Minicourse on *information, complexity and organisation in multidimensional symbolic dynamics*

Exact computations of entropy for multidimensional SFT

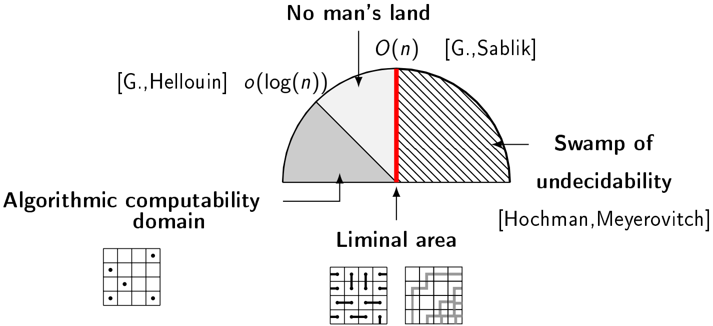
Silvere Gangloff

April 30, 2021

sgangloff@agh.edu.pl ; silvere.gangloff@gmx.com

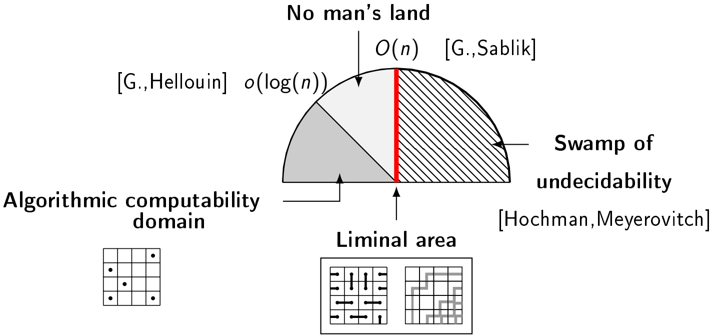
Multidimensional SFT: a computational 'transition':

Reminder (third lecture):



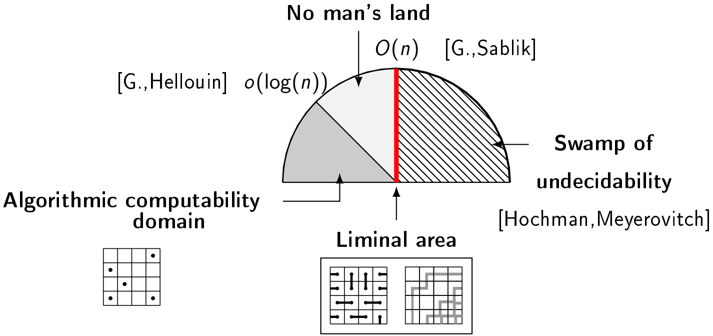
Multidimensional SFT: a computational 'transition':

Reminder (third lecture):



Multidimensional SFT: a computational 'transition':

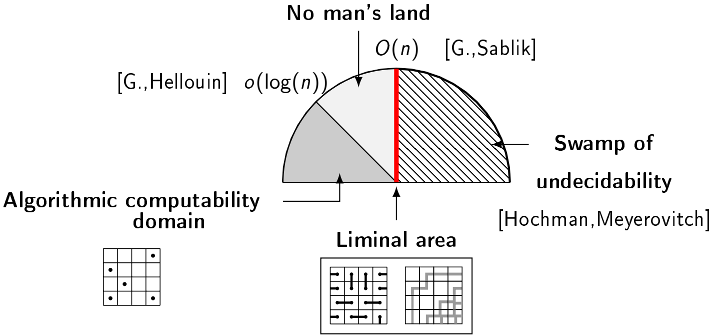
Reminder (third lecture):



In practice, formula for entropy: development of tools for analysis of many variables systems.

Multidimensional SFT: a computational 'transition':

Reminder (third lecture):



In practice, formula for entropy: development of tools for analysis of many variables systems.

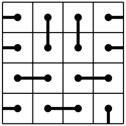
Question: what makes the entropy of subshifts in the liminal area computable ?

Dimers model:

Subshift X_0:

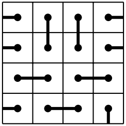
Dimers model:

Subshift X_0:



Dimers model:

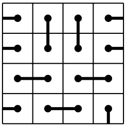
Subshift X_0:



Theorem[Kasteleyn(1961)]: $h(X_0) = \frac{G}{\pi}$,

Dimers model:

Subshift X_0:

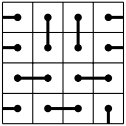


Theorem[Kasteleyn(1961)]: $h(X_0) = \frac{G}{\pi}$, where:

$$G = \sum_{n=0}^{+\infty} \frac{(-1)^n}{(2n+1)^2}.$$

Dimers model:

Subshift X_0:

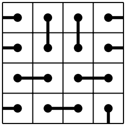


Theorem[Kasteleyn(1961)]: $h(X_0) = \frac{G}{\pi}$, where:

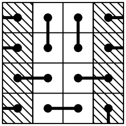
$$G = \sum_{n=0}^{+\infty} \frac{(-1)^n}{(2n+1)^2}.$$

(Called Catalan constant)

Ideas of the proof: Symmetries:

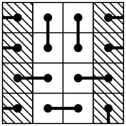


Ideas of the proof: Symmetries:



w

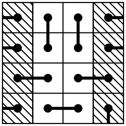
Ideas of the proof: **Symmetries**:



$\mathbf{w}$

Number of such patterns: $N_n^{\mathbf{w}}(X_0)$.

Ideas of the proof: **Symmetries**:

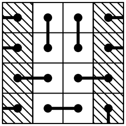


$\mathbf{w}$

Number of such patterns: $N_n^{\mathbf{w}}(X_0)$.

Denote $\overline{\mathbf{w}}$ symmetric of $\mathbf{w}$.

Ideas of the proof: Symmetries:



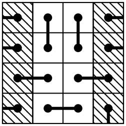
$\mathbf{w}$

Number of such patterns: $N_n^{\mathbf{w}}(X_0)$.

Denote $\overline{\mathbf{w}}$ symmetric of $\mathbf{w}$.

Lemma: $N_n^{\mathbf{w}}(X_0) = N_n^{\overline{\mathbf{w}}}(X_0)$.

Ideas of the proof: **Symmetries**:



$\mathbf{w}$

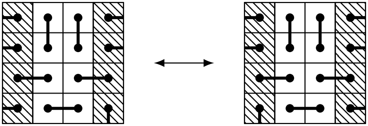
Number of such patterns: $N_n^{\mathbf{w}}(X_0)$.

Denote $\overline{\mathbf{w}}$ symmetric of $\mathbf{w}$.

Lemma: $N_n^{\mathbf{w}}(X_0) = N_n^{\overline{\mathbf{w}}}(X_0)$.

Proof: the symmetry map is an involution.

Ideas of the proof: **Symmetries**:



$\mathbf{w}$ $\qquad\qquad\qquad$ $\overline{\mathbf{w}}$

Number of such patterns: $N_n^{\mathbf{w}}(X_0)$.

Denote $\overline{\mathbf{w}}$ symmetric of $\mathbf{w}$.

Lemma: $N_n^{\mathbf{w}}(X_0) = N_n^{\overline{\mathbf{w}}}(X_0)$.

Proof: the symmetry map is an involution.

Ideas of the proof: **Toric and cylindric entropies**:

Ideas of the proof: Toric and cylindric entropies:

Definition: $h_t(X) \underset{\text{def}}{=} \lim_n \frac{\log_2(N^t_{n,n}(X))}{n^2}$,

Ideas of the proof: **Toric and cylindric entropies**:

Definition: $h_t(X) \underset{\text{def}}{=} \lim_n \frac{\log_2(N^t_{n,n}(X))}{n^2}$,

where $N^t_{m,n}(X)$ is the number of (m,n) rectangle patterns in which rules apply as on torus.

Ideas of the proof: **Toric and cylindric entropies**:

Definition: $h_t(X) \underset{\text{def}}{=} \lim_n \frac{\log_2(N^t_{n,n}(X))}{n^2}$,

where $N^t_{m,n}(X)$ is the number of (m,n) rectangle patterns in which rules apply as on torus.

Similar definitions $h_c(X)$, $N^c_{m,n}(X)$ for cylinders.

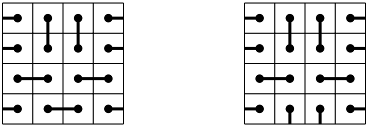
Ideas of the proof: **Toric and cylindric entropies:**

Definition: $h_t(X) \underset{\text{def}}{=} \lim_n \frac{\log_2(N^t_{n,n}(X))}{n^2}$,

where $N^t_{m,n}(X)$ is the number of (m, n) rectangle patterns in which rules apply as on torus.

Similar definitions $h_c(X)$, $N^c_{m,n}(X)$ for cylinders.

Examples:



Ideas of the proof: Toric entropy equals entropy:

Ideas of the proof: Toric entropy equals entropy:

Lemma: $h_t(X_0) = h(X_0)$.

Ideas of the proof: Toric entropy equals entropy:

Lemma: $h_t(X_0) = h(X_0)$.

Ideas of the proof: **Toric entropy equals entropy**:

Lemma: $h_t(X_0) = h(X_0)$.

Proof: we have that:

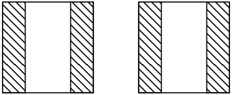
$$\sum_{\mathbf{w}} (N_n^{\mathbf{w}}(X_0))^2 \leq N_{2n,n}^c(X_0) \leq \left(\sum_{\mathbf{w}} N_n^{\mathbf{w}}(X_0) \right)^2 .$$

Ideas of the proof: **Toric entropy equals entropy:**

Lemma: $h_t(X_0) = h(X_0)$.

Proof: we have that:

$$\sum_{\mathbf{w}} (N_n^{\mathbf{w}}(X_0))^2 \leq N_{2n,n}^c(X_0) \leq \left(\sum_{\mathbf{w}} N_n^{\mathbf{w}}(X_0) \right)^2 .$$



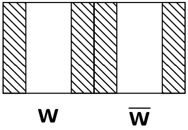
$\mathbf{w}$ $\overline{\mathbf{w}}$

Ideas of the proof: Toric entropy equals entropy:

Lemma: $h_t(X_0) = h(X_0)$.

