Mutations of Puzzles and Equivariant cohomology of two-step flag varieties

Anders Skovsted Buch
Rutgers University

arXiv:1401.3065

Two-step flag varieties

$X \;=\; \mathrm{Fl}(a,b;n) \;=\; \{(A,B) \mid A \subset B \subset \mathbb{C}^n\,;\, \dim(A)=a\,;\, \dim(B)=b\}$

Def: A **012-string** for X is a permutation of $0^a\,1^{b-a}\,2^{n-b}$.

Example: $u = 10212$ is a 012-string for $\mathrm{Fl}(1,3;5)$.

$\mathbb{C}^n$ has basis $\{e_1, e_2, \ldots, e_n\}$. $\quad u = (u_1, u_2, \ldots, u_n)$ 012-string.

Set $A_u = \mathrm{Span}\{e_i : u_i = 0\}$ and $B_u = \mathrm{Span}\{e_i : u_i \leq 1\}$.

Schubert variety: $X^u = \overline{\mathbf{B}.(A_u, B_u)}$; $\mathbf{B} \subset \mathrm{GL}(\mathbb{C}^n)$ lower triangular.

$\mathrm{codim}(X^u, X) \;=\; \ell(u) \;=\; \#\{i < j \mid u_i > u_j\}$

Equivariant cohomology

$T \subset \mathrm{GL}(\mathbb{C}^n)$ maximal torus of diagonal matrices.

$H_T^*(\text{point}) \;=\; \mathbb{Z}[y_1, \ldots, y_n]$, where $y_i = -c_1(\mathbb{C}e_i)$.

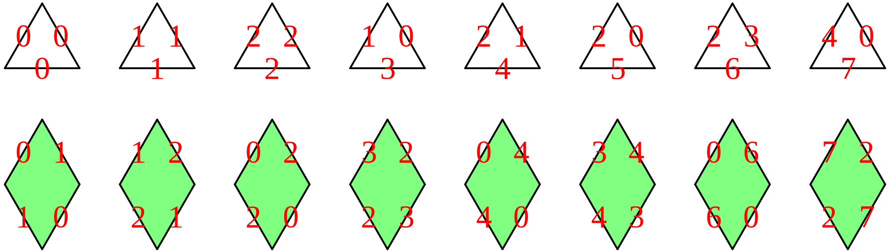
$H_T^*(X) \;=\; \bigoplus_u \mathbb{Z}[y_1, \ldots, y_n] \cdot [X^u]$ is an algebra over $H_T^*(\text{point})$.

The **equivariant Schubert structure constants** of X
are the classes $C_{u,v}^w \in \mathbb{Z}[y_1, \ldots, y_n]$ defined by

$$[X^u] \cdot [X^v] \;=\; \sum_w C_{u,v}^w \, [X^w]$$

Theorem (Graham) $C_{u,v}^w \;\in\; \mathbb{Z}_{\geq 0}[y_2 - y_1, \ldots, y_n - y_{n-1}]$

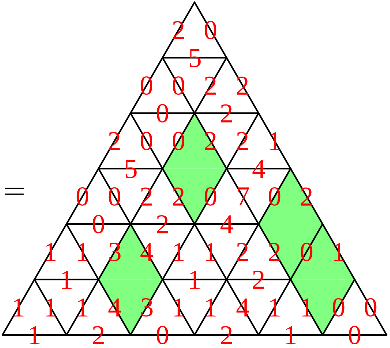
Puzzle pieces

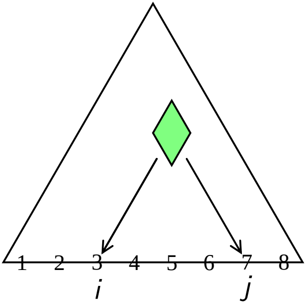


Simple labels: 0 , 1 , 2

Composed labels: $3 = 10$, $4 = 21$, $5 = 20$, $6 = 2(10)$, $7 = (21)0$

Equivariant puzzles

$P =$ 



weight(◊) $= y_j - y_i$

Boundary: $\partial P = \triangle_w^{u,v}$ where $u = 110202$, $v = 021210$, $w = 120210$.

Theorem: $C_{u,v}^w = \sum_{\partial P = \triangle_w^{u,v}} \prod_{◊ \in P}$ weight(◊)

Known cases:

Puzzle rule for $H^*(\mathrm{Gr}(m,n))$ (Knutson, Tao, Woodward)

Puzzle rule for $H_T^*(\mathrm{Gr}(m,n))$ (Knutson, Tao)

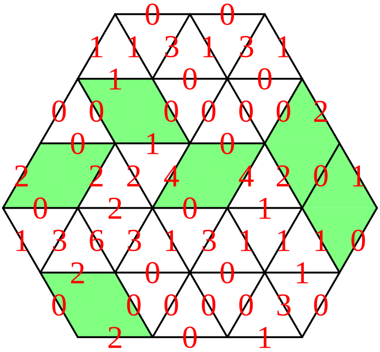
Puzzle rule for $H^*(\mathrm{Fl}(a,b;n))$ (conjectured by Knutson,
proof in [B-Kresch-Purbhoo-Tamvakis])

Consequence:

Equivariant quantum Littlewood-Richardson rule for $QH_T(\mathrm{Gr}(m,n))$.

This uses equivariant version (B, Mihalcea) of the quantum equals classical result (B, Kresch, Tamvakis).

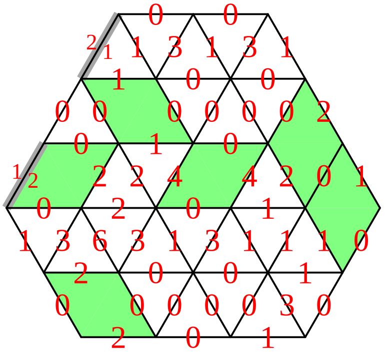
The mutation algorithm



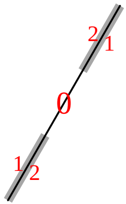
Puzzle:

- Shape is a hexagon.
- All pieces may be rotated.
- Boundary labels are simple.

