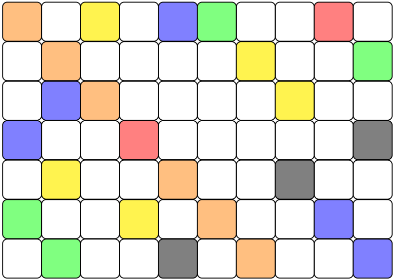
Counting partial Latin rectangles

Rebecca J. Stones (Nankai University, China)

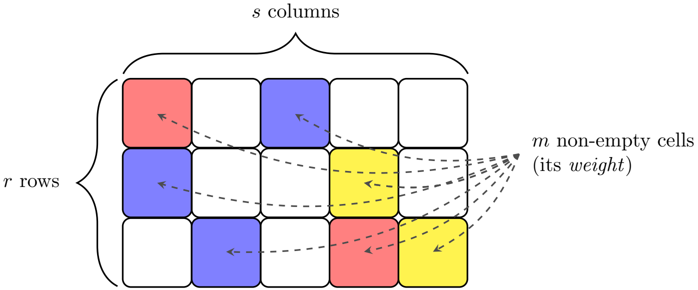
joint work with Raúl Falcón (University of Seville, Spain).

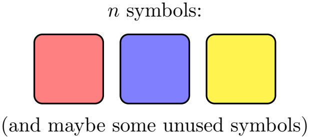
July 7, 2015



Partial Latin rectangles

Here's what I mean by a partial Latin rectangle in this talk:





Latin squares are the case when $r = s = n$ and $m = n^2$.

Latin squares are the case when $r = s = n$ and $m = n^2$.

Thus the partial Latin rectangles we're looking at are

generalized,

generalized,

generalized

Latin squares.

Latin squares are the case when $r = s = n$ and $m = n^2$.

Thus the partial Latin rectangles we're looking at are

generalized,

generalized,

generalized

Latin squares.

number of rows

Latin squares are the case when $r = s = n$ and $m = n^2$.

Thus the partial Latin rectangles we're looking at are

generalized, ← number of rows

generalized, ← number of symbols

generalized

Latin squares.

Latin squares are the case when $r = s = n$ and $m = n^2$.

Thus the partial Latin rectangles we're looking at are

generalized, ← number of rows

generalized, ← number of symbols

generalized ← number of non-empty cells

Latin squares.

Latin squares are the case when $r = s = n$ and $m = n^2$.

Thus the partial Latin rectangles we're looking at are

generalized, ← number of rows

generalized, ← number of symbols

generalized ← number of non-empty cells

Latin squares.

And we're going to count these?!?!

Latin squares are the case when $r = s = n$ and $m = n^2$.

Thus the partial Latin rectangles we're looking at are

generalized, ← number of rows

generalized, ← number of symbols

generalized ← number of non-empty cells

Latin squares.

And we're going to count these?!?!

Method 1: Inclusion-Exclusion

In this method, we count generalized, ordered partial Latin rectangles.

Method 1: Inclusion-Exclusion

In this method, we count generalized, ordered partial Latin rectangles.

we allow clashes, then include-exclude over the number of clashes

Method 1: Inclusion-Exclusion

In this method, we count generalized, ordered partial Latin rectangles.

we allow clashes, then include-exclude over the number of clashes

partial Latin rectangles are interpreted as ordered lists of entries

Method 1: Inclusion-Exclusion

In this method, we count generalized, ordered partial Latin rectangles.

we allow clashes, then include-exclude over the number of clashes

partial Latin rectangles are interpreted as ordered lists of entries

Inclusion-Exclusion gives:

$$m! \, \#\mathrm{PLR}(r, s, n; m) = \sum_{V} (-1)^{|V|} |\mathcal{B}_V|.$$

Method 1: Inclusion-Exclusion

In this method, we count generalized, ordered partial Latin rectangles.

we allow clashes, then include-exclude over the number of clashes

partial Latin rectangles are interpreted as ordered lists of entries

Inclusion-Exclusion gives:

$$m! \, \#\mathrm{PLR}(r, s, n; m) = \sum_{V} (-1)^{|V|} |\mathcal{B}_V|.$$

set of clashes

Method 1: Inclusion-Exclusion

In this method, we count generalized, ordered partial Latin rectangles.

we allow clashes, then include-exclude over the number of clashes

partial Latin rectangles are interpreted as ordered lists of entries

Inclusion-Exclusion gives:

$$m!\,\#\mathrm{PLR}(r,s,n;m) = \sum_{V} (-1)^{|V|} |\mathcal{B}_V|.$$

set of clashes

set of generalized ordered PLRs with clashes in V (and maybe more)

Example set of clashes V :

$e_1, e_2,$ same cell

$e_1, e_3,$ same cell

$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row

$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column

Example set of clashes V :

$e_1, e_2,$ same cell

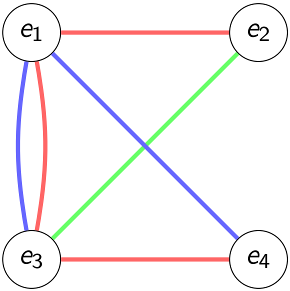
$e_1, e_3,$ same cell

$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row

$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column



Example set of clashes V :

$e_1, e_2,$ same cell

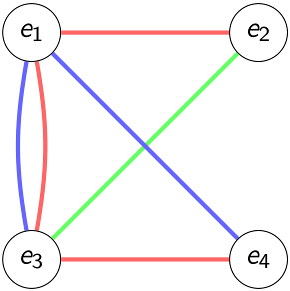
$e_1, e_3,$ same cell

$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row

$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column



Computing $|\mathcal{B}_V|$ is now a graph coloring problem.

