Networks of Relations in the Service of Constrained Coding

Moshe Schwartz

Electrical and Computer Engineering
Ben-Gurion University of the Negev, Israel
schwartz@ee.bgu.ac.il

Based on a joint work with Jehoshua Bruck

Outline

The Origin of Coding Constraints

Observation

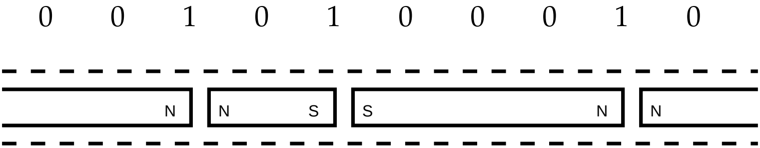
Hardware constraints translate into coding constraints.

The Origin of Coding Constraints

Observation

Hardware constraints translate into coding constraints.

Example of magnetic recording:



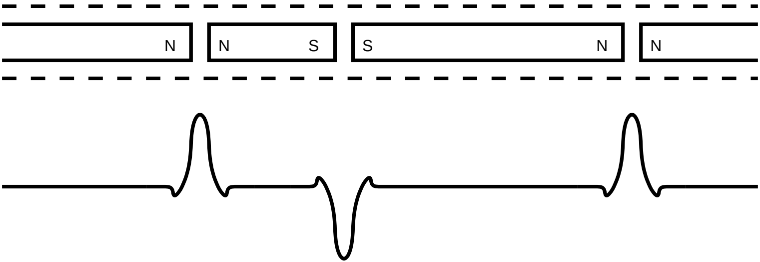
The Origin of Coding Constraints

Observation

Hardware constraints translate into coding constraints.

Example of magnetic recording:

$0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0$

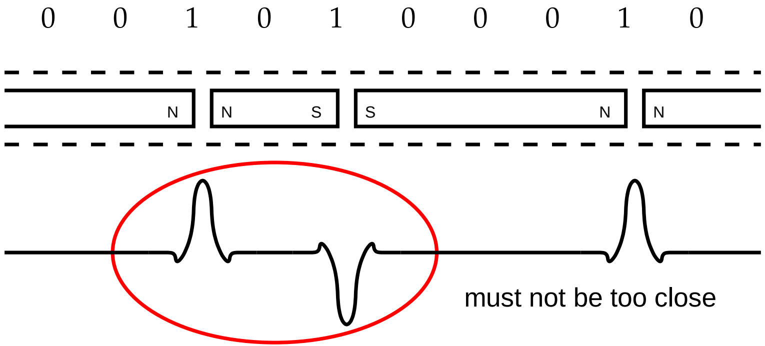


The Origin of Coding Constraints

Observation

Hardware constraints translate into coding constraints.

Example of magnetic recording:



Constrained Systems

Definition

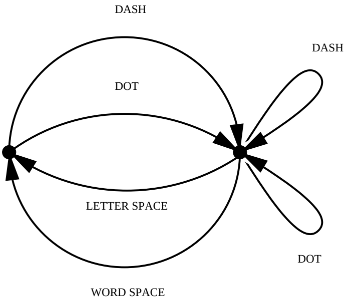
A one-dimensional constrained system S is a set of finite words (over a finite alphabet) which obey a certain constraint.

Observation

Most useful one-dimensional constraints are regular languages.

Goal

We want to losslessly translate arbitrary sequences of input bits to constrained sequences.

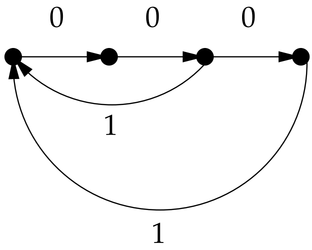


Telegraph channel constraint, **C. E. Shannon**, 1948.

Constrained Systems – Examples

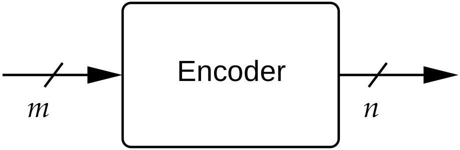
Example

The (d,k)-RLL (Run-Length Limited) constrained system is the set of all $\{0,1\}$-sequences such that the number of 0's between adjacent 1's is at least d, and there are no $k+1$ consecutive zeroes.