Proof: we have that:

$$\sum_{\mathbf{w}} (N_n^{\mathbf{w}}(X_0))^2 \leq N_{2n,n}^c(X_0) \leq \left(\sum_{\mathbf{w}} N_n^{\mathbf{w}}(X_0)\right)^2 .$$

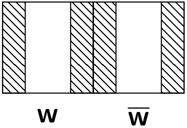


Ideas of the proof: **Toric entropy equals entropy:**

Lemma: $h_t(X_0) = h(X_0)$.

Proof: we have that:

$$\sum_{\mathbf{w}} (N_n^{\mathbf{w}}(X_0))^2 \leq N_{2n,n}^c(X_0) \leq \left(\sum_{\mathbf{w}} N_n^{\mathbf{w}}(X_0) \right)^2 .$$



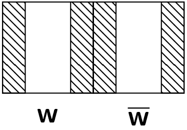
Thus $h_c(X_0) = h(X_0)$.

Ideas of the proof: **Toric entropy equals entropy:**

Lemma: $h_t(X_0) = h(X_0)$.

Proof: we have that:

$$\sum_{\mathbf{w}} (N_n^{\mathbf{w}}(X_0))^2 \leq N_{2n,n}^c(X_0) \leq \left(\sum_{\mathbf{w}} N_n^{\mathbf{w}}(X_0) \right)^2 .$$



Thus $h_c(X_0) = h(X_0)$.

In a similar way $h_t(X_0) = h(X_0)$.

Ideas of the proof:

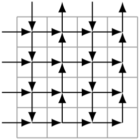
Consider $n \geq 1$;

Ideas of the proof:

Consider $n \geq 1$; arrow pattern

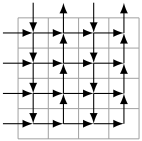
Ideas of the proof:

Consider $n \geq 1$; arrow pattern (ex $n = 4$):



Ideas of the proof:

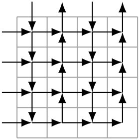
Consider $n \geq 1$; arrow pattern (ex $n = 4$):



Denote $K^{(n)}$, $n^2 \times n^2$ matrix

Ideas of the proof:

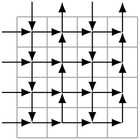
Consider $n \geq 1$; arrow pattern (ex $n = 4$):



Denote $K^{(n)}$, $n^2 \times n^2$ matrix s.t. $K^{(n)}(\mathbf{i}, \mathbf{j}) = \pm 1$ when $\mathbf{i}, \mathbf{j}$ adjacent, otherwise 0.

Ideas of the proof:

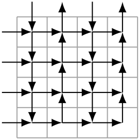
Consider $n \geq 1$; arrow pattern (ex $n = 4$):



Denote $K^{(n)}$, $n^2 \times n^2$ matrix s.t. $K^{(n)}(\mathbf{i},\mathbf{j}) = \pm 1$ when $\mathbf{i},\mathbf{j}$ adjacent, otherwise 0. The coefficient is 1 when $\mathbf{i} \to \mathbf{j}$ and -1 when $\mathbf{i} \leftarrow \mathbf{j}$.

Ideas of the proof:

Consider $n \geq 1$; arrow pattern (ex $n = 4$):

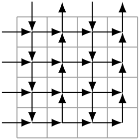


Denote $K^{(n)}$, $n^2 \times n^2$ matrix s.t. $K^{(n)}(\mathbf{i},\mathbf{j}) = \pm 1$ when $\mathbf{i},\mathbf{j}$ adjacent, otherwise 0. The coefficient is 1 when $\mathbf{i} \rightarrow \mathbf{j}$ and -1 when $\mathbf{i} \leftarrow \mathbf{j}$.

$$N_n^t(X_0) = \det(K^{(n)}) = \sum_{\sigma \in S_{n^2}} s(\sigma) \prod_{\mathbf{i}} K^{(n)}_{\mathbf{i},\sigma(\mathbf{i})}$$

Ideas of the proof:

Consider $n \geq 1$; arrow pattern (ex $n = 4$):

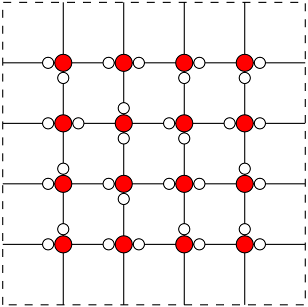


Denote $K^{(n)}$, $n^2 \times n^2$ matrix s.t. $K^{(n)}(\mathbf{i}, \mathbf{j}) = \pm 1$ when $\mathbf{i}, \mathbf{j}$ adjacent, otherwise 0. The coefficient is 1 when $\mathbf{i} \to \mathbf{j}$ and -1 when $\mathbf{i} \leftarrow \mathbf{j}$.

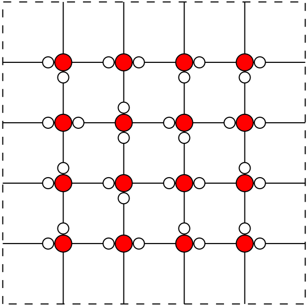
$$N_n^t(X_0) = \det(K^{(n)}) = \sum_{\sigma \in S_{n^2}} s(\sigma) \prod_{\mathbf{i}} K^{(n)}_{\mathbf{i}, \sigma(\mathbf{i})}$$

Diagonalisation of $K^{(n)} \to$ formula for $N_n^t(X_0)$ as sum of trigonometric functions.

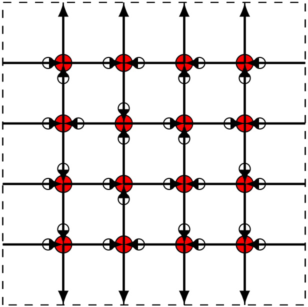
Square ice: Wang tiles representation:



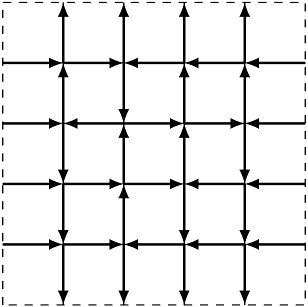
Square ice: Wang tiles representation:



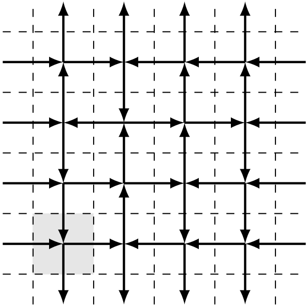
Square ice: **Wang tiles representation**:



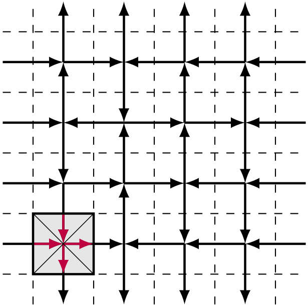
Square ice: Wang tiles representation:



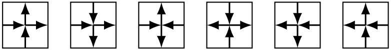
Square ice: Wang tiles representation:

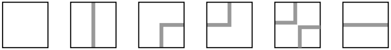


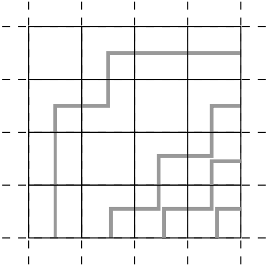
Square ice: Wang tiles representation:



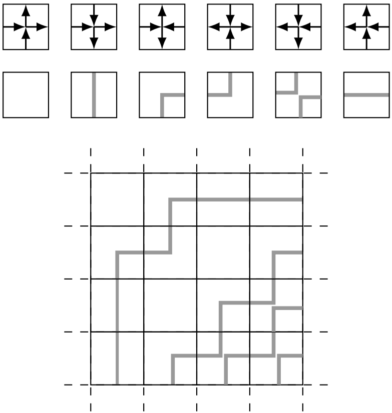
Square ice: Subshift X^s:





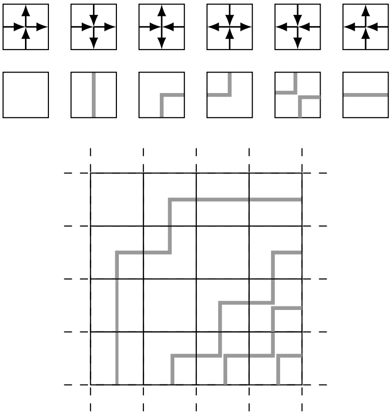


Square ice: Subshift X^s:



E.H. Lieb, *Residual entropy of square ice*, Physical Review, 1967.

Square ice: Subshift X^s:



E.H. Lieb, *Residual entropy of square ice*, Physical Review, 1967.

S. Gangloff, *A proof that square ice entropy is* $\frac{3}{2}\log_2(4/3)$, 2019 (based on the work of R.Baxter, K.Kozlowski).