Example set of clashes V :

$e_1, e_2,$ same cell

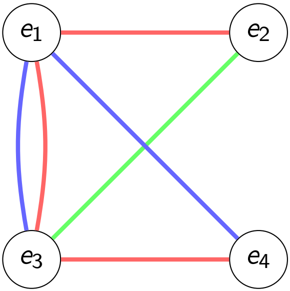
$e_1, e_3,$ same cell

$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row

$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column



Computing $|\mathcal{B}_V|$ is now a graph coloring problem.

If we "color" e_1 with (r_1, c_1, s_1) and e_2 with (r_2, c_2, s_2), then we want $r_1 = r_2$ and $c_1 = c_2$ to match the edge coloring.

Example set of clashes V :

$e_1, e_2,$ same cell

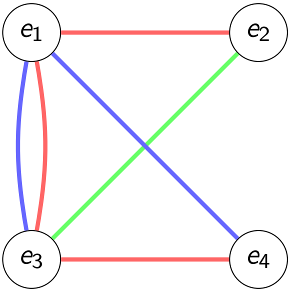
$e_1, e_3,$ same cell

$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row

$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column



Computing $|\mathcal{B}_V|$ is now a graph coloring problem.

If we "color" e_1 with (r_1, c_1, s_1) and e_2 with (r_2, c_2, s_2), then we want $r_1 = r_2$ and $c_1 = c_2$ to match the edge coloring.

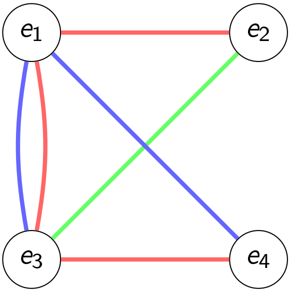
And so on for the other edges.

Example set of clashes V :

$e_1, e_2,$ same cell
$e_1, e_3,$ same cell
$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row
$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column



Computing $|\mathcal{B}_V|$ is now a graph coloring problem.

If we "color" e_1 with (r_1, c_1, s_1) and e_2 with (r_2, c_2, s_2), then we want $r_1 = r_2$ and $c_1 = c_2$ to match the edge coloring.

And so on for the other edges.

Parallel edges imply $(r_i, c_i, s_i) = (r_j, c_j, s_j)$, regardless of the colors of the edges.

Example set of clashes V :

$e_1, e_2,$ same cell

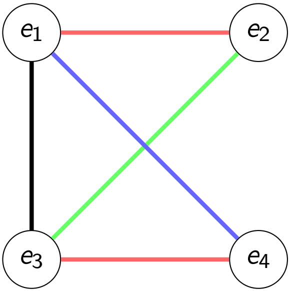
$e_1, e_3,$ same cell

$e_3, e_4,$ same cell

$e_1, e_3,$ same symbol and row

$e_1, e_4,$ same symbol and row

$e_2, e_3,$ same symbol and column



Computing $|\mathcal{B}_V|$ is now a graph coloring problem.

If we “color” e_1 with (r_1, c_1, s_1) and e_2 with (r_2, c_2, s_2), then we want $r_1 = r_2$ and $c_1 = c_2$ to match the edge coloring.

And so on for the other edges.

Parallel edges imply $(r_i, c_i, s_i) = (r_j, c_j, s_j)$, regardless of the colors of the edges. So we replace them with a single black edge.

If we rephrase in terms of these colorings...

For all $m, r, s, n \geq 1$, we have

$$m!\, \#\mathrm{PLR}(r, s, n; m) =$$

$$(rsn)^m + \sum_{v \geq 2} \sum_{e \geq 1} (-1)^e \binom{m}{v} (rsn)^{m-v+1} \sum_{G \in \Gamma_{e,v}} \frac{v!}{|\mathrm{Aut}(G)|} P(G)$$

where

$$P(G) = P(G; r, s, n) = \sum_{\delta} (-2)^{\mathbf{b}(\delta)} r^{c(H_3)-1} s^{c(H_2)-1} n^{c(H_1)-1}$$

where the sum is over all (red, blue, green, black) edge colorings δ of G.

If we rephrase in terms of these colorings...

For all $m, r, s, n \geq 1$, we have

$$m!\,\#\mathrm{PLR}(r, s, n; m) =$$

$$(rsn)^m + \sum_{v \geq 2} \sum_{e \geq 1} (-1)^e \binom{m}{v} (rsn)^{m-v+1} \sum_{G \in \Gamma_{e,v}} \frac{v!}{|\mathrm{Aut}(G)|} P(G)$$

where

$$P(G) = P(G; r, s, n) = \sum_{\delta} (-2)^{\mathbf{b}(\delta)} r^{c(H_3)-1} s^{c(H_2)-1} n^{c(H_1)-1}$$

where the sum is over all (red, blue, green, black) edge colorings δ of G.