$(2,3)$-RLL Graph

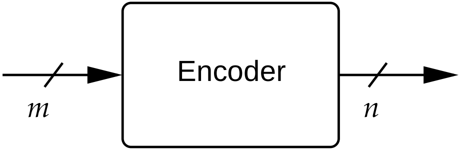
Constrained Systems – Encoders



Definition

A rate $R = m/n$ encoder for a constrained system S is a mapping $\{0,1\}^m \to \{0,1\}^n$ such that the concatenated output is a sequence of S.

Constrained Systems – Encoders



Definition

A rate $R = m/n$ encoder for a constrained system S is a mapping $\{0,1\}^m \to \{0,1\}^n$ such that the concatenated output is a sequence of S.

Question

How high can the code rate be?

The Capacity of Constrained Systems

Definition

The capacity of the constrained system S is

$$\mathrm{cap}(S) \stackrel{\text{def}}{=} \lim_{n\to\infty} \frac{\log_2 |S_n|}{n},$$

where $|S_n|$ is the number of sequences in S of length n.

Theorem (**Shannon**, 1948)

If there exists a decodable code at rate $R = m/n$ for S, then $R \leqslant \mathrm{cap}(S)$.

Theorem (**Shannon**, 1948)

For any rate $R = m/n < \mathrm{cap}(S)$ there exists a block code for S with rate R.

More on the Origin of (d,k)-RLL

Question

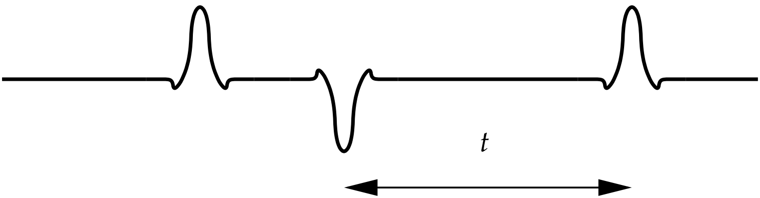
Where does the k in (d,k)-RLL come from?

More on the Origin of (d,k)-RLL

Question

Where does the k in (d,k)-RLL come from?

Example of magnetic recording:

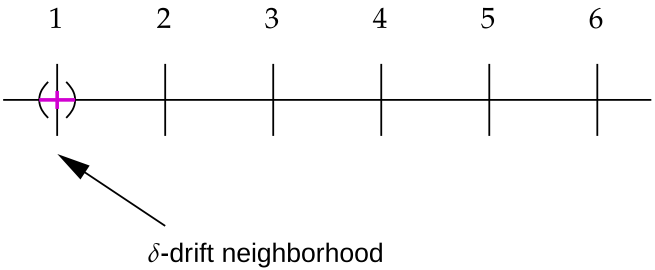


The writer intends to write duration t, but because of a clock drift, the reader may obtain $(1-\delta)t < t' < (1+\delta)t$. Thus, long runs may result in spurious or missing zeros after decoding.

More on the Origin of (d,k)-RLL

Question

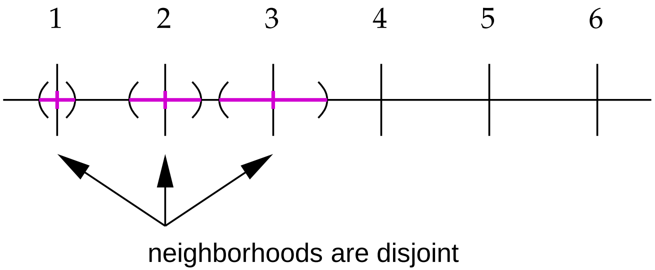
Where does the k in (d,k)-RLL come from?



More on the Origin of (d,k)-RLL

Question

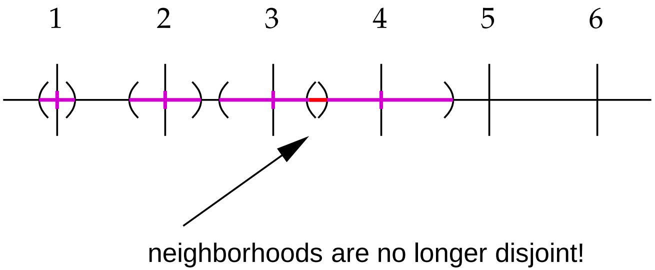
Where does the k in (d,k)-RLL come from?