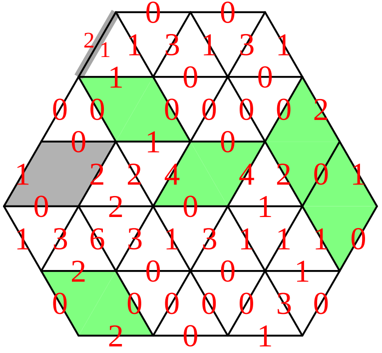
The mutation algorithm



Flawed puzzle containing the **gash pair**:

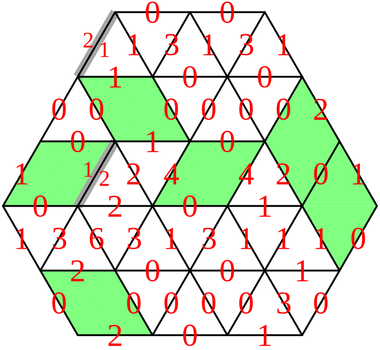


The mutation algorithm

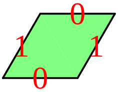


Remove problematic piece.

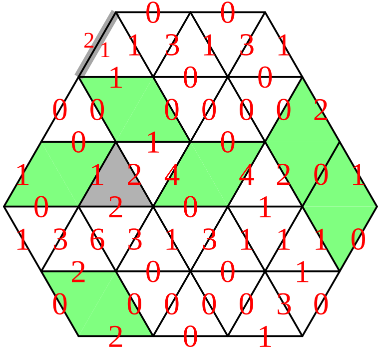
The mutation algorithm

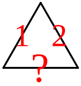


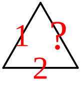
Replace with:



The mutation algorithm

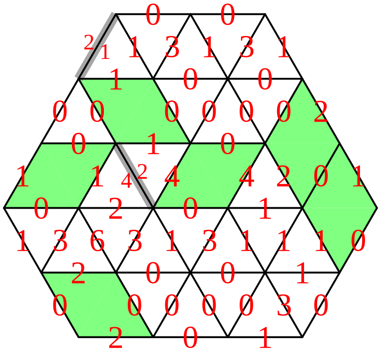


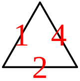
Replace with 

OR 

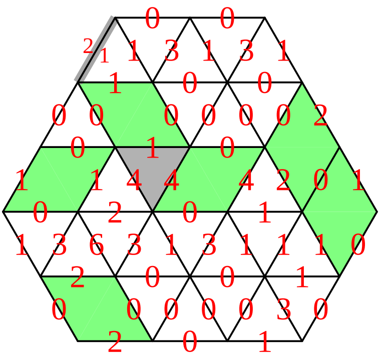
?

The mutation algorithm



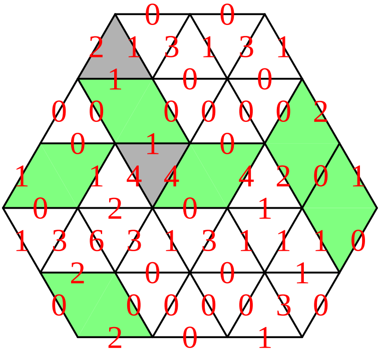
The piece  fits. Always at most one choice !!!

The mutation algorithm

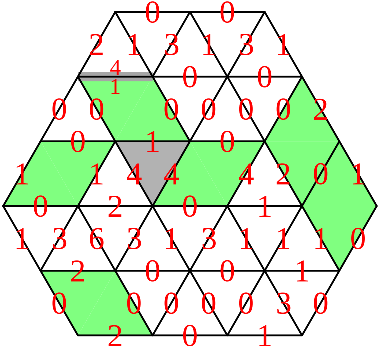


But no puzzle piece fits this time.

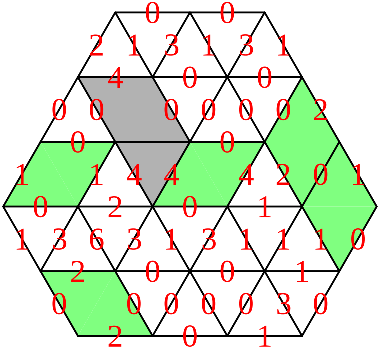
The mutation algorithm



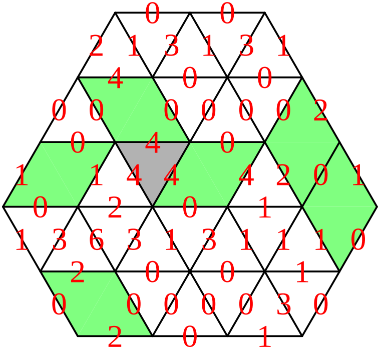
The mutation algorithm



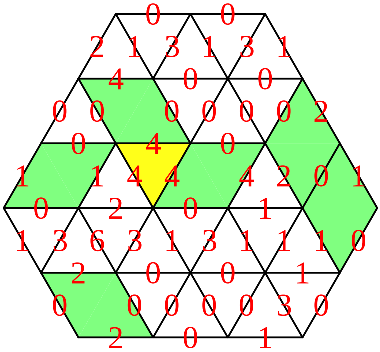
The mutation algorithm

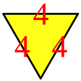


The mutation algorithm

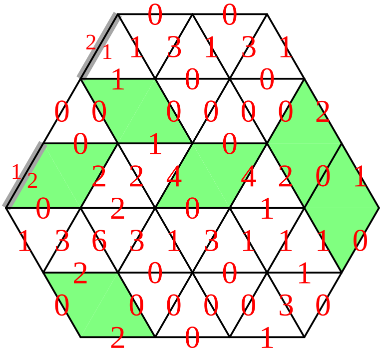


The mutation algorithm

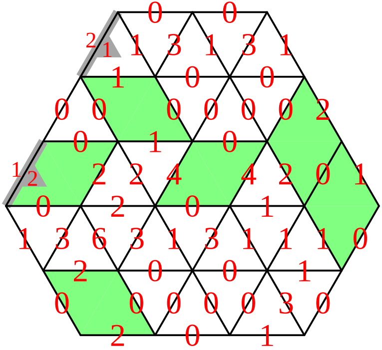


Flawed puzzle containing the **illegal puzzle piece**: 