What's important here:

If we rephrase in terms of these colorings...

For all $m, r, s, n \geq 1$, we have

$$m!\,\#\mathrm{PLR}(r, s, n; m) =$$
$$(rsn)^m + \sum_{v\geq 2}\sum_{e\geq 1}(-1)^e\binom{m}{v}(rsn)^{m-v+1}\sum_{G\in\Gamma_{e,v}}\frac{v!}{|\mathrm{Aut}(G)|}P(G)$$

where

$$P(G) = P(G; r, s, n) = \sum_{\delta}(-2)^{\mathbf{b}(\delta)}r^{c(H_3)-1}s^{c(H_2)-1}n^{c(H_1)-1}$$

where the sum is over all (red, blue, green, black) edge colorings δ of G.

What's important here:

- For arbitrary simple graphs G, there is a graph polynomial $P(G)$.

If we rephrase in terms of these colorings...

For all $m, r, s, n \geq 1$, we have

$$m!\,\#\mathrm{PLR}(r,s,n;m) =$$
$$(rsn)^m + \sum_{v\geq 2}\sum_{e\geq 1}(-1)^e\binom{m}{v}(rsn)^{m-v+1}\sum_{G\in\Gamma_{e,v}}\frac{v!}{|\mathrm{Aut}(G)|}P(G)$$

where

$$P(G) = P(G;r,s,n) = \sum_{\delta}(-2)^{\mathbf{b}(\delta)}r^{c(H_3)-1}s^{c(H_2)-1}n^{c(H_1)-1}$$

where the sum is over all (red, blue, green, black) edge colorings δ of G.

What's important here:

- For arbitrary simple graphs G, there is a graph polynomial $P(G)$.
- If we compute $P(G)$ and $|\mathrm{Aut}(G)|$ for small graphs, we find $\#\mathrm{PLR}(r,s,n;m)$ for small m.

We compute these polynomials and automorphism group sizes:

G	v	e	$c(G)$	$\lvert\mathrm{Aut}(G)\rvert$	$P(G) = P(G; r, s, n)$
	2	1	1	2	$\overline{100} - 2$
	3	2	1	2	$P($$)^2$
	3	3	1	6	$\overline{200} - 2$
	4	2	2	8	$\overline{111}\, P($$)^2$
	4	3	1	6	$P($$)^3$
	4	3	1	2	$P($$)^3$
	4	1	1	2	$P($$)P($$)$
	4	4	1	8	$\overline{300} + 6\,\overline{110} - 12\,\overline{100} + 16$
	4	5	1	4	$\overline{300} + 2\,\overline{110} - 4\,\overline{100} + 4$
	4	6	1	24	$\overline{300} - 2$
	5	3	2	4	$\overline{111}\, P($$)^3$
	5	4	2	12	$\overline{111}\, P($$)P($$)$
	6	3	3	48	$\overline{222}\, P($$)^3$

...and so on.

We compute these polynomials and automorphism group sizes:

G	v	e	$c(G)$	$\lvert\mathrm{Aut}(G)\rvert$	$P(G) = P(G; r, s, n)$
	2	1	1	2	$\overline{100} - 2$
	3	2	1	2	$P($$)^2$
	3	3	1	6	$\overline{200} - 2$
	4	2	2	8	$\overline{111}\,P($$)^2$
	4	3	1	6	$P($$)^3$
	4	3	1	2	$P($$)^3$
	4	1	1	2	$P($$)P($$)$
	4	4	1	8	$\overline{300} + 6\,\overline{110} - 12\,\overline{100} + 16$
	4	5	1	4	$\overline{300} + 2\,\overline{110} - 4\,\overline{100} + 4$
	4	6	1	24	$\overline{300} - 2$
	5	3	2	4	$\overline{111}\,P($$)^3$
	5	4	2	12	$\overline{111}\,P($$)P($$)$
	6	3	3	48	$\overline{222}\,P($$)^3$

...and so on.

Here, we use this shorthand:

$$\overline{210} = r^2 s + r^2 n + s^2 r + s^2 n + n^2 r + n^2 s, \text{ and}$$

$$2\,\overline{100} = 2(r + s + t).$$

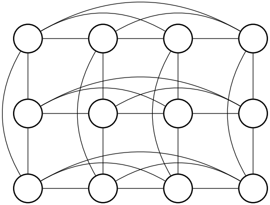
The asymptotic number of partial Latin rectangles of fixed weight...