Square ice: Computing entropy using stripes subshifts:

$$h(X) \underset{\text{def}}{=} \lim_{n} \frac{\log_2(N_n(X))}{n^2}.$$

Square ice: Computing entropy using stripes subshifts:

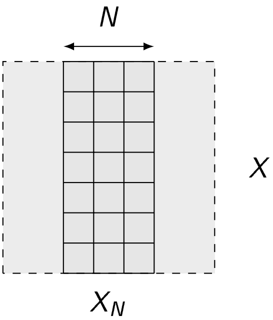
$$h(X) \underset{\text{def}}{=} \lim_{n} \frac{\log_2(N_n(X))}{n^2}.$$

$$h(X) = \lim_{m,n} \frac{\log_2(N_{m,n}(X))}{mn}.$$

Square ice: Computing entropy using stripes subshifts:

$$h(X) \underset{\text{def}}{=} \lim_n \frac{\log_2(N_n(X))}{n^2}.$$

$$h(X) = \lim_{m,n} \frac{\log_2(N_{m,n}(X))}{mn}.$$

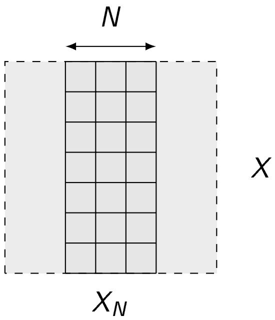


Square ice: Computing entropy using stripes subshifts:

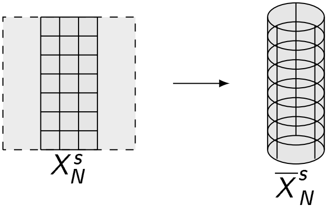
$$h(X) \underset{\text{def}}{=} \lim_{n} \frac{\log_2(N_n(X))}{n^2}.$$

$$h(X) = \lim_{m,n} \frac{\log_2(N_{m,n}(X))}{mn}.$$

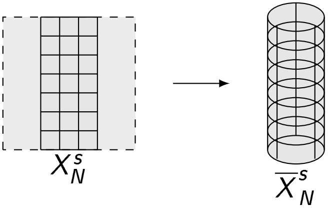
$$h(X) = \lim_N \frac{h(X_N)}{N}$$



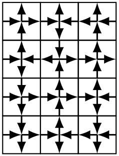
Square ice: Cylindric stripes subshifts:



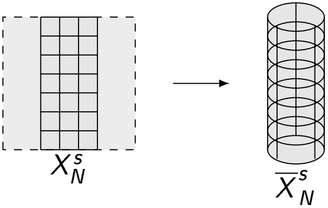
Square ice: Cylindric stripes subshifts:



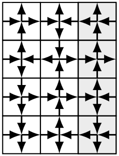
Symmetry properties:



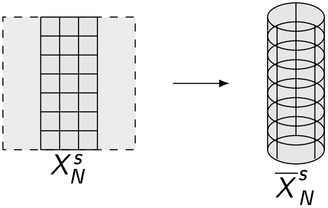
Square ice: Cylindric stripes subshifts:



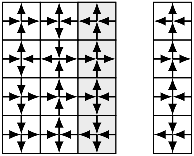
Symmetry properties:



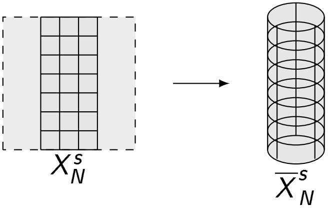
Square ice: Cylindric stripes subshifts:



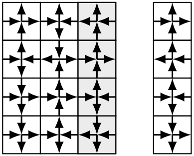
Symmetry properties:



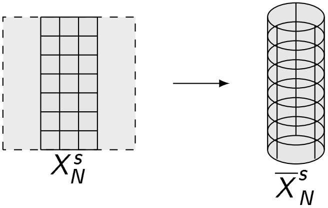
Square ice: Cylindric stripes subshifts:



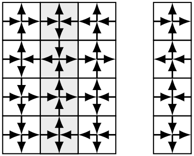
Symmetry properties:



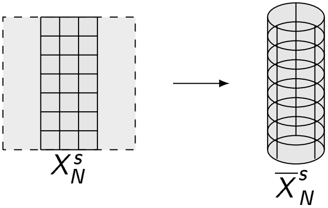
Square ice: Cylindric stripes subshifts:



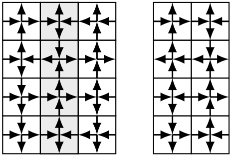
Symmetry properties:



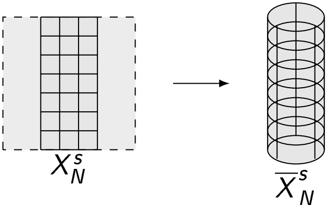
Square ice: Cylindric stripes subshifts:



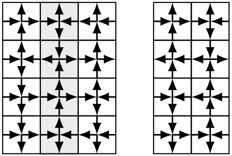
Symmetry properties:



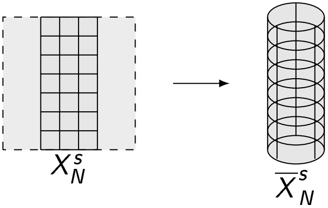
Square ice: Cylindric stripes subshifts:



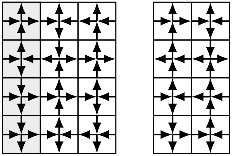
Symmetry properties:



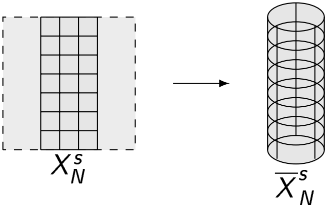
Square ice: Cylindric stripes subshifts:



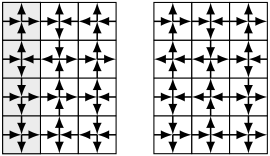
Symmetry properties:



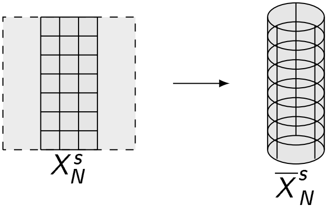
Square ice: Cylindric stripes subshifts:



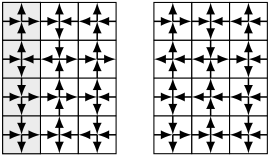
Symmetry properties:



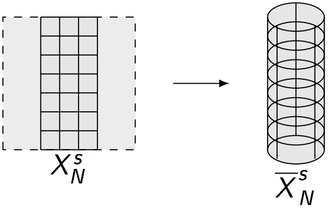
Square ice: Cylindric stripes subshifts:



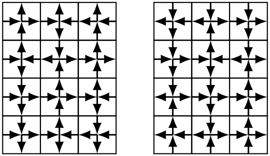
Symmetry properties:



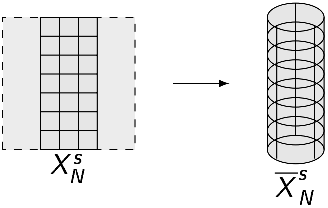
Square ice: Cylindric stripes subshifts:



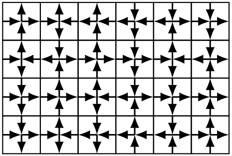
Symmetry properties:



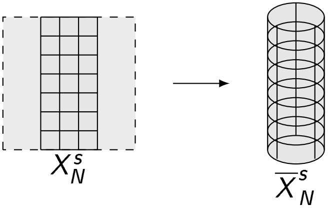
Square ice: Cylindric stripes subshifts:



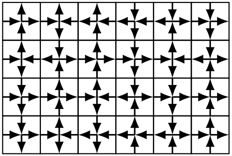
Symmetry properties:



Square ice: Cylindric stripes subshifts:

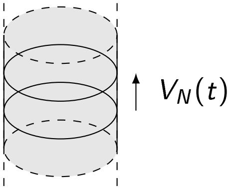


Symmetry properties:

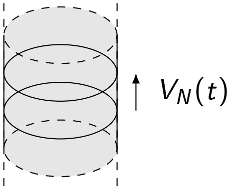


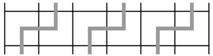
$$h(X^s) = \lim_N \frac{h(\overline{X}_N^s)}{N}$$

Square ice: Lieb transfer matrices:

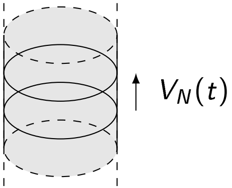


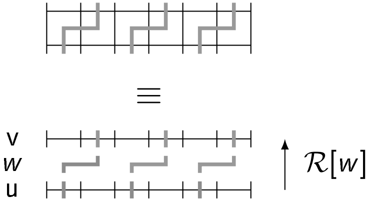
Square ice: Lieb transfer matrices:



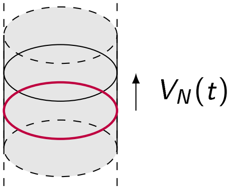


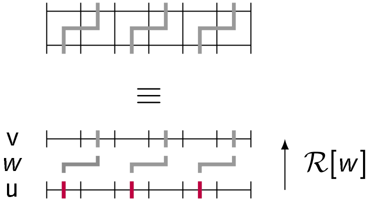
Square ice: Lieb transfer matrices:



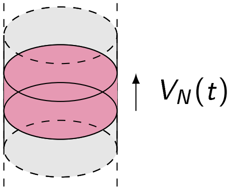


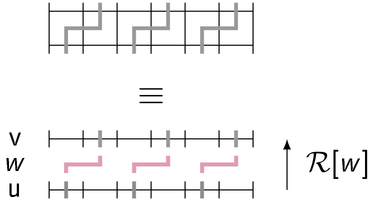
Square ice: Lieb transfer matrices:



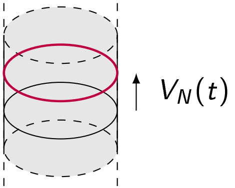


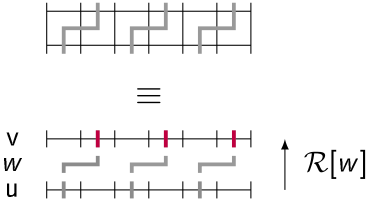
Square ice: **Lieb transfer matrices:**



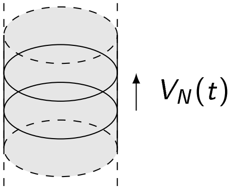


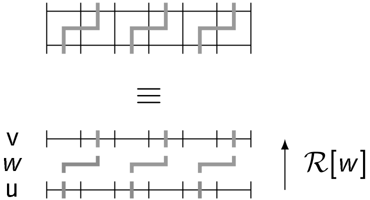
Square ice: Lieb transfer matrices:





Square ice: Lieb transfer matrices:

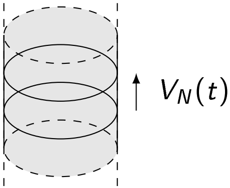


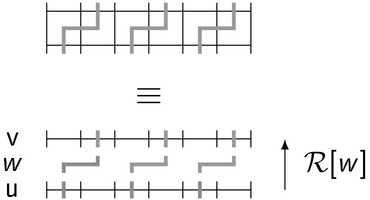


$$V_N(t)[\mathrm{u},\mathrm{v}] = \sum_{\mathrm{u}\mathcal{R}[w]\mathrm{v}} t^{|w|}.$$

where $|w| = \#$ of  and 

Square ice: Lieb transfer matrices:





$$V_N(t)[\mathrm{u},\mathrm{v}] = \sum_{\mathrm{u}\mathcal{R}[w]\mathrm{v}} t^{|w|}.$$

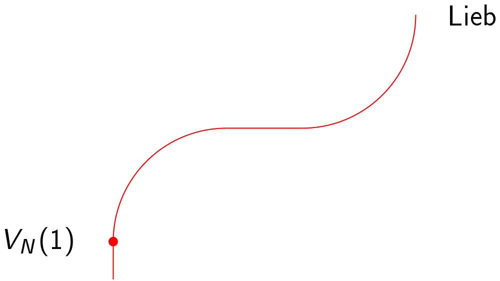
where $|w| = \#$ of  and 

$$h(X^s) = \lim_N \frac{\log_2(\lambda_{\max}(V_N(1)))}{N}$$

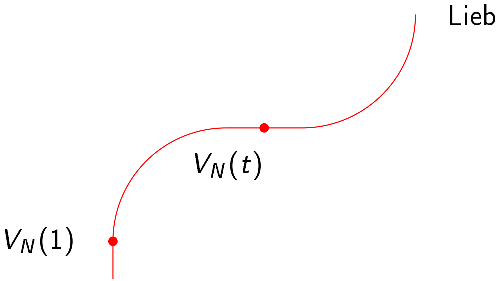
Square ice: Computing maximal eigenvalue of $V_N(1)$, strategy:

$V_N(1)$ •

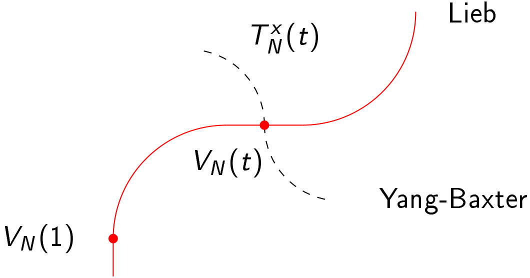
Square ice: Computing maximal eigenvalue of $V_N(1)$, strategy:



Square ice: Computing maximal eigenvalue of $V_N(1)$, strategy:

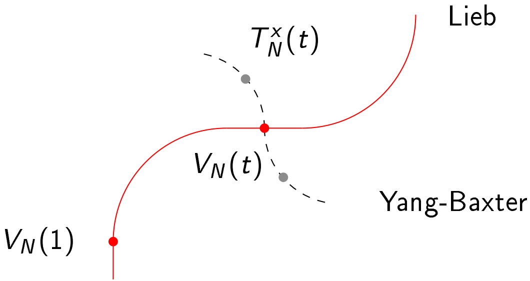


Square ice: **Computing maximal eigenvalue of $V_N(1)$, strategy:**



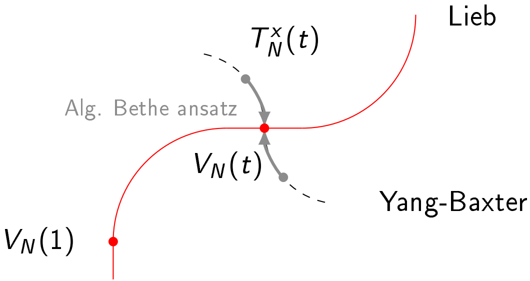
$$\forall x, y,\ T_N^y(t)\,T_N^x(t) = T_N^y(t)\,T_N^x(t).$$

Square ice: **Computing maximal eigenvalue of $V_N(1)$, strategy:**



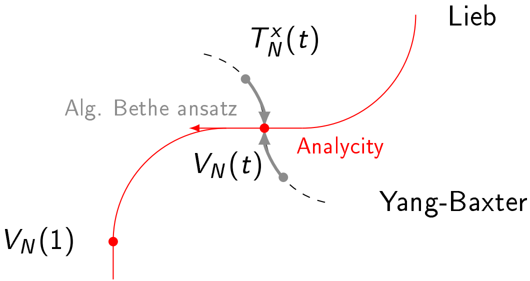
$$\forall x, y,\ T_N^y(t)\,T_N^x(t) = T_N^y(t)\,T_N^x(t).$$

Square ice: **Computing maximal eigenvalue of $V_N(1)$, strategy:**



$$\forall x, y,\ T_N^y(t)\,T_N^x(t) = T_N^y(t)\,T_N^x(t).$$

Square ice: Computing maximal eigenvalue of $V_N(1)$, strategy:



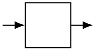
$$\forall x, y,\ T_N^y(t)\,T_N^x(t) = T_N^y(t)\,T_N^x(t).$$

Square ice: R-matrices and monodromy matrices:

Square ice: R-matrices and monodromy matrices: $\rightarrow$:
input/output:

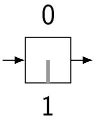
Square ice: **R-matrices and monodromy matrices:** $\rightarrow$:

input/output:



Square ice: R-matrices and monodromy matrices: $\to$:

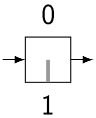
input/output:



$$R(0,1) = \begin{pmatrix} & \\ & \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:

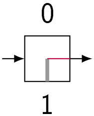
input/output:



$$R(0,1) = \begin{pmatrix} 0 & \\ & \end{pmatrix}$$

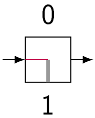
Square ice: R-matrices and monodromy matrices: $\rightarrow$:

input/output:



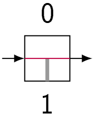
$$R(0,1) = \begin{pmatrix} 0 & * \\ & \end{pmatrix}$$

Square ice: **R-matrices and monodromy matrices:** $\rightarrow$:
input/output:



$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & \end{pmatrix}$$

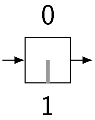
Square ice: **R-matrices and monodromy matrices:** $\rightarrow$:
input/output:



$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

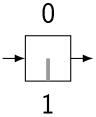
Square ice: R-matrices and monodromy matrices: $\rightarrow$:

input/output:



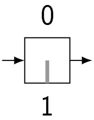
$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

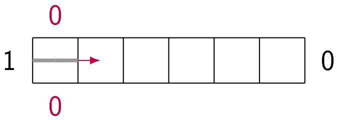
Square ice: **R-matrices and monodromy matrices:** $\rightarrow$:
input/output:



$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:
input/output:

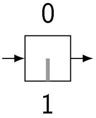


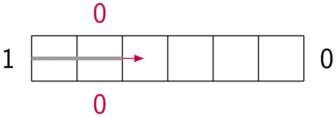


$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:

input/output:

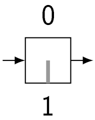


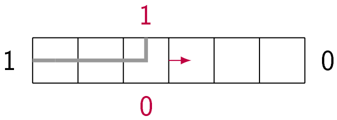


$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:

input/output:

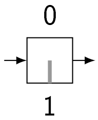


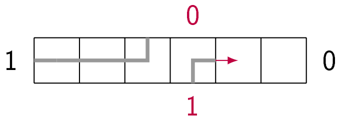


$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:

input/output:

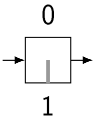


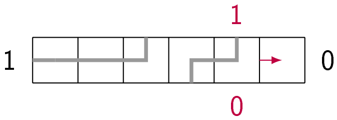


$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:

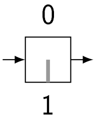
input/output:

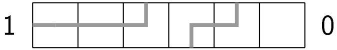




$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

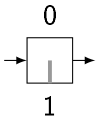
Square ice: R-matrices and monodromy matrices: $\rightarrow$: input/output:





$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

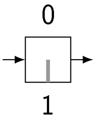
Square ice: R-matrices and monodromy matrices: $\rightarrow$: input/output:

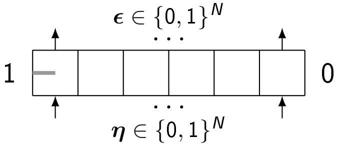


$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$:

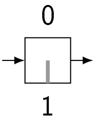
input/output:



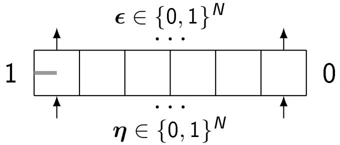


$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$: input/output:

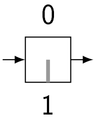


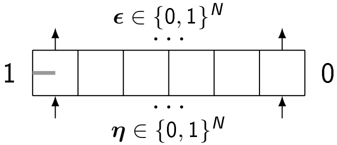
$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix}$$



$$M_N(1,0)[\epsilon, \eta]$$

Square ice: R-matrices and monodromy matrices: $\rightarrow$: input/output:





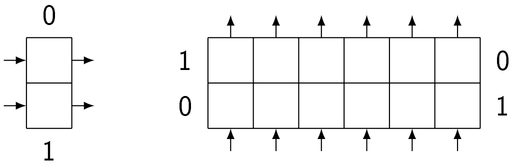
$$R(0,1) = \begin{pmatrix} 0 & * \\ 0 & 0 \end{pmatrix} \qquad M_N(1,0)[\epsilon, \eta]$$

Yang-Baxter transfer matrices:

$$T_N[\epsilon, \eta] = \sum_{u \in \{0,1\}} M_N(u,u)[\epsilon, \eta].$$

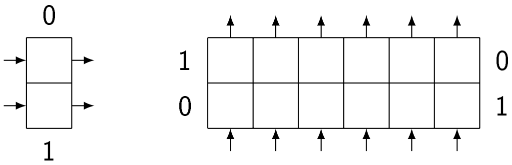
Square ice: Commutation of Yang-Baxter matrices:

Composition of R-matrices and monodromy matrices:

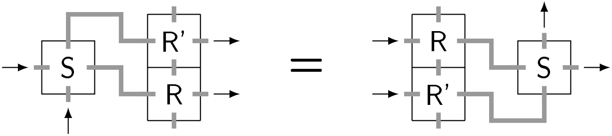


Square ice: Commutation of Yang-Baxter matrices:

Composition of R-matrices and monodromy matrices:

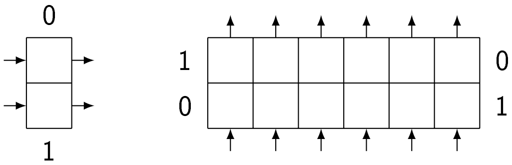


Yang-Baxter equation:

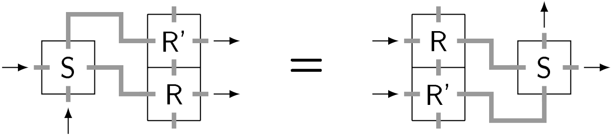


Square ice: Commutation of Yang-Baxter matrices:

Composition of R-matrices and monodromy matrices:



Yang-Baxter equation:



Yang-Baxter equation $\Rightarrow$ transfer matrices commute.

