^n)$: Magyar, Reiner, Shimozono (1990s).

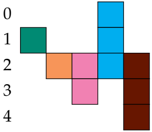
Theorem (G, 2018)

In type A, the crystal $B(\mathbf{R})$ is the crystal of a generalised Schur module, for a diagram D depending on $\mathbf{R}$. Conversely, this gives the crystal of every generalised Schur module for a column-convex diagram.

As I remarked earlier, finding the crystal of a given representation is usually a difficult task. In the case of the generalised Schur module, the work of Peter Magyar, Victor Reiner, and Mark Shimozono around the end of the 1990s produced a character formula for the generalised Schur module when n is sufficiently large. Using this character formula and my own, I was able to show that (provided the columns of D have no gaps, i.e. it is *column-convex*) there is a way to take the defining diagram D and send it to a multiset R, such that the product monomial crystal $B(\mathbf{R})$ is the crystal of the generalised Schur module. This gives, to my best knowledge, the first explicit crystal for these modules, and therefore the first combinatorial character formula for these modules.

Correspondence of diagrams and multisets

1. Diagram D



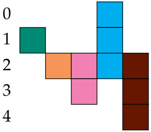
Correspondence of diagrams and multisets

1. Diagram D

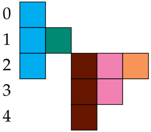
I'll finish by showing an example of the correspondence between the **R** multisets, and generalised column-convex diagrams. We start with a diagram.

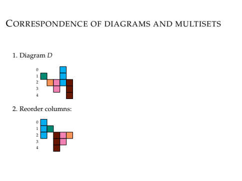
Correspondence of diagrams and multisets

1. Diagram D



2. Reorder columns:

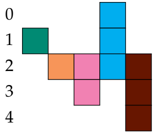




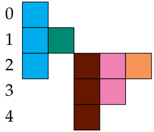
We re-order columns so that the columns with higher top boxes come first, for aesthetic reasons. The Schur functor of the rearranged diagram is equal to the Schur functor of the original diagram, since we are just precomposing d_λ with an automorphism of the tensor product of exterior powers.

CORRESPONDENCE OF DIAGRAMS AND MULTISETS

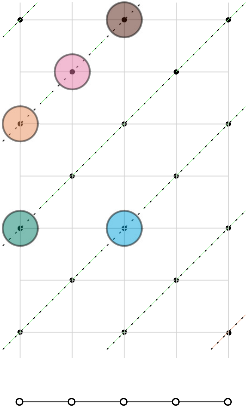
1. Diagram D

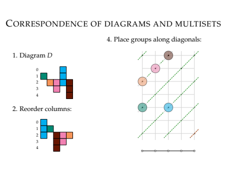


2. Reorder columns:



4. Place groups along diagonals:

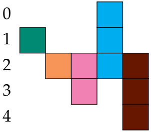




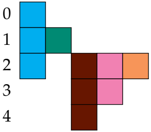
Then for each group of columns at a certain height, we “place them along the diagonal”. The first group is placed along the lowest diagonal, with the blue column of height 3 becoming a vertex over the node 3 in the Dynkin diagram. Similarly for the second group. The third group has three columns, and they are placed over their respective heights, all along the third diagonal.

Correspondence of diagrams and multisets

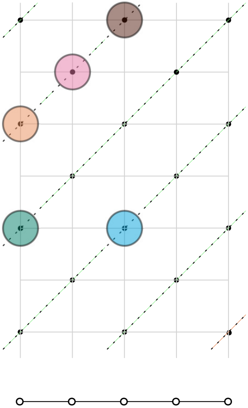
1. Diagram D



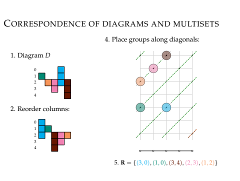
2. Reorder columns:



4. Place groups along diagonals:



5. $\mathbf{R} = \{(3,0), (1,0), (3,4), (2,3), (1,2)\}$



This gives us the corresponding **R** multiset.

We can notice some of the symmetries here: both the diagram and the multiset are invariant under translation up and down, and so only the relative heights matter. Interchanging the columns of D does not change the Schur functor, while the multiset has no order at all.

This whole process could be run backwards, to take a multiset **R** and produce a column-convex diagram D, up to column rearrangement, and vertical translation.

FUTURE DIRECTIONS

1. Truncations could apply to other monomial crystals.
2. Similar results should hold for simply-laced bipartite Kac-Moody types.
3. Do the truncations have a deeper meaning?

2019-01-17

1. Truncations could apply to other monomial crystals.
2. Similar results should hold for simply-laced bipartite Kac-Moody types.
3. Do the truncations have a deeper meaning?

There are some ways these results could be expanded on. Firstly, the method of analysing the crystal is fairly robust, and could possibly be applied to the "other" type of monomial crystal which appears in Kashiwara's paper, or other subcrystals of this monomial crystal. Secondly, a lot of this work should apply in arbitrary simply-laced Kac-moody type, provided that the Dynkin diagram is bipartite. Lastly, the truncations I mentioned are all examples of Demazure crystals, which are quite special. It would be interesting to investigate what these Demazure crystals correspond to on the quiver variety side.