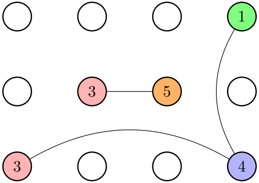
For fixed m, we have

$$\mathbf{m!\,\#\mathrm{PLR}(r,s,n;m)} = (rsn)^m + \binom{m}{2}(rsn)^{m-1}(2-\overline{100}) + \binom{m}{3}(rsn)^{m-2}(14 - 12\,\overline{100} + 6\,\overline{110} + 2\,\overline{200}) + \binom{m}{4}(rsn)^{m-3}(198 - 228\,\overline{100} + 198\,\overline{110} - 84\,\overline{111} + 72\,\overline{200} - 36\,\overline{210} - 12\,\overline{211} + 6\,\overline{221} - 6\,\overline{300} + 3\,\overline{311}) + \binom{m}{5}(rsn)^{m-4}(-6360\,\overline{100} + 7440\,\overline{110} - 6080\,\overline{111} + 2880\,\overline{200} - 2520\,\overline{210} + 820\,\overline{211} + 480\,\overline{220} + 360\,\overline{221} - 180\,\overline{222} - 480\,\overline{300} + 240\,\overline{310} + 160\,\overline{311} - 80\,\overline{321} + 24\,\overline{400} - 20\,\overline{411}) + \binom{m}{6}(rsn)^{m-5}(-13170\,\overline{211} + 17340\,\overline{221} - 15990\,\overline{222} + 7580\,\overline{311} - 7050\,\overline{321} + 3300\,\overline{322} + 1520\,\overline{331} + 180\,\overline{332} - 90\,\overline{333} - 1740\,\overline{411} + 870\,\overline{421} + 90\,\overline{422} - 45\,\overline{432} + 130\,\overline{511} - 15\,\overline{522}) + \binom{m}{7}(rsn)^{m-6}(-10920\,\overline{322} + 15540\,\overline{332} - 15120\,\overline{333} + 7350\,\overline{422} - 7140\,\overline{432} + 3570\,\overline{433} + 1680\,\overline{442} - 2100\,\overline{522} + 1050\,\overline{532} + 210\,\overline{622}) + \binom{m}{8}(rsn)^{m-7}(-3360\,\overline{433} + 5040\,\overline{443} - 5040\,\overline{444} + 2520\,\overline{533} - 2520\,\overline{543} + 1260\,\overline{544} + 630\,\overline{553} - 840\,\overline{633} + 420\,\overline{643} + 105\,\overline{733}) + \textbf{some polynomial of degree } \mathbf{\leq 3m-12}.$$

Method 2: Chromatic Polynomials

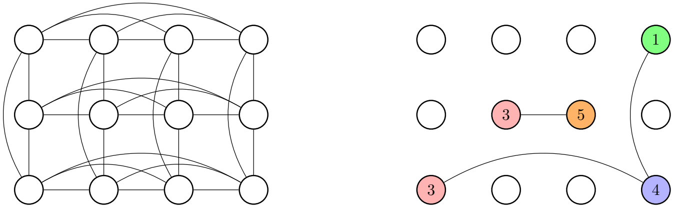
Any partial Latin rectangle $\mathrm{PLR}(r, s, n; m)$ can be interpreted as a proper n-coloring of an m-vertex induced subgraph of the $r \times s$ rook's graph.





Method 2: Chromatic Polynomials

Any partial Latin rectangle $\mathrm{PLR}(r, s, n; m)$ can be interpreted as a proper n-coloring of an m-vertex induced subgraph of the $r \times s$ rook's graph.



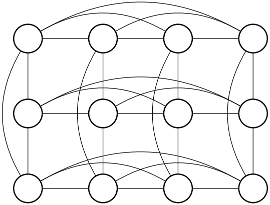
If Π denotes the chromatic polynomial, we thus have

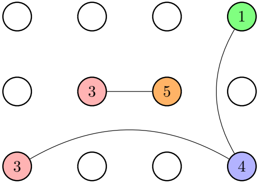
$$\#\mathrm{PLR}(r, s, n; m) = \sum_{M} \Pi(M; n)$$

over all m-vertex induced subgraphs M of the $r \times s$ rook's graph.

Method 2: Chromatic Polynomials

Any partial Latin rectangle $\mathrm{PLR}(r, s, n; m)$ can be interpreted as a proper n-coloring of an m-vertex induced subgraph of the $r \times s$ rook's graph.





If Π denotes the chromatic polynomial, we thus have

$$\#\mathrm{PLR}(r, s, n; m) = \sum_{M} \Pi(M; n)$$

over all m-vertex induced subgraphs M of the $r \times s$ rook's graph. Or, equivalently, $(0, 1)$-matrices with m ones.

We can permute the rows and columns of a $(0,1)$-matrix with m ones into a canonical form:

$$\begin{array}{|c|c|c|c|c|}
\hline
K_1 & \emptyset & \cdots & \emptyset & \emptyset \\
\hline
\emptyset & K_2 & & \emptyset & \emptyset \\
\hline
\vdots & & \ddots & & \vdots \\
\hline
\emptyset & \emptyset & & K_k & \emptyset \\
\hline
\emptyset & \emptyset & \cdots & \emptyset & \emptyset \\
\hline
\end{array}.$$

The blocks $K_1, K_2, \ldots, K_k$ are in some kind of canonical form under row/column permutations.

If we sum over such canonical forms, for fixed m, we get:

$$m!\,\#\mathrm{PLR}(r,s,n;m) =$$

$$\sum_{k\geq 0}\ \sum_{\substack{(K_1,K_2,\ldots,K_k)\\ m \text{ ones}}}\ \sum_{\text{good } (t_i)_{i=1}^k} [r]_{e_{\mathrm{row}}}[s]_{e_{\mathrm{col}}} \frac{\prod_{i=1}^k \Pi(K_i;n)}{\left(\prod_{i=1}^k |\mathrm{Aut}(G_{K_i})|\right)\left(\prod_{i=1}^\ell k_i!\right)}$$

where...