The mutation algorithm

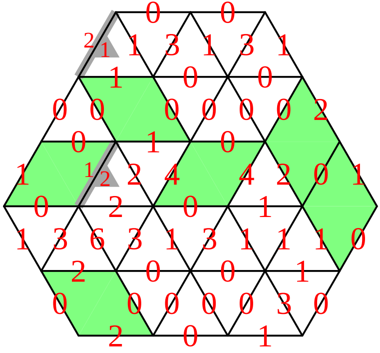


The mutation algorithm

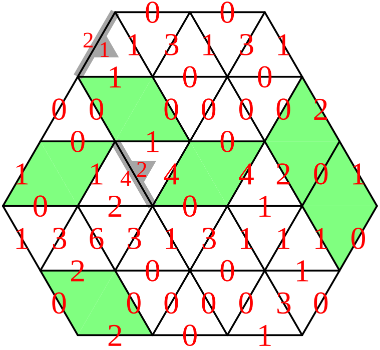


Use **directed gashes**.

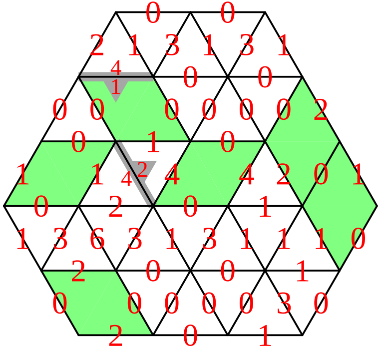
The mutation algorithm



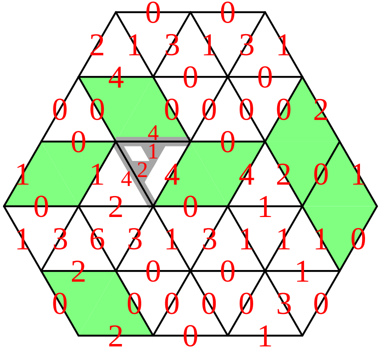
The mutation algorithm



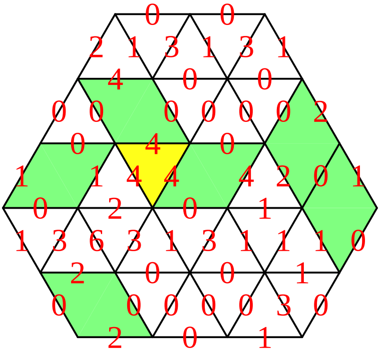
The mutation algorithm



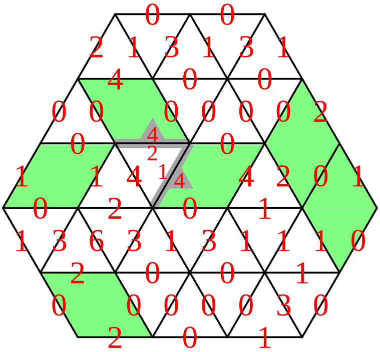
The mutation algorithm



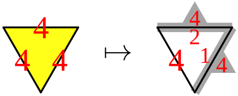
The mutation algorithm



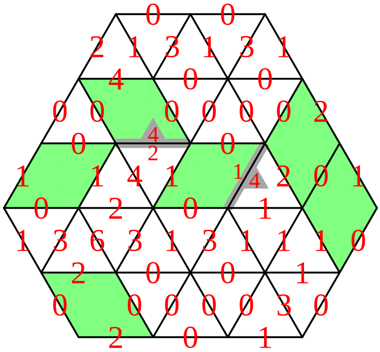
The mutation algorithm



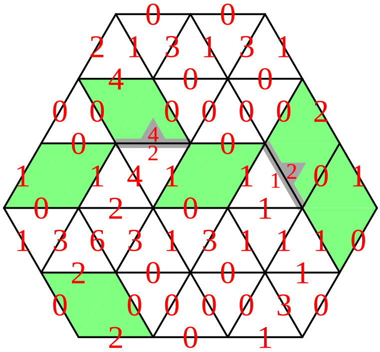
Resolution:



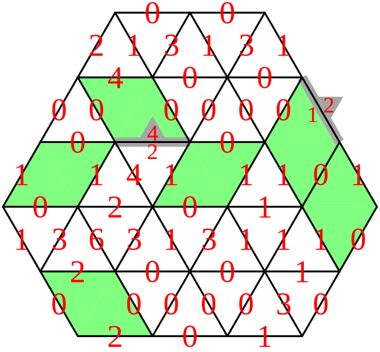
The mutation algorithm



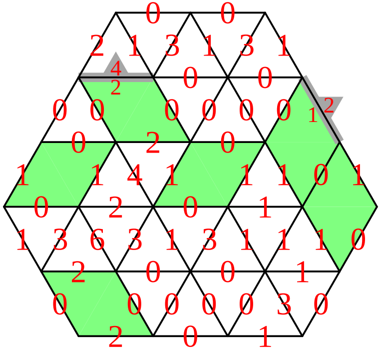
The mutation algorithm



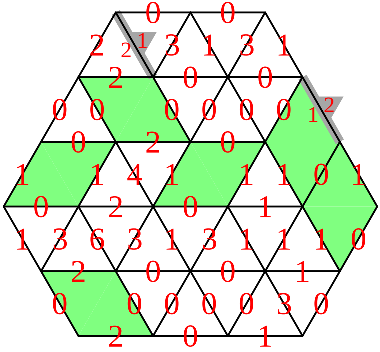
The mutation algorithm



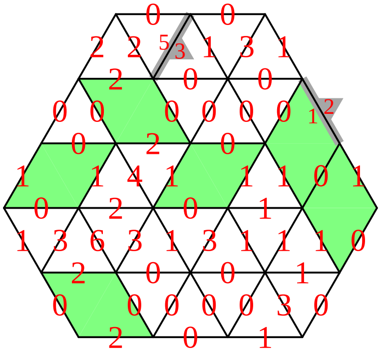
The mutation algorithm



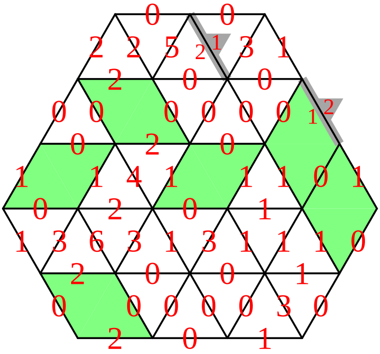
The mutation algorithm



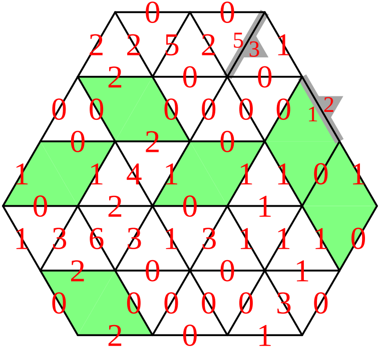
The mutation algorithm



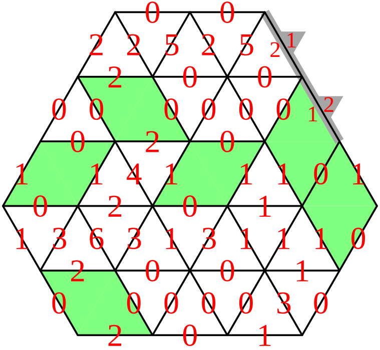
The mutation algorithm



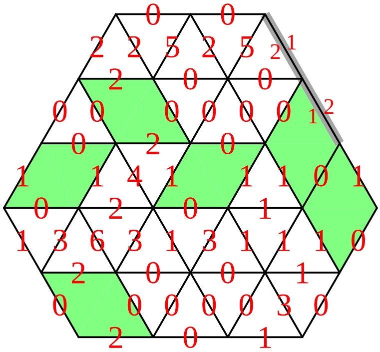
The mutation algorithm



The mutation algorithm

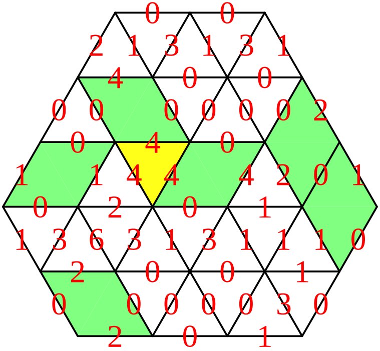


The mutation algorithm

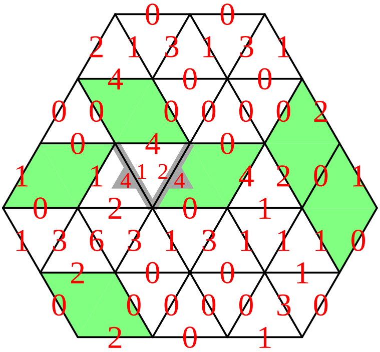


Flawed puzzle containing a gash pair.

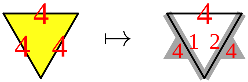
The mutation algorithm



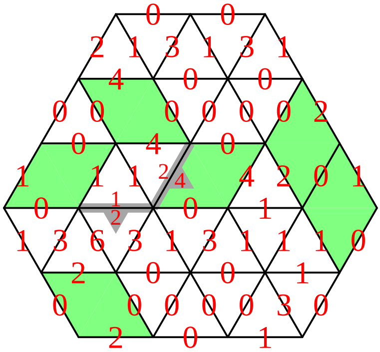
The mutation algorithm



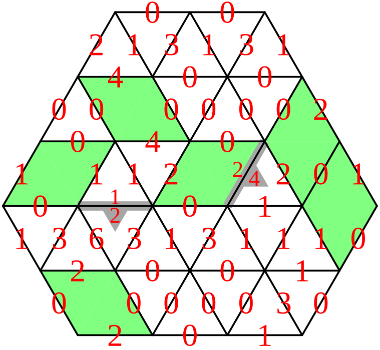
Resolution:



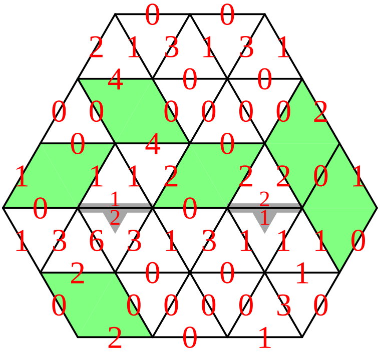
The mutation algorithm



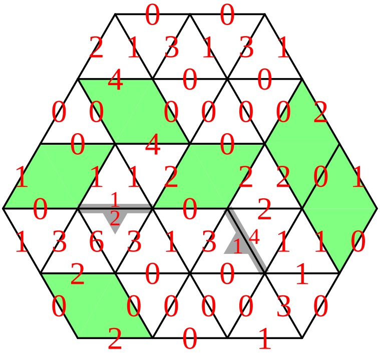
The mutation algorithm



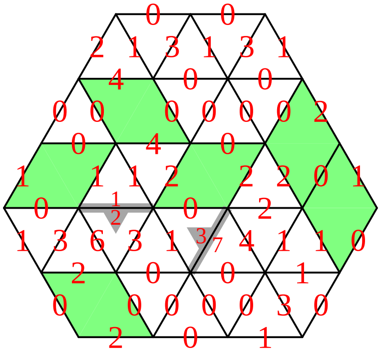
The mutation algorithm



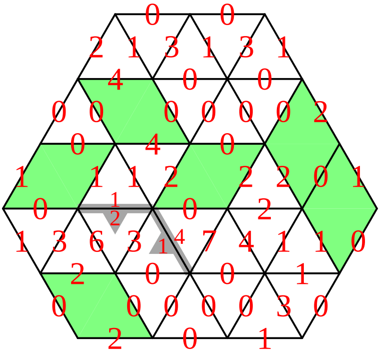
The mutation algorithm



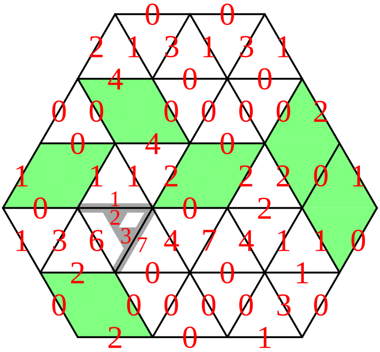
The mutation algorithm



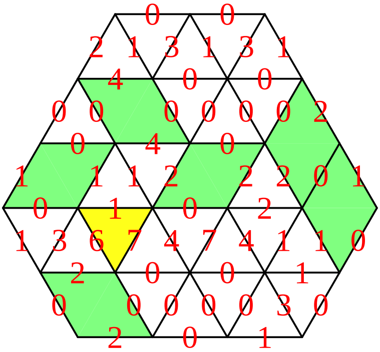
The mutation algorithm

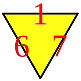


The mutation algorithm

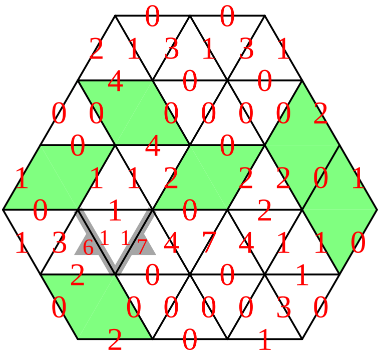


The mutation algorithm

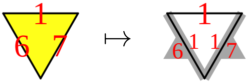


Flawed puzzle containing the **illegal puzzle piece**: 

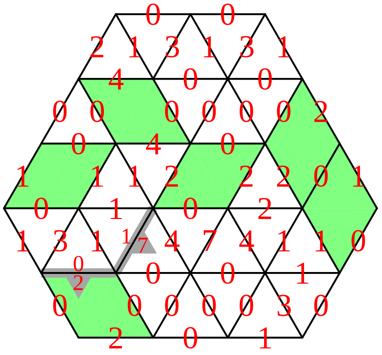
The mutation algorithm



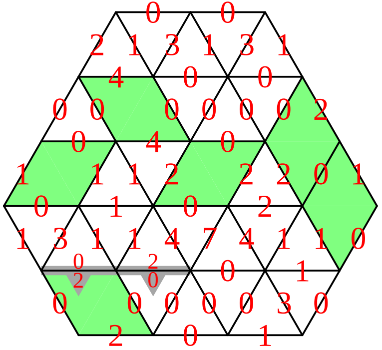
Resolution:



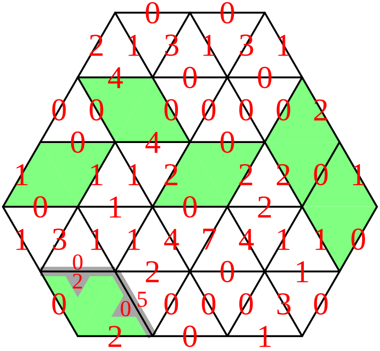
The mutation algorithm



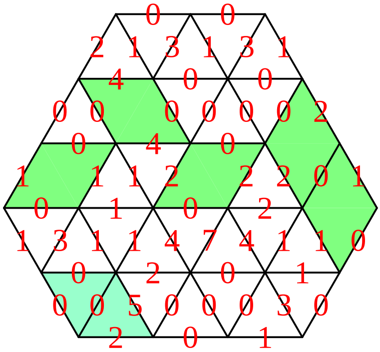
The mutation algorithm



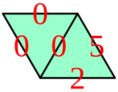
The mutation algorithm



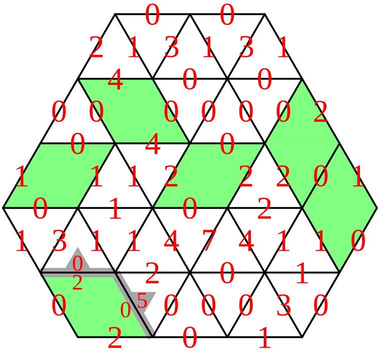
The mutation algorithm



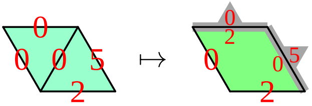
Flawed puzzle containing the **marked scab**:



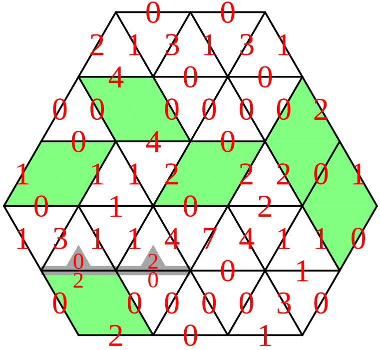
The mutation algorithm



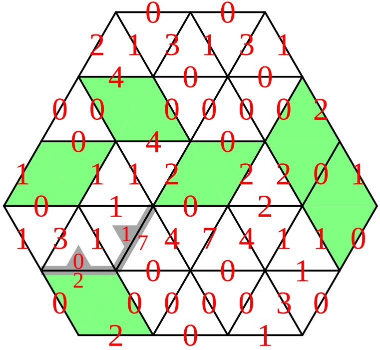
Resolution:



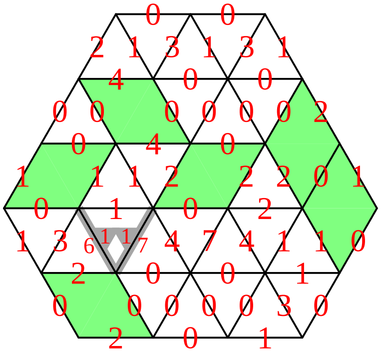
The mutation algorithm



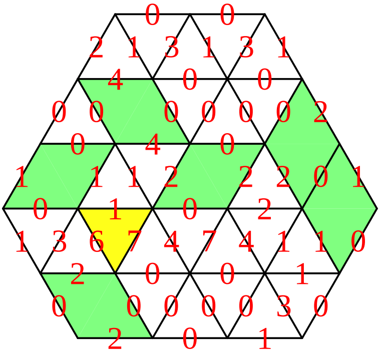
The mutation algorithm



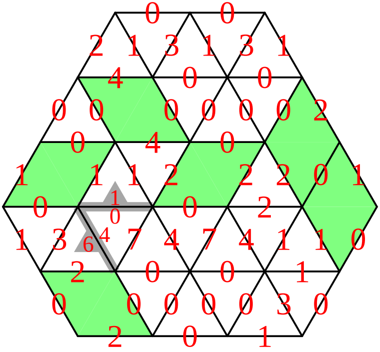
The mutation algorithm



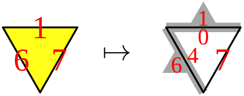
The mutation algorithm



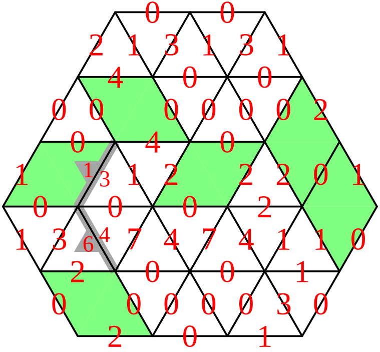
The mutation algorithm



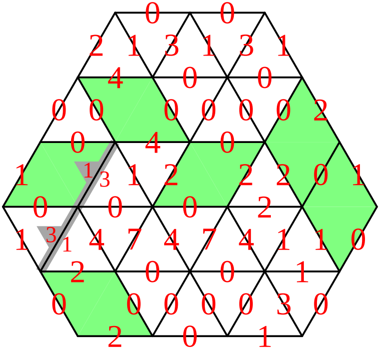
Resolution:



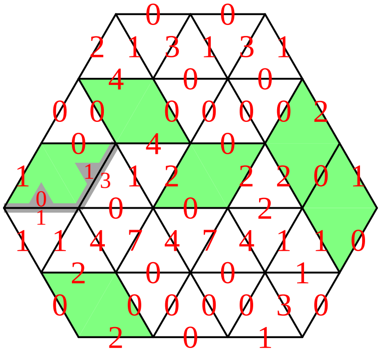
The mutation algorithm



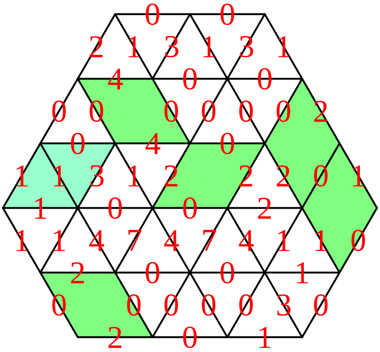
The mutation algorithm



The mutation algorithm

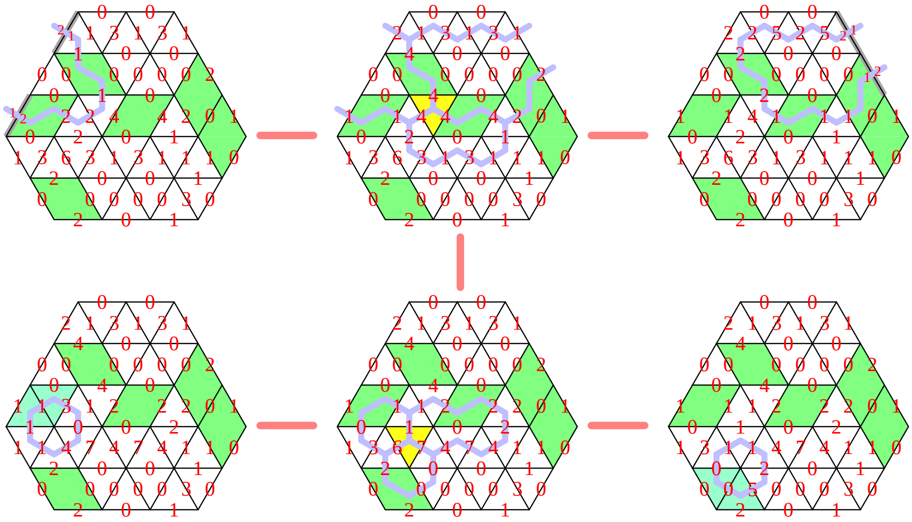


The mutation algorithm



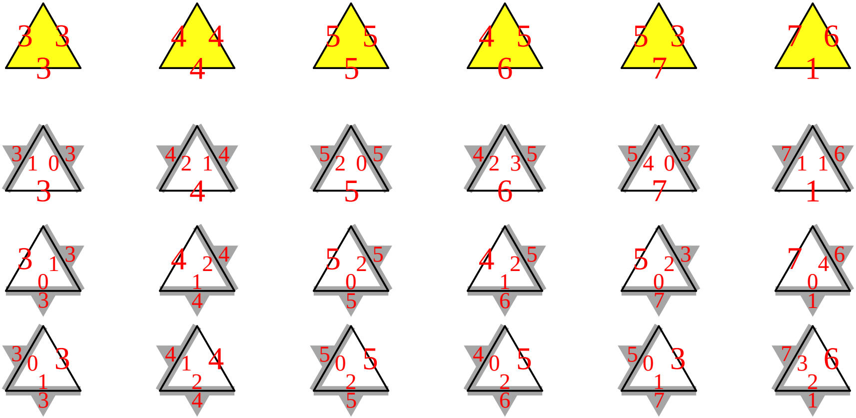
Flawed puzzle containing a marked scab.