Square ice: **Trigonometric R-matrices:** $t \in (0, \sqrt{2}]$.

Square ice: Trigonometric R-matrices: $t \in (0, \sqrt{2}]$.

Denote $\mu_t \in \left(0, \frac{\pi}{2}\right)$ s.t. $\cos(\mu_t) = \frac{2-t^2}{2}$.

Square ice: Trigonometric R-matrices: $t \in (0, \sqrt{2}]$.

Denote $\mu_t \in \left(0, \frac{\pi}{2}\right)$ s.t. $\cos(\mu_t) = \frac{2-t^2}{2}$.

$$R^x_{\mu_t} = \frac{1}{\sin(\mu_t/2)} \begin{pmatrix} \sin(\mu_t - x) & 0 & 0 & 0 \\ 0 & \sin(x) & \sin(\mu_t) & 0 \\ 0 & \sin(\mu_t) & \sin(x) & 0 \\ 0 & 0 & 0 & \sin(\mu_t - x) \end{pmatrix}.$$

Square ice: Trigonometric R-matrices: $t \in (0, \sqrt{2}]$.

Denote $\mu_t \in \left(0, \frac{\pi}{2}\right)$ s.t. $\cos(\mu_t) = \frac{2-t^2}{2}$.

$$R^x_{\mu_t} = \frac{1}{\sin(\mu_t/2)} \begin{pmatrix} \sin(\mu_t - x) & 0 & 0 & 0 \\ 0 & \sin(x) & \sin(\mu_t) & 0 \\ 0 & \sin(\mu_t) & \sin(x) & 0 \\ 0 & 0 & 0 & \sin(\mu_t - x) \end{pmatrix}.$$

where $R^x_{\mu_t}(0,0)$ is the up-left 2×2 part of this matrix, etc.

Square ice: Bethe ansatz:

Square ice: Bethe ansatz:

If there exists $(p_j(t))_{j=1}^n$ solution of:

$$Np_j = 2\pi j - (n+1)\pi - \sum_{k=1}^{n} \Theta_t(p_j, p_k) \qquad (E_t)$$

Square ice: Bethe ansatz:

If there exists $(p_j(t))_{j=1}^n$ solution of:

$$Np_j = 2\pi j - (n+1)\pi - \sum_{k=1}^{n} \Theta_t(p_j, p_k) \qquad (E_t)$$

construction of a candidate eigenvector for the transfer matrix for value:

$$\prod_{k=1}^{n} L_t(e^{ip_k}) + \prod_{k=1}^{n} M_t(e^{ip_k}).$$

where L_t and M_t are simple rational functions.

Square ice: Bethe ansatz:

If there exists $(p_j(t))_{j=1}^n$ solution of:

$$Np_j = 2\pi j - (n+1)\pi - \sum_{k=1}^{n} \Theta_t(p_j, p_k) \qquad (E_t)$$

construction of a candidate eigenvector for the transfer matrix for value:

$$\prod_{k=1}^{n} L_t(e^{ip_k}) + \prod_{k=1}^{n} M_t(e^{ip_k}).$$

where L_t and M_t are simple rational functions.
Lemma: for all t there exists $(p_j(t))_{j=1}^n$ solution of (E_t) and it is an analytic function in t.

Square ice: Identification to $\lambda_{\mathbf{max}}(V_N(t))$: ingredients:

Square ice: Identification to $\lambda_{\mathbf{max}}(V_N(t))$: ingredients:

1. When $t = \sqrt{2}$, simplified expression of candidate eigenvector and eigenvalue.

Square ice: **Identification to $\lambda_{\mathbf{max}}(V_N(t))$:** ingredients:

1. When $t = \sqrt{2}$, simplified expression of candidate eigenvector and eigenvalue.
2. There is H_N (given by derivative of $V_N(t)$) diagonalisable s.t. $V_N(\sqrt{2})H_N = H_N V_N(\sqrt{2})$: candidate value is the maximal eigenvalue.

Square ice: Identification to $\lambda_{\mathbf{max}}(V_N(t))$: ingredients:

1. When $t = \sqrt{2}$, simplified expression of candidate eigenvector and eigenvalue.
2. There is H_N (given by derivative of $V_N(t)$) diagonalisable s.t. $V_N(\sqrt{2})H_N = H_N V_N(\sqrt{2})$: candidate value is the maximal eigenvalue.
3. Perron-Frobenius: candidate eigenvector has positive coordinates $\rightarrow$ eigenvector for the maximal eigenvalue.

Square ice: Identification to $\lambda_{\mathbf{max}}(V_N(t))$: ingredients:

1. When $t = \sqrt{2}$, simplified expression of candidate eigenvector and eigenvalue.
2. There is H_N (given by derivative of $V_N(t)$) diagonalisable s.t. $V_N(\sqrt{2})H_N = H_N V_N(\sqrt{2})$: candidate value is the maximal eigenvalue.
3. Perron-Frobenius: candidate eigenvector has positive coordinates $\rightarrow$ eigenvector for the maximal eigenvalue.
4. Indentification of maximal eigenvalue and eigeinvector around $\sqrt{2}$ (positive coordinates), then on $(0, \sqrt{2})$ by analycity.

Square ice: Asymptotics:

Square ice: **Asymptotics:** consider $n_k/N_k \to 1/2$.

Square ice: Asymptotics: consider $n_k/N_k \to 1/2$.

Denote $(\mathbf{p}_j^{(k))})_j$ solutions of equations (E_t) for N_k, n_k.

Square ice: Asymptotics: consider $n_k/N_k \to 1/2$.

Denote $(\mathbf{p}_j^{(k))})_j$ solutions of equations (E_t) for N_k, n_k.

Change of variable $(\mathbf{p}_j^{(k))})_j = (\kappa_t(\boldsymbol{\alpha}_j^{(k)}))_j$.

Square ice: Asymptotics: consider $n_k/N_k \to 1/2$.

Denote $(\mathbf{p}_j^{(k))})_j$ solutions of equations (E_t) for N_k, n_k.

Change of variable $(\mathbf{p}_j^{(k))})_j = (\kappa_t(\boldsymbol{\alpha}_j^{(k)}))_j$.

Lemma:

$$\lim_N \frac{\log_2(\lambda_{\max}(V_N(1)))}{N} = \lim_k \frac{1}{N_k} \sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}),$$

with $f(x) = \log_2(2|\sin(\kappa_t(x))/2|)$.

Square ice: Asymptotics: consider $n_k/N_k \to 1/2$.

Denote $(\mathbf{p}_j^{(k))})_j$ solutions of equations (E_t) for N_k, n_k.

Change of variable $(\mathbf{p}_j^{(k))})_j = (\kappa_t(\boldsymbol{\alpha}_j^{(k)}))_j$.

Lemma:

$$\lim_N \frac{\log_2(\lambda_{\max}(V_N(1)))}{N} = \lim_k \frac{1}{N_k} \sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}),$$

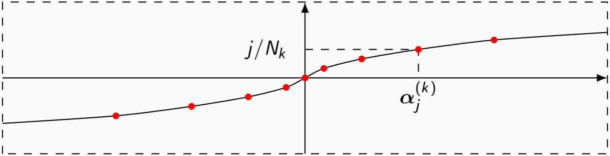
with $f(x) = \log_2(2|\sin(\kappa_t(x))/2|)$.

Theorem: there exists ρ_t s.t. for all $f \in L^1$:

$$\lim_k \frac{1}{N_k} \sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}) = \int_{\mathbb{R}} f(\alpha)\rho_t(\alpha)d\alpha.$$

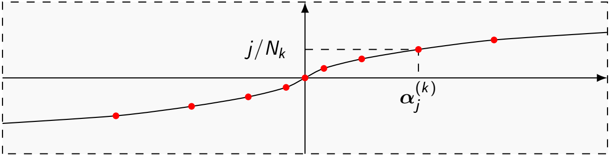
Square ice: Counting functions:

$$\xi_t^{(k)} : \alpha \mapsto \frac{1}{2\pi}\kappa_t(\alpha) + \frac{n_k + 1}{2N_k} + \frac{1}{2\pi N_k}\sum_{j=1}^{n_k} \theta_t(\alpha, \boldsymbol{\alpha}_j^{(k)})$$



Square ice: Counting functions:

$$\xi_t^{(k)} : \alpha \mapsto \frac{1}{2\pi}\kappa_t(\alpha) + \frac{n_k+1}{2N_k} + \frac{1}{2\pi N_k}\sum_{j=1}^{n_k} \theta_t(\alpha, \boldsymbol{\alpha}_j^{(k)})$$