If we sum over such canonical forms, for fixed m, we get:

$$m!\,\#\mathrm{PLR}(r,s,n;m) =$$

$$\sum_{k\geq 0}\ \sum_{\substack{(K_1,K_2,\ldots,K_k)\\ m \text{ ones}}}\ \sum_{\text{good } (t_i)_{i=1}^{k}} [r]_{e_{\text{row}}}[s]_{e_{\text{col}}} \frac{\prod_{i=1}^{k} \Pi(K_i;n)}{\left(\prod_{i=1}^{k} |\mathrm{Aut}(G_{K_i})|\right)\left(\prod_{i=1}^{\ell} k_i!\right)}$$

where...

- the t_i's keep track of which matrices are transposed (saves computation),

If we sum over such canonical forms, for fixed m, we get:

$$m!\,\#\mathrm{PLR}(r,s,n;m) =$$

$$\sum_{k\geq 0}\ \sum_{\substack{(K_1,K_2,\dots,K_k)\\ m \text{ ones}}}\ \sum_{\text{good } (t_i)_{i=1}^k} [r]_{e_{\text{row}}}[s]_{e_{\text{col}}} \frac{\prod_{i=1}^k \Pi(K_i;n)}{\left(\prod_{i=1}^k |\mathrm{Aut}(G_{K_i})|\right)\left(\prod_{i=1}^\ell k_i!\right)}$$

where...

- the t_i's keep track of which matrices are transposed (saves computation),
- $[r]_{e_{\text{row}}} = r(r-1)\cdots(r-e_{\text{row}}+1)$ and $[s]_{e_{\text{col}}} = s(s-1)\cdots(s-e_{\text{col}}+1)$; e_{row} and e_{col} denote the number of empty rows and columns

If we sum over such canonical forms, for fixed m, we get:

$$m!\,\#\mathrm{PLR}(r,s,n;m) =$$

$$\sum_{k\geq 0}\ \sum_{\substack{(K_1,K_2,\ldots,K_k)\\ m \text{ ones}}}\ \sum_{\text{good }(t_i)_{i=1}^{k}} [r]_{e_{\text{row}}}[s]_{e_{\text{col}}} \frac{\prod_{i=1}^{k}\Pi(K_i;n)}{\left(\prod_{i=1}^{k}|\mathrm{Aut}(G_{K_i})|\right)\left(\prod_{i=1}^{\ell}k_i!\right)}$$

where...

- the t_i's keep track of which matrices are transposed (saves computation),
- $[r]_{e_{\text{row}}} = r(r-1)\cdots(r-e_{\text{row}}+1)$ and $[s]_{e_{\text{col}}} = s(s-1)\cdots(s-e_{\text{col}}+1)$; e_{row} and e_{col} denote the number of empty rows and columns
- k_i, for $i\in\{1,2,\ldots,\ell\}$, be the number of copies of the i-th distinct matrix (given ℓ distinct matrices).

So we compute...

block K	$\lvert \mathrm{Aut}(G_K) \rvert$	$\Pi(K;n)$
$\begin{matrix}1\end{matrix}$	1	n
$\begin{matrix}1 & 1\end{matrix}$	2	$n^2 - n$
$\begin{matrix}1 & 1 & 1\end{matrix}$	6	$n^3 - 3n^2 + 2n$
$\begin{matrix}1 & 1 \\ 1 & 0\end{matrix}$	1	$n^3 - 2n^2 + n$
$\begin{matrix}1 & 1 & 1 & 1\end{matrix}$	24	$n^4 - 6n^3 + 11n^2 - 6n$
$\begin{matrix}1 & 1 & 1 \\ 1 & 0 & 0\end{matrix}$	2	$n^4 - 4n^3 + 5n^2 - 2n$
$\begin{matrix}1 & 1 & 0 \\ 1 & 0 & 1\end{matrix}$	2	$n^4 - 3n^3 + 3n^2 - n$
$\begin{matrix}1 & 1 \\ 1 & 1\end{matrix}$	4	$n^4 - 4n^3 + 6n^2 - 3n$
$\begin{matrix}1 & 1 & 1 & 1 & 1\end{matrix}$	120	$n^5 - 10n^4 + 35n^3 - 50n^2 + 24n$
$\begin{matrix}1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0\end{matrix}$	6	$n^5 - 7n^4 + 17n^3 - 17n^2 + 6n$
$\begin{matrix}1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1\end{matrix}$	2	$n^5 - 5n^4 + 9n^3 - 7n^2 + 2n$
$\begin{matrix}1 & 1 & 1 \\ 1 & 1 & 0\end{matrix}$	2	$n^5 - 6n^4 + 14n^3 - 15n^2 + 6n$
$\begin{matrix}1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0\end{matrix}$	4	$n^5 - 6n^4 + 13n^3 - 12n^2 + 4n$
$\begin{matrix}1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0\end{matrix}$	2	$n^5 - 5n^4 + 9n^3 - 7n^2 + 2n$
$\begin{matrix}1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0\end{matrix}$	2	$n^5 - 5n^4 + 9n^3 - 7n^2 + 2n$
$\begin{matrix}1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0\end{matrix}$	1	$n^5 - 4n^4 + 6n^3 - 4n^2 + n$

...and so on.