Component of the mutation graph:

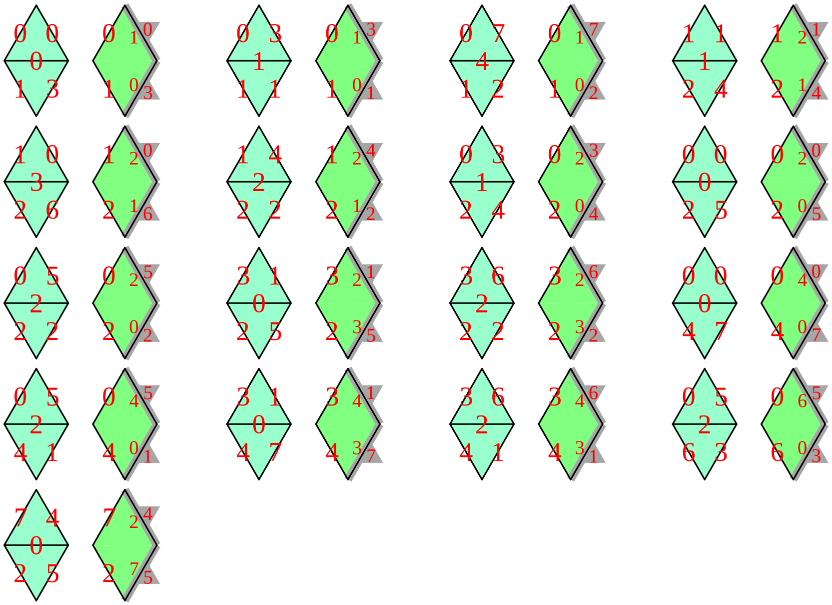


Conjecture: Every component of the mutation graph is a tree.

Resolutions of illegal puzzle pieces:



Resolutions of marked scabs:



Aura: A linear form in $R = \mathbb{C}[\delta_0, \delta_1, \delta_2]$.

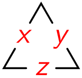
$$\mathcal{A}(\overset{0}{\text{—}}) = \overset{\delta_0}{\uparrow} \qquad \mathcal{A}(\overset{1}{\text{—}}) = \overset{\delta_1}{\uparrow} \qquad \mathcal{A}(\overset{2}{\text{—}}) = \overset{\delta_2}{\uparrow}$$

Aura: A linear form in $R = \mathbb{C}[\delta_0, \delta_1, \delta_2]$.

$\mathcal{A}(\underline{0}) =$ 

$\mathcal{A}(\underline{1}) =$ 

$\mathcal{A}(\underline{2}) =$ 

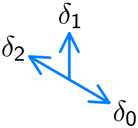
If 

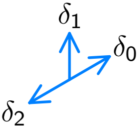
is a puzzle piece, then $\mathcal{A}(/x) + \mathcal{A}(y\backslash) + \mathcal{A}(\underline{z}) = 0$.

$\mathcal{A}(\underline{3}) =$ 

$\mathcal{A}(\underline{4}) =$ 

$\mathcal{A}(\underline{5}) =$ 

$\mathcal{A}(\underline{6}) =$ 

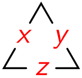
$\mathcal{A}(\underline{7}) =$ 

Aura: A linear form in $R = \mathbb{C}[\delta_0, \delta_1, \delta_2]$.

$\mathcal{A}(\frac{0}{\ }) =$ 

$\mathcal{A}(\frac{1}{\ }) =$ 

$\mathcal{A}(\frac{2}{\ }) =$ 

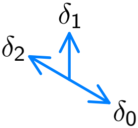
If 

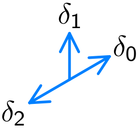
is a puzzle piece, then $\mathcal{A}(/x) + \mathcal{A}(y\backslash) + \mathcal{A}(\frac{z}{\ }) \ = \ 0$.

$\mathcal{A}(\frac{3}{\ }) =$ 

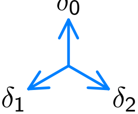
$\mathcal{A}(\frac{4}{\ }) =$ 

$\mathcal{A}(\frac{5}{\ }) =$ 

$\mathcal{A}(\frac{6}{\ }) =$ 

$\mathcal{A}(\frac{7}{\ }) =$ 

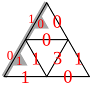
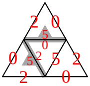
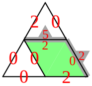
Aura of gash: $\mathcal{A}(\frac{x}{y}) \ = \ \mathcal{A}(\frac{x}{\ }) + \mathcal{A}(\frac{\ }{y})$

Example: $\mathcal{A}(\frac{0}{4}) \ = \ \mathcal{A}(\frac{0}{\ }) + \mathcal{A}(\frac{\ }{4}) \ =$ 

Aura of puzzles

Let $\widetilde{P}$ be a resolution of a flawed puzzle P.

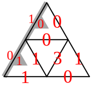
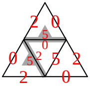
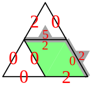
Def: $\mathcal{A}(\widetilde{P}) = \mathcal{A}($ right gash in $\widetilde{P}$ $)$

$\mathcal{A}($  $) = \mathcal{A}(0/1)$ $\mathcal{A}($  $) = \mathcal{A}(\frac{5}{0})$ $\mathcal{A}($  $) = \mathcal{A}(0\backslash 2)$

Aura of puzzles

Let $\widetilde{P}$ be a resolution of a flawed puzzle P.

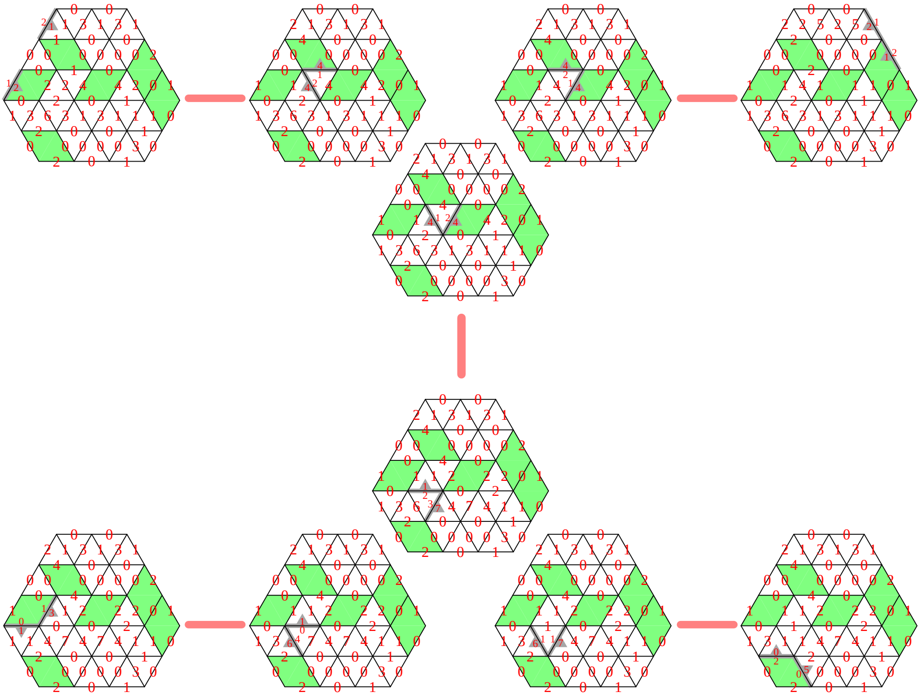
Def: $\mathcal{A}(\widetilde{P}) \;=\; \mathcal{A}($ right gash in $\widetilde{P}$ $)$