Equality with a Riemann sum:

$$\lim_k \frac{1}{N_k}\sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}) = \lim_k \frac{1}{N_k}\sum_{j=1}^{n_k} \left(\boldsymbol{\alpha}_{j+1}^{(k)} - \boldsymbol{\alpha}_j^{(k)}\right) \frac{\left(\xi_t^{(k)}(\boldsymbol{\alpha}_{j+1}^{(k)}) - \xi_t^{(k)}(\boldsymbol{\alpha}_j^{(k)})\right)}{\left(\boldsymbol{\alpha}_{j+1}^{(k)} - \boldsymbol{\alpha}_j^{(k)}\right)} f(\boldsymbol{\alpha}_j^{(k)})$$

Square ice: Asymptotics of counting functions:

Square ice: Asymptotics of counting functions:

Square ice: Asymptotics of counting functions:

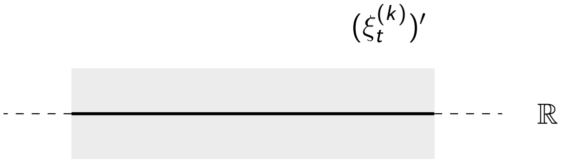
1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:

Square ice: Asymptotics of counting functions:

1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:

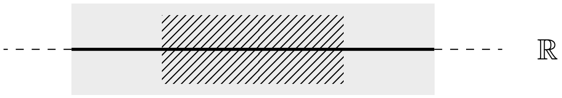
Square ice: Asymptotics of counting functions:

1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:



Square ice: Asymptotics of counting functions:

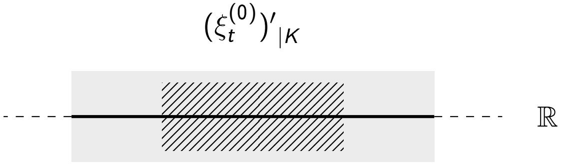
1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:



2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.

Square ice: Asymptotics of counting functions:

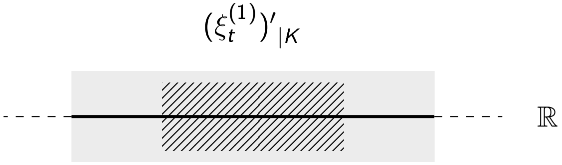
1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:



2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.

Square ice: Asymptotics of counting functions:

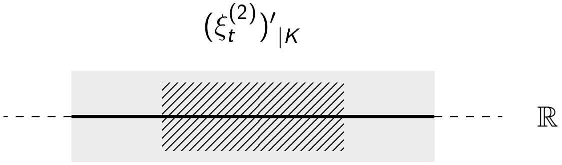
1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:



2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.

Square ice: Asymptotics of counting functions:

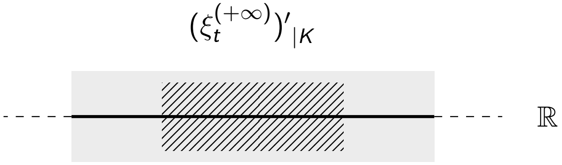
1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:



2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.

Square ice: Asymptotics of counting functions:

1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:

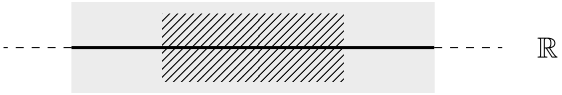


2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.

Square ice: **Asymptotics of counting functions:**

1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:

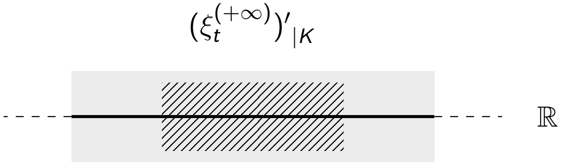
$$(\xi_t^{(+\infty)})'_{|K}$$



2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.
3. F_t satisfies an integral equation with unique solution.

Square ice: Asymptotics of counting functions:

1. Extend $(\xi_t^{(k)})'$ on a stripe including $\mathbb{R}$:



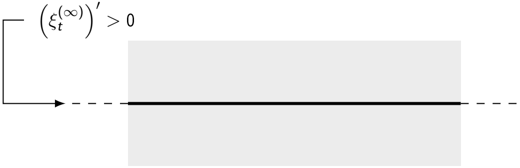
2. Assume $((\xi_t)^{\nu(k)})' \to F_t$ on any compact K.
3. F_t satisfies an integral equation with unique solution.
4. Thus, $(\xi_t^{(k)})'$ converges to this solution.

Square ice: Rarefaction of the roots and $\xi_t^{(k)}$

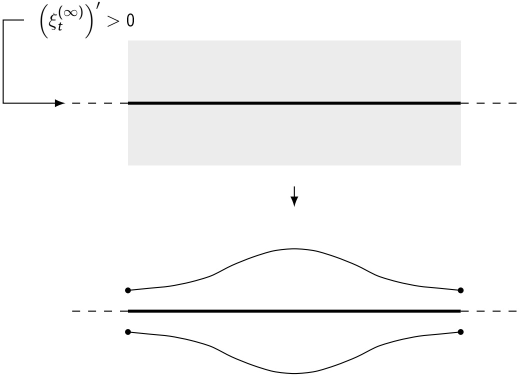
biholomorphisms: $\epsilon > 0$:

Square ice: Rarefaction of the roots and $\xi_t^{(k)}$

biholomorphisms: $\epsilon > 0$:

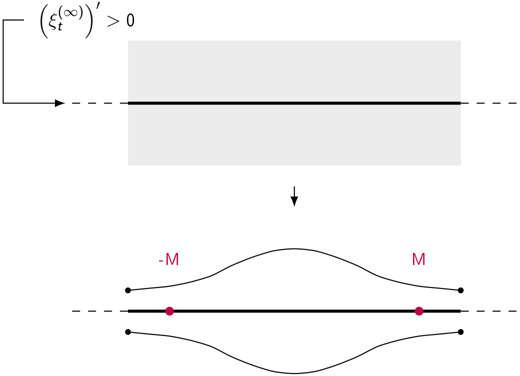


Square ice: Rarefaction of the roots and $\xi_t^{(k)}$ biholomorphisms: $\epsilon > 0$:



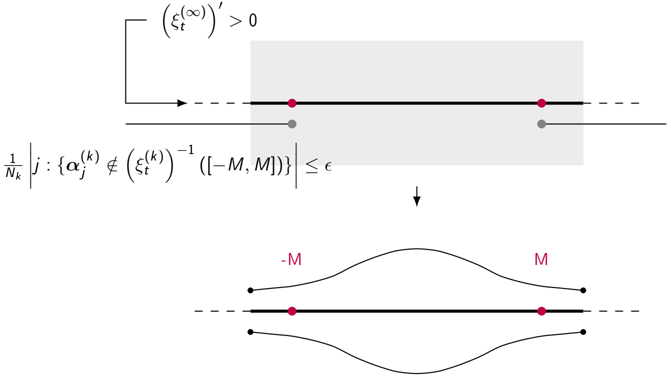
Square ice: Rarefaction of the roots and $\xi_t^{(k)}$

biholomorphisms: $\epsilon > 0$:

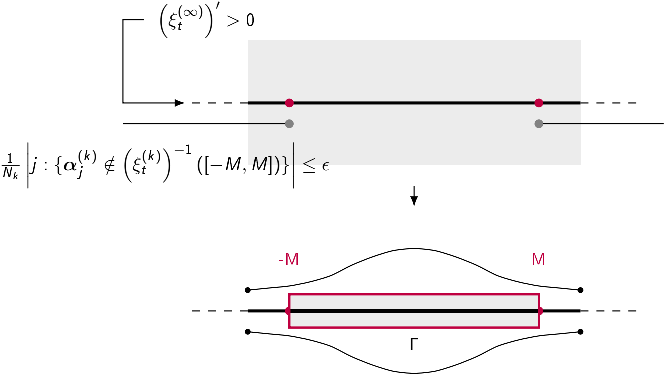


Square ice: Rarefaction of the roots and $\xi_t^{(k)}$

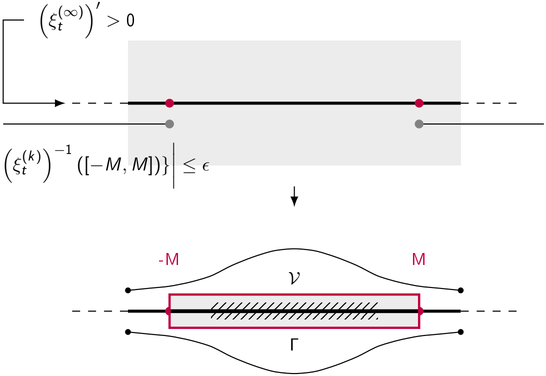
biholomorphisms: $\epsilon > 0$:



Square ice: Rarefaction of the roots and $\xi_t^{(k)}$ biholomorphisms: $\epsilon > 0$:



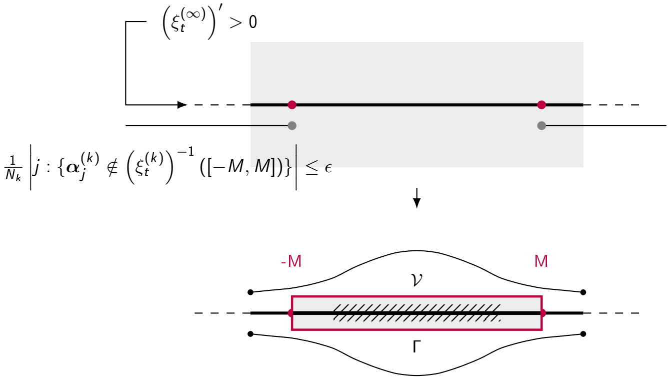
Square ice: Rarefaction of the roots and $\xi_t^{(k)}$ biholomorphisms: $\epsilon > 0$:



$$\frac{1}{N_k}\left|j : \{\boldsymbol{\alpha}_j^{(k)} \notin \left(\xi_t^{(k)}\right)^{-1}([-M, M])\}\right| \leq \epsilon$$

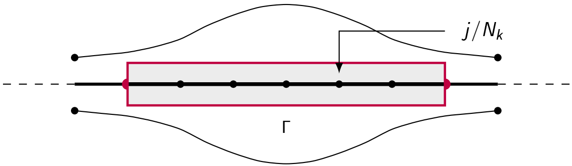
Square ice: Rarefaction of the roots and $\xi_t^{(k)}$

biholomorphisms: $\epsilon > 0$:

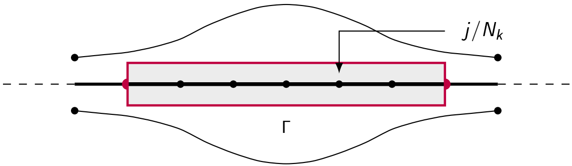


The functions $\xi_t^{(k)}$ have distinct values on $\mathcal{V}$ and Γ. Thus they are bihilomorphisms onto $\mathcal{V}$ (Cauchy formula).