And we get **exact formulas** for the number of small-weight partial Latin rectangles:

$$1!\,\#\mathrm{PLR}(r,s,n;1)=\overline{111}.$$

$$2!\,\#\mathrm{PLR}(r,s,n;2)=\overline{222}-\overline{211}+2\,\overline{111}.$$

$$3!\,\#\mathrm{PLR}(r,s,n;3)=$$
$$\overline{333}-3\,\overline{322}+6\,\overline{222}+2\,\overline{311}+6\,\overline{221}-12\,\overline{211}+14\,\overline{111}.$$

$$4!\,\#\mathrm{PLR}(r,s,n;4)=$$
$$\overline{444}-6\,\overline{433}+12\,\overline{333}+11\,\overline{422}+30\,\overline{332}-60\,\overline{322}-6\,\overline{411}-$$
$$36\,\overline{321}-28\,\overline{222}+72\,\overline{311}+198\,\overline{221}-228\,\overline{211}+198\,\overline{111}.$$

And we get **exact formulas** for the number of small-weight partial Latin rectangles:

$$1!\,\#\mathrm{PLR}(r,s,n;1) = \overline{111}.$$

$$2!\,\#\mathrm{PLR}(r,s,n;2) = \overline{222} - \overline{211} + 2\,\overline{111}.$$

$$3!\,\#\mathrm{PLR}(r,s,n;3) = \\ \overline{333} - 3\,\overline{322} + 6\,\overline{222} + 2\,\overline{311} + 6\,\overline{221} - 12\,\overline{211} + 14\,\overline{111}.$$

$$4!\,\#\mathrm{PLR}(r,s,n;4) = \\ \overline{444} - 6\,\overline{433} + 12\,\overline{333} + 11\,\overline{422} + 30\,\overline{332} - 60\,\overline{322} - 6\,\overline{411} - \\ 36\,\overline{321} - 28\,\overline{222} + 72\,\overline{311} + 198\,\overline{221} - 228\,\overline{211} + 198\,\overline{111}.$$

In this way, we managed to compute the exact formulas for up to weight $m = 14$.

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Or a combination of both of these.

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Or a combination of both of these. This equivalence relation is called *Sade equivalence* or *template equivalence*.

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Or a combination of both of these. This equivalence relation is called *Sade equivalence* or *template equivalence*.

We implement Sade's method by:

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Or a combination of both of these. This equivalence relation is called *Sade equivalence* or *template equivalence*.

We implement Sade's method by: (a) maintaining a list of Sade inequivalent $r \times n$ Latin rectangles, and the number of equivalent Latin rectangles for each representative,

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \dots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Or a combination of both of these. This equivalence relation is called *Sade equivalence* or *template equivalence*.

We implement Sade's method by: (a) maintaining a list of Sade inequivalent $r \times n$ Latin rectangles, and the number of equivalent Latin rectangles for each representative, (b) extending these representatives to $(r+1) \times n$ Latin rectangles in all possible ways, and

Method 3: Generalizing Sade's Method

Sade's method (c. 1948) outstrips all other methods for finding the number of Latin squares.

Two $r \times n$ Latin rectangles on the symbol set $\{1, 2, \ldots, n\}$ have same number of extensions $(r+1) \times n$ Latin rectangles if:

- They have the same **set of symbols** in each column.
- We can permute the columns and/or symbols of one to give the other.

Or a combination of both of these. This equivalence relation is called *Sade equivalence* or *template equivalence*.

We implement Sade's method by: (a) maintaining a list of Sade inequivalent $r \times n$ Latin rectangles, and the number of equivalent Latin rectangles for each representative, (b) extending these representatives to $(r+1) \times n$ Latin rectangles in all possible ways, and (c) filtering out Sade equivalent extensions.

Sade's method works almost identically for partial Latin rectangles (unsurprisingly),

Sade's method works almost identically for partial Latin rectangles (unsurprisingly), but there's additional work in keeping track of the weight (number of non-empty cells) as we go along.

Sade's method works almost identically for partial Latin rectangles (unsurprisingly), but there's additional work in keeping track of the weight (number of non-empty cells) as we go along.

There's more partial Latin rectangles than Latin rectangles.

Sade's method works almost identically for partial Latin rectangles (unsurprisingly), but there's additional work in keeping track of the weight (number of non-empty cells) as we go along.

There's more partial Latin rectangles than Latin rectangles.

We're hoping to compute $\#\mathrm{PLR}(r,s,n;m)$ in this way whenever $r,s,n \leq 7$.

Sade's method works almost identically for partial Latin rectangles (unsurprisingly), but there's additional work in keeping track of the weight (number of non-empty cells) as we go along.

There's more partial Latin rectangles than Latin rectangles.

We're hoping to compute $\#\mathrm{PLR}(r, s, n; m)$ in this way whenever $r, s, n \leq 7$.

My computer is currently up to 5×7 (after about 3 months computation).

Sade's method works almost identically for partial Latin rectangles (unsurprisingly), but there's additional work in keeping track of the weight (number of non-empty cells) as we go along.

There's more partial Latin rectangles than Latin rectangles.