$\mathcal{A}($  $) = \mathcal{A}(0/1)$ $\qquad$ $\mathcal{A}($  $) = \mathcal{A}(\frac{5}{0})$ $\qquad$ $\mathcal{A}($  $) = \mathcal{A}(0\backslash 2)$

If $\widetilde{P}$ is the only resolution of P, then set $\mathcal{A}(P) = \mathcal{A}(\widetilde{P})$.

Key identity: Let S be any finite set of flawed puzzles that is closed under mutations. Then

$$\sum_{P \in S_{\mathrm{scab}}} \mathcal{A}(P) \;+\; \sum_{P \in S_{\mathrm{gash}}} \mathcal{A}(P) \;=\; 0$$



From now on:

- All puzzles are triangles.
- All equivariant puzzle pieces and scabs are vertical.

Def: For any 012-string $u = (u_1, u_2, \ldots, u_n)$ we set

$$C_u = \sum_{i=1}^{n} \delta_{u_i}\, y_i \;\in\; R[y_1, \ldots, y_n]$$

Exercise: $\partial P = \triangle_w^{u,v} \quad \Rightarrow$

$$\sum_{s \text{ scab in } P} -\operatorname{weight}(s)\,\mathcal{A}(s) \;=\; C_u \cdot \searrow \;+\; C_v \cdot \swarrow \;+\; C_w \cdot \uparrow$$

From now on:

- All puzzles are triangles.
- All equivariant puzzle pieces and scabs are vertical.

Def: For any 012-string $u = (u_1, u_2, \ldots, u_n)$ we set

$$C_u = \sum_{i=1}^{n} \delta_{u_i}\, y_i \;\in\; R[y_1, \ldots, y_n]$$

Exercise: $\partial P = \triangle_w^{u,v} \quad \Rightarrow$

$$\sum_{s \text{ scab in } P} -\text{weight}(s)\, \mathcal{A}(s) \;=\; C_u \cdot \searrow \;+\; C_v \cdot \swarrow \;+\; C_w \cdot \uparrow$$

Write $u \to u'$ if $u \leq u'$ in Bruhat order and $\ell(u) + 1 = \ell(u')$.

Set $\delta(\frac{u}{u'}) \;=\; \delta_{u_i} - \delta_{u'_i}$ where i is minimal with $u_i \neq u'_i$.

Def: $$\widehat{C}_{u,v}^{w} \;=\; \sum_{\partial P = \triangle_w^{u,v}} \; \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

Molev–Sagan type recursion:

$$\left(C_u \cdot \searrow \;+\; C_v \cdot \swarrow \;+\; C_w \cdot \uparrow \right) \cdot \widehat{C}_{u,v}^{w}$$

$$= \sum_{\partial P = \triangle_{w}^{u,v}} \;\; \sum_{s \text{ scab in } P} -\mathcal{A}(s) \, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

Molev–Sagan type recursion:

$$\left(C_u \cdot \searrow \;+\; C_v \cdot \swarrow \;+\; C_w \cdot \uparrow \right) \cdot \widehat{C}^{w}_{u,v}$$

$$= \sum_{\partial P = \triangle^{u,v}_{w}} \; \sum_{s \text{ scab in } P} -\mathcal{A}(s)\, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

$$= \sum_{\substack{\partial P = \triangle^{u,v}_{w} \\ P \text{ has marked scab } s}} -\mathcal{A}(P)\, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

Molev–Sagan type recursion:

$$\left(C_u \cdot \searrow \;+\; C_v \cdot \swarrow \;+\; C_w \cdot \uparrow \right) \cdot \widehat{C}_{u,v}^{w}$$

$$= \sum_{\partial P = \triangle_w^{u,v}} \; \sum_{s \text{ scab in } P} -\mathcal{A}(s)\, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

$$= \sum_{\substack{\partial P = \triangle_w^{u,v} \\ P \text{ has marked scab } s}} -\mathcal{A}(P)\, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

$$= \sum_{\substack{\partial P = \triangle_w^{u,v} \\ P \text{ has gash pair}}} \mathcal{A}(P) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

Molev–Sagan type recursion:

$$\left(C_u \cdot \searrow \; + \; C_v \cdot \swarrow \; + \; C_w \cdot \uparrow \right) \cdot \widehat{C}_{u,v}^{w}$$

$$= \sum_{\partial P = \triangle_{w}^{u,v}} \; \sum_{s \text{ scab in } P} -\mathcal{A}(s) \, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

$$= \sum_{\substack{\partial P = \triangle_{w}^{u,v} \\ P \text{ has marked scab } s}} -\mathcal{A}(P) \, \text{weight}(s) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

$$= \sum_{\substack{\partial P = \triangle_{w}^{u,v} \\ P \text{ has gash pair}}} \mathcal{A}(P) \prod_{\lozenge \in P} \text{weight}(\lozenge)$$

$$= \nwarrow \cdot \sum_{u \to u'} \delta\left(\frac{u}{u'}\right) \widehat{C}_{u',v}^{w} \; + \; \nearrow \cdot \sum_{v \to v'} \delta\left(\frac{v}{v'}\right) \widehat{C}_{u,v'}^{w} \; + \; \downarrow \cdot \sum_{w' \to w} \delta\left(\frac{w'}{w}\right) \widehat{C}_{u,v}^{w'}$$