Square ice: Lace integral expression of $\xi_t^{(k)}$:



Square ice: Lace integral expression of $\xi_t^{(k)}$:



By residues theorem:

$$\xi_t^{(k)}(\alpha) = \frac{1}{2\pi}\kappa_t(\alpha) + \frac{n_k+1}{2N_k} + \oint_\Gamma \theta_t\left(\left(\xi_t^{(k)}\right)^{-1}(\alpha)\right)\frac{e^{2i\pi s N_k}}{e^{2i\pi s N_k}-1}ds + O(\epsilon).$$

Square ice: Fredholm integral equation: Limit and change of variable:

$$\xi_t^{(\infty)}(\alpha) = \frac{1}{2\pi}\kappa_t(\alpha) + \frac{1}{4} + \int_0^{+\infty} \theta_t(\alpha) \left(\xi_t^{(\infty)}\right)'(\alpha) d\alpha.$$

Square ice: Fredholm integral equation: Limit and change of variable:

$$\xi_t^{(\infty)}(\alpha) = \frac{1}{2\pi}\kappa_t(\alpha) + \frac{1}{4} + \int_0^{+\infty} \theta_t(\alpha)\left(\xi_t^{(\infty)}\right)'(\alpha)d\alpha.$$

Unique solution by Fourier transforms.

Square ice: Final computation:

Square ice: Final computation:

$$\lim_N \frac{\log_2(\lambda_{\max}(V_N(1)))}{N} = \lim_k \frac{1}{N_k} \sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}),$$

with $f(x) = \log_2(2|\sin(\kappa_t(x))/2|)$.

Square ice: Final computation:

$$\lim_N \frac{\log_2(\lambda_{\max}(V_N(1)))}{N} = \lim_k \frac{1}{N_k} \sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}),$$

with $f(x) = \log_2(2|\sin(\kappa_t(x))/2|)$.

$$h(X^s) = \int_{\mathbb{R}} \log_2(2|\sin(\kappa_t(\alpha))/2|).\rho_t(\alpha) d\alpha.$$

Square ice: Final computation:

$$\lim_N \frac{\log_2(\lambda_{\max}(V_N(1)))}{N} = \lim_k \frac{1}{N_k} \sum_{j=1}^{n_k} f(\boldsymbol{\alpha}_j^{(k)}),$$

with $f(x) = \log_2(2|\sin(\kappa_t(x))/2|)$.

$$h(X^s) = \int_{\mathbb{R}} \log_2(2|\sin(\kappa_t(\alpha))/2|).\rho_t(\alpha)d\alpha.$$

Expression of $\rho_t = \left(\xi_t^{(\infty)}\right)'$ and lace integrals computations:

$$\boxed{h(X^s) = \frac{3}{2}\log_2(4/3)}.$$

Friedland's theorem:

Theorem[Friedland(1967)]: if the set of forbidden patterns $\mathcal{F}$ is stable by symmetry, $h(X_{\mathcal{F}})$ is a computable number.

Friedland's theorem:

Theorem[Friedland(1967)]: if the set of forbidden patterns $\mathcal{F}$ is stable by symmetry, $h(X_{\mathcal{F}})$ is a computable number.

Examples: dimers, square ice, hard squares.

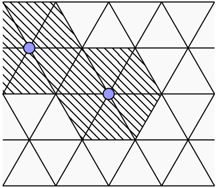
Friedland's theorem:

Theorem[Friedland(1967)]: if the set of forbidden patterns $\mathcal{F}$ is stable by symmetry, $h(X_{\mathcal{F}})$ is a computable number.

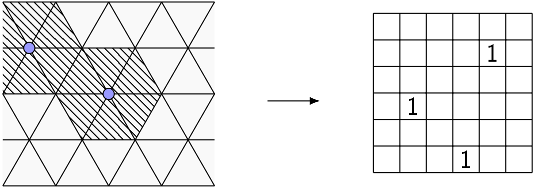
Examples: dimers, square ice, hard squares.

Question: what are the possible values of entropy for symmetric bidimensional subshifts of finite type ?

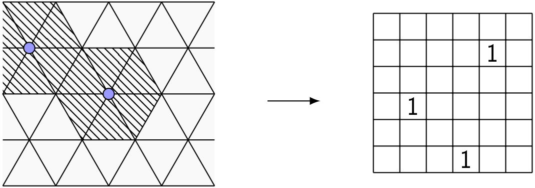
Baxter's hard hexagons:[Exactly solvable models in statistical physics]



Baxter's hard hexagons: [Exactly solvable models in statistical physics]

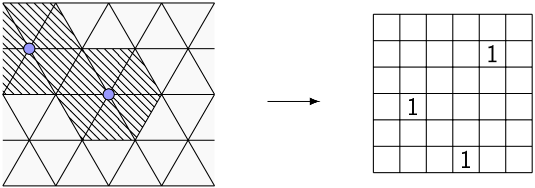


Baxter's hard hexagons:[Exactly solvable models in statistical physics]



Formula for entropy as sum of a series:

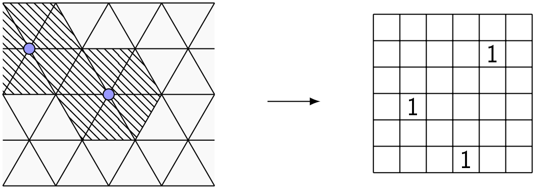
Baxter's hard hexagons:[Exactly solvable models in statistical physics]



Formula for entropy as sum of a series:

1. Transfer matrices $\to$ **diagonal** transfer matrices;

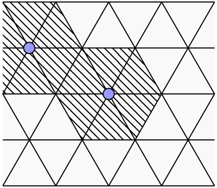
Baxter's hard hexagons:[Exactly solvable models in statistical physics]

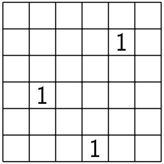


Formula for entropy as sum of a series:

1. Transfer matrices $\rightarrow$ **diagonal** transfer matrices;
2. Obtain a functional equation;

Baxter's hard hexagons:[Exactly solvable models in statistical physics]

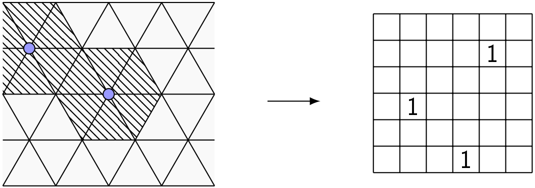




Formula for entropy as sum of a series:

1. Transfer matrices $\rightarrow$ **diagonal** transfer matrices;
2. Obtain a functional equation;
3. Maximal eigenvalue satisfies the same functional equation;

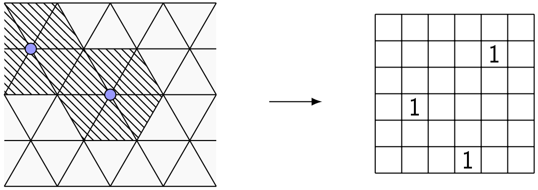
Baxter's hard hexagons: [Exactly solvable models in statistical physics]



Formula for entropy as sum of a series:

1. Transfer matrices $\rightarrow$ **diagonal** transfer matrices;
2. Obtain a functional equation;
3. Maximal eigenvalue satisfies the same functional equation;
4. Maximal egeinvalue equal for the two diagonal transfer matrices $\rightarrow$ other function equation.

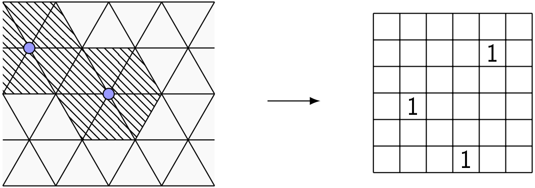
Baxter's hard hexagons:[Exactly solvable models in statistical physics]



Formula for entropy as sum of a series:

1. Transfer matrices $\rightarrow$ **diagonal** transfer matrices;
2. Obtain a functional equation;
3. Maximal eigenvalue satisfies the same functional equation;
4. Maximal egeinvalue equal for the two diagonal transfer matrices $\rightarrow$ other function equation.
5. System of equations with unique solution: evaluate it in specific parameter.

Baxter's hard hexagons:[Exactly solvable models in statistical physics]

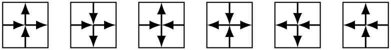


Formula for entropy as sum of a series:

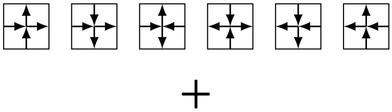
1. Transfer matrices → **diagonal** transfer matrices;
2. Obtain a functional equation;
3. Maximal eigenvalue satisfies the same functional equation;
4. Maximal egeinvalue equal for the two diagonal transfer matrices → other function equation.
5. System of equations with unique solution: evaluate it in specific parameter.

Main problems: points **3**, **4**.

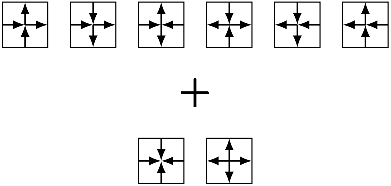
Eight-vertex model:[Exactly solvable models in statistical physics]



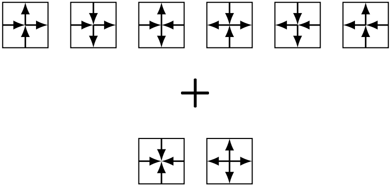
Eight-vertex model:[Exactly solvable models in statistical physics]



Eight-vertex model: [Exactly solvable models in statistical physics]



Eight-vertex model:[Exactly solvable models in statistical physics]



Entropy computation: similar to square ice; analytical part non verified.