We're hoping to compute $\#\mathrm{PLR}(r, s, n; m)$ in this way whenever $r, s, n \leq 7$.

My computer is currently up to 5×7 (after about 3 months computation). After getting to 6×7 it'll switch to counting via a backtracking algorithm (which is faster for the last row).

Method 4: Algebraic Geometry

I won't go to far into this: Falcón will talk about this in detail at EuroComb.

Method 4: Algebraic Geometry

I won't go to far into this: Falcón will talk about this in detail at EuroComb.

Partial Latin rectangles $\mathrm{PLR}(r, s, n)$ correspond to zeros of the ideal

$$I_{r,s,n} = \langle \text{ carefully selected polynomials } \rangle$$

of $GF(2)[x_{111}, \ldots, x_{rsn}]$.

Method 4: Algebraic Geometry

I won't go to far into this: Falcón will talk about this in detail at EuroComb.

Partial Latin rectangles $\mathrm{PLR}(r, s, n)$ correspond to zeros of the ideal

$$I_{r,s,n} = \langle \text{ carefully selected polynomials } \rangle$$

of $GF(2)[x_{111}, \ldots, x_{rsn}]$.

↑ $x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$

Method 4: Algebraic Geometry

I won't go to far into this: Falcón will talk about this in detail at EuroComb.

Partial Latin rectangles $\mathrm{PLR}(r, s, n)$ correspond to zeros of the ideal

$$I_{r,s,n} = \langle \text{ carefully selected polynomials } \rangle$$

of $GF(2)[x_{111}, \ldots, x_{rsn}]$.

$x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$
$x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$

Method 4: Algebraic Geometry

I won't go to far into this: Falcón will talk about this in detail at EuroComb.

Partial Latin rectangles $\mathrm{PLR}(r,s,n)$ correspond to zeros of the ideal

$$I_{r,s,n} = \langle \text{ carefully selected polynomials } \rangle$$

of $GF(2)[x_{111},\dots,x_{rsn}]$.

$x_{ijk}x_{i'jk}$ for distinct $i,i' \in [r]$

$x_{ijk}x_{ij'k}$ for distinct $j,j' \in [s]$

$x_{ijk}x_{ijk'}$ for distinct $k,k' \in [n]$

Method 4: Algebraic Geometry

I won't go to far into this: Falcón will talk about this in detail at EuroComb.

Partial Latin rectangles $\mathrm{PLR}(r, s, n)$ correspond to zeros of the ideal

$$I_{r,s,n} = \langle \text{ carefully selected polynomials } \rangle$$

of $GF(2)[x_{111}, \ldots, x_{rsn}]$.

- $x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$
- $x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$
- $x_{ijk}x_{ijk'}$ for distinct $k, k' \in [n]$

For the partial Latin rectangle $P = (p_{ij})$ we have $p_{ij} = k$ whenever $x_{ijk} = 1$, and p_{ij} is undefined otherwise.

Thus (from algebraic geometry)

$$\#\mathrm{PLR}(r, s, n) = \dim_{\mathrm{GF}(2)}\bigl(\mathrm{GF}(2)[\mathbf{x}]/I_{r,s,n}\bigr)$$

and

$$\#\mathrm{PLR}(r, s, n; m) = \mathrm{HF}_{GF(2)[\mathbf{x}]/I_{r,s,n}}(m)$$

where HF denotes the *Hilbert function* and [other things I'm going to skip].

Thus (from algebraic geometry)

$$\#\mathrm{PLR}(r, s, n) = \dim_{\mathrm{GF}(2)}\big(\mathrm{GF}(2)[\mathbf{x}]/I_{r,s,n}\big)$$

and

$$\#\mathrm{PLR}(r, s, n; m) = \mathrm{HF}_{GF(2)[\mathbf{x}]/I_{r,s,n}}(m)$$

where HF denotes the *Hilbert function* and [other things I'm going to skip].

There are algorithms in algebraic geometry to compute this Hilbert function.

Thus (from algebraic geometry)

$$\#\mathrm{PLR}(r, s, n) = \dim_{\mathrm{GF}(2)}\bigl(\mathrm{GF}(2)[\mathbf{x}]/I_{r,s,n}\bigr)$$

and

$$\#\mathrm{PLR}(r, s, n; m) = \mathrm{HF}_{GF(2)[\mathbf{x}]/I_{r,s,n}}(m)$$

where HF denotes the *Hilbert function* and [other things I'm going to skip].

There are algorithms in algebraic geometry to compute this Hilbert function. In this way, we compute $\#\mathrm{PLR}(r, s, n; m)$ whenever $r, s, n \leq 6$.

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry.

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols.

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate.

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

↑ $x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

$x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$

$x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

- $x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$
- $x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$
- $x_{ijk}x_{ijk'}$ for distinct $k, k' \in [n]$

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

$x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$

$x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$

$x_{ijk}x_{ijk'}$ for distinct $k, k' \in [n]$

$x_{ijk} - x_{\alpha(i)\beta(j)\gamma(k)}$ for $i, j, k \in [n]$

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

- $x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$
- $x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$
- $x_{ijk}x_{ijk'}$ for distinct $k, k' \in [n]$
- $x_{ijk} - x_{\alpha(i)\beta(j)\gamma(k)}$ for $i, j, k \in [n]$

Partial Latin rectangles $\mathrm{PLR}(r, s, n)$ that admit the symmetry (α, β, γ) correspond to zeros of this ideal.