More on the Origin of (d,k)-RLL

Question

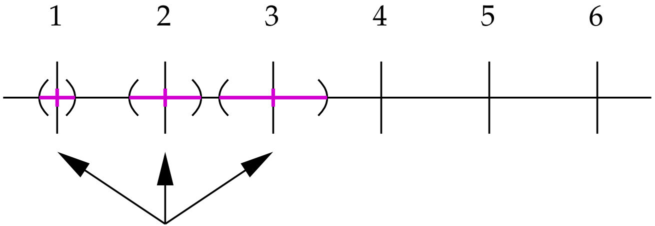
Where does the k in (d,k)-RLL come from?



More on the Origin of (d, k)-RLL

Question

Where does the k in (d, k)-RLL come from?



This results in $(0, 2)$-RLL.

Two-Dimensional Constrained Systems

Definition

A two-dimensional constrained system S is a set of $n \times m$ arrays (over a finite alphabet) obeying some constraint.

Two-Dimensional Constrained Systems

Definition

A two-dimensional constrained system S is a set of $n \times m$ arrays (over a finite alphabet) obeying some constraint.

Example

The (d,k)-RLL system is the set of all $\{0,1\}$-arrays such that in each column and row, the number of 0's between adjacent 1's is at least d, and there are no $k+1$ consecutive zeroes.

Two-Dimensional Constrained Systems

Definition

A two-dimensional constrained system S is a set of $n \times m$ arrays (over a finite alphabet) obeying some constraint.

Example

The (d, k)-RLL system is the set of all $\{0, 1\}$-arrays such that in each column and row, the number of 0's between adjacent 1's is at least d, and there are no $k + 1$ consecutive zeroes.

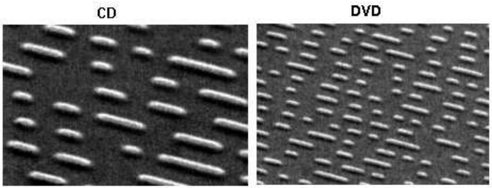
Example

The no isolated bit system is the set of all $\{0, 1\}$-arrays such that they contain no 0 surrounded by 1's and no 1 surrounded by 0's.

Two-Dimensional Constrained Systems (Cont.)

Motivation

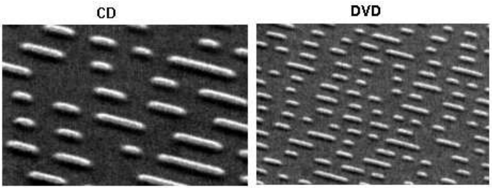
Some two-dimensional applications pose constraints, e.g., magnetic drives, optical and holographic storage devices.



Two-Dimensional Constrained Systems (Cont.)

Motivation

Some two-dimensional applications pose constraints, e.g., magnetic drives, optical and holographic storage devices.



Definition

The capacity of the constrained system S is

$$\mathrm{cap}(S) \stackrel{\text{def}}{=} \lim_{n,m\to\infty} \frac{\log_2 |S_{n,m}|}{nm},$$

where $|S_{n,m}|$ is the number of arrays in S of size $n \times m$.

Prior Work

- **N. Calkin and H. Wilf**, *SIAM J. Disc. Math.*, 1998

  Used the transfer-matrix method to provide numerical bounds on $\mathsf{cap}(S^{1,\infty})$:

  $$0.5878911617 \leqslant \mathsf{cap}(S^{1,\infty}) \leqslant 0.5878911618$$

  ($S^{1,\infty}$ is the set of all $(1,\infty)$-RLL arrays, i.e., binary arrays which do not have adjacent 1's. Equivalently, it is the set of all independent sets in the grid graph.)

- **A. Kato and K. Zeger**, *IEEE Trans. Inform. Theory*, 1999

  Found the zero-capacity regions of two-dimensional (d,k)-RLL constraints: $\mathsf{cap}(S^{d,d+1}) = 0$ for all $d > 0$, and $\mathsf{cap}(S^{d,k}) > 0$ for $k \geqslant d+2