Subsidiary questions:

Question: can we use similar methods to talk about invariant measures (for instance $\times 2$, $\times 3$ conjecture) ?

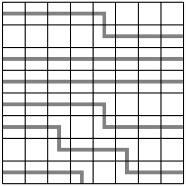
Subsidiary questions:

Question: can we use similar methods to talk about invariant measures (for instance $\times 2, \times 3$ conjecture) ?

Question: can we find solutions of Yang-Baxter equations for other subshifts of finite type ? *Example*: Kari-Culik tilings (know: positive entropy [Durand,Gamard,Grandjean(2017)]).

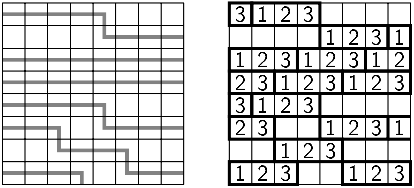
Combinatorial methods

Definition subshift Δ_r; ex for $r = 3$:



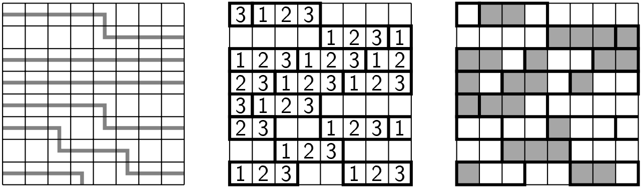
Combinatorial methods

Definition subshift Δ_r; ex for $r = 3$:



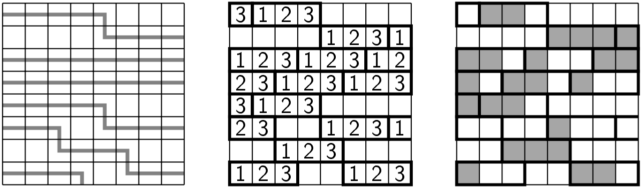
Combinatorial methods

Definition subshift Δ_r; ex for $r = 3$:



Combinatorial methods

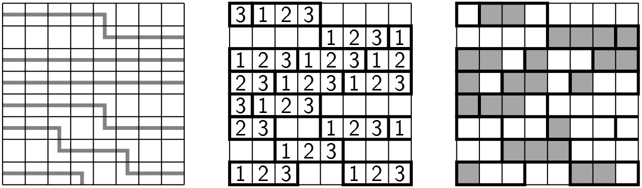
Definition subshift Δ_r; ex for $r = 3$:



Theorem:[G.,Sablik] $h(\Delta_r) = \frac{\log_2(r+1)}{r}$.

Combinatorial methods

Definition subshift Δ_r; ex for $r = 3$:



Theorem:[G.,Sablik] $h(\Delta_r) = \frac{\log_2(r+1)}{r}$.

Question: for what kind of subshifts can we compute entropy with similar methods ?

Proof for $r = 1$

Proof for $r = 1$

Second layer is trivial: we consider only first and third.

Proof for $r = 1$

Second layer is trivial: we consider only first and third.

Lower bound:

Proof for $r = 1$

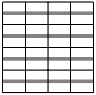
Second layer is trivial: we consider only first and third.

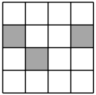
Lower bound: $\mathcal{L}_n(\Delta_1)$ contains the following patterns:

Proof for $r = 1$

Second layer is trivial: we consider only first and third.

Lower bound: $\mathcal{L}_n(\Delta_1)$ contains the following patterns:

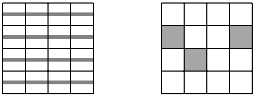




Proof for $r = 1$

Second layer is trivial: we consider only first and third.

Lower bound: $\mathcal{L}_n(\Delta_1)$ contains the following patterns:



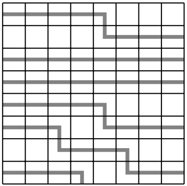
$$N_n(\Delta_1) \geq 2^{n^2}.$$

Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:

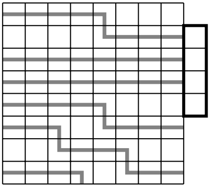
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



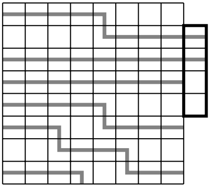
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



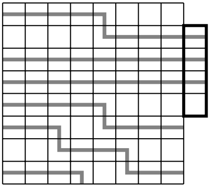
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



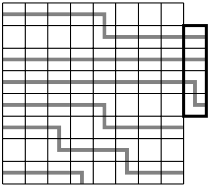
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



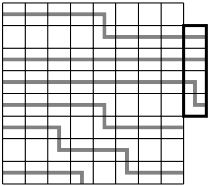
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



Proof for $r = 1$

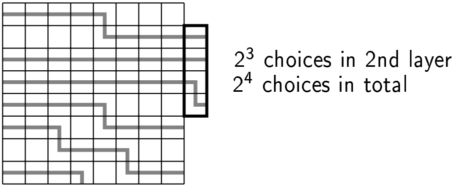
Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



2^3 choices in 2nd layer

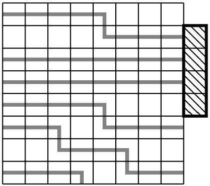
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



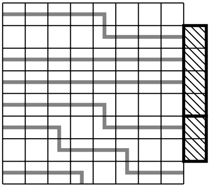
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



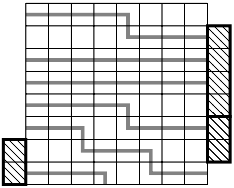
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



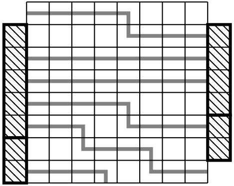
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



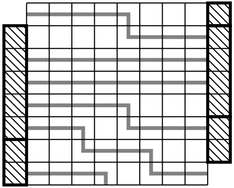
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



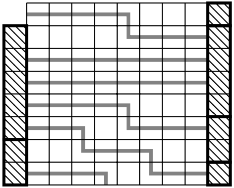
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



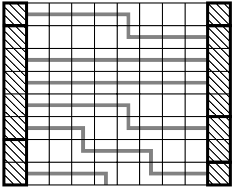
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



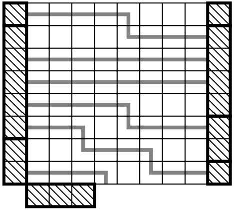
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



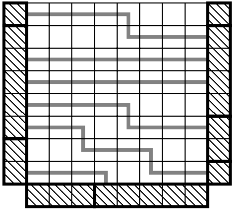
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



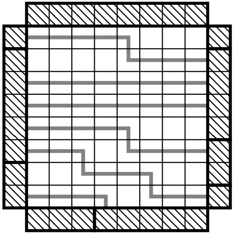
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



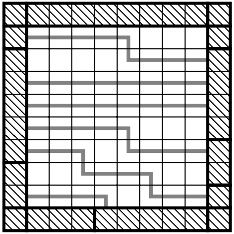
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



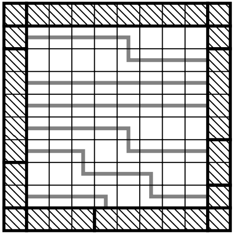
Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



Proof for $r = 1$

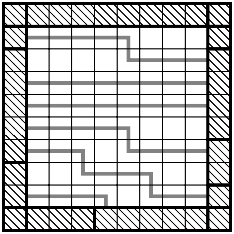
Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



Number of possible extensions into a pattern of $\mathcal{L}_{n+1}(\Delta_1)$:
$\leq 2^{4n} 2^{3*3*4} = 2^{4n+4} 2^{C}$.

Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:

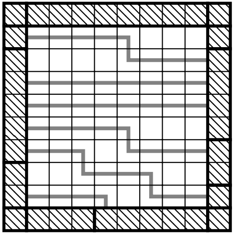


Number of possible extensions into a pattern of $\mathcal{L}_{n+1}(\Delta_1)$:
$\leq 2^{4n} 2^{3*3*4} = 2^{4n+4} 2^{C}$.

$$2^{n^2} \leq N_n(\Delta_1) \leq 2^{Cn} \cdot 2^{n^2}.$$

Proof for $r = 1$

Upper bound: Consider a pattern in $\mathcal{L}_n(\Delta_1)$ for some n:



Number of possible extensions into a pattern of $\mathcal{L}_{n+1}(\Delta_1)$:
$\leq 2^{4n} 2^{3*3*4} = 2^{4n+4} 2^{C}$.

$$2^{n^2} \leq N_n(\Delta_1) \leq 2^{Cn} \cdot 2^{n^2}.$$

$h(\Delta_1) = 1 = \frac{\log_2(2)}{1}$.

Combinatorial method for square ice entropy ?

Some ideas:

Combinatorial method for square ice entropy ?

Some ideas:

1. For instance: use Hochman-Meyerovitch's theorem to realize square ice entropy; then simplify the construction without changing the entropy; find a subshift which is isomorphic to square ice.

Combinatorial method for square ice entropy ?

Some ideas:

1. For instance: use Hochman-Meyerovitch's theorem to realize square ice entropy; then simplify the construction without changing the entropy; find a subshift which is isomorphic to square ice.
2. Find smaller and smaller subshifts of square ice with same entropy.

Combinatorial method for square ice entropy ?

Some ideas:

1. For instance: use Hochman-Meyerovitch's theorem to realize square ice entropy; then simplify the construction without changing the entropy; find a subshift which is isomorphic to square ice.
2. Find smaller and smaller subshifts of square ice with same entropy.
3. Try Baxter's method for square ice; more precisely: do both transfer matrices of square ice have same maximal eigenvalues or not ?