But we'll be more interested in using this method for enumerating partial Latin rectangles up to symmetry. *Isotopism*: permute rows/columns/symbols. *Paratopism*: permute rows/columns/symbols and conjugate. *Isomorphism*: permute rows/columns/symbols (same permutations; square case only).

E.g. for isotopism equivalence:

$$\langle \text{ carefully selected polynomials } \rangle$$

$x_{ijk}x_{i'jk}$ for distinct $i, i' \in [r]$

$x_{ijk}x_{ij'k}$ for distinct $j, j' \in [s]$

$x_{ijk}x_{ijk'}$ for distinct $k, k' \in [n]$

$x_{ijk} - x_{\alpha(i)\beta(j)\gamma(k)}$ for $i, j, k \in [n]$

Partial Latin rectangles $\mathrm{PLR}(r, s, n)$ that admit the symmetry (α, β, γ) correspond to zeros of this ideal.

We then use Burnside's Lemma, and sum over possible symmetries to give the number of equivalence classes.

This has been used to find the number of isomorphism and isotopism classes of partial Latin rectangles in small cases.

	# isomorphism classes					
	n					
m	1	2	3	4	5	6
0	1	1	1	1	1	1
1	1	4	5	5	5	5
2		10	50	84	93	94
3		4	221	1120	2112	2548
4		1	525	10128	43955	85234
5			651	60092	674957	2508483
6			415	239302	7679384	59110661
7			136	639098	65404265	1103309385
8			20	1148454	422142208	16466869051
9			5	1374447	2080853035	198621450446
10				1082019	7867483199	1953036511736
11				548440	22843744418	15756857221135
12				176137	50867669444	104784604156741
13				35473	86544642569	576125696499417
14				4696	111836743580	2623564948795633
15				403	108882205792	9901507463165937
16				35	79051125332	30959687376379661
17					42275685836	80100291981771263
18					16420711804	171118574787473668
19					4563456676	300957676311237853
20					894429087	434125855232450974
21					122238972	511227919780309083
22					11569016	488771341028032846
23					759296	376957644290919036
24					33736	232788472371575258
25					1411	114149339445885218
26						44033009520708974
27						13227534274721732
28						3061826358557444
29						540473537486248
30						72090555296085
31						7217657260917
32						540810639064
33						30364554576
34						1285684592
35						40649375
36						1130531
Total	2	20	2029	5319934	534759300182	2815323435872410905

	# isotopism classes					
	n					
m	1	2	3	4	5	6
0	1	1	1	1	1	1
1	1	1	1	1	1	1
2		4	4	4	4	4
3		1	11	11	11	11
4		1	18	52	52	52
5			23	139	221	221
6			15	507	1158	1396
7			6	1161	6310	9130
8			1	2136	33293	72145
9			1	2429	150964	583339
10				2004	554285	4627607
11				975	1594532	33362634
12				364	3539461	210409407
13				72	6017824	1129335392
14				18	7772366	5091624997
15				2	7568187	19140028219
16				2	5493206	59761963636
17					2939617	154544375137
18					1141472	330108625102
19					317980	580559388329
20					62319	837440466326
21					8676	986167409118
22					823	942850011453
23					69	727157075193
24					6	449054224783
25					2	220195944263
26						84941236104
27						25516234965
28						5906586539
29						1042616896
30						139114631
31						13928529
32						1048656
33						59130
34						2846
35						109
36						22
Total	2	8	81	9878	37202839	5431010366322

	# main classes				
	n				
m	1	2	3	4	5
0	1	1	1	1	1
1	1	1	1	1	1
2		2	2	2	2
3		1	5	5	5
4		1	8	18	18
5			9	39	59
6			7	121	256
7			4	253	1224
8			1	442	5997
9			1	495	26188
10				420	94479
11				218	269456
12				96	595649
13				25	1010706
14				8	1304319
15				2	1270356
16				2	923128
17					495565
18					193531
19					54746
20					11052
21					1693
22					192
23					26
24					4
25					2
Total	2	6	39	2148	6239377

Concluding remarks

- Currently, we are waiting for some computations to finish off.

Concluding remarks

- Currently, we are waiting for some computations to finish off.
- We are making an effort to ensure the computations are correct:

Concluding remarks

- Currently, we are waiting for some computations to finish off.
- We are making an effort to ensure the computations are correct:
  - Cross-checking by using two methods (where possible).

Concluding remarks

- Currently, we are waiting for some computations to finish off.
- We are making an effort to ensure the computations are correct:
  - Cross-checking by using two methods (where possible).
  - Comparing with constructive enumeration.

Concluding remarks

- Currently, we are waiting for some computations to finish off.
- We are making an effort to ensure the computations are correct:
  - Cross-checking by using two methods (where possible).
  - Comparing with constructive enumeration.
  - Checking divisors.

Concluding remarks

- Currently, we are waiting for some computations to finish off.
- We are making an effort to ensure the computations are correct:
  - Cross-checking by using two methods (where possible).
  - Comparing with constructive enumeration.
  - Checking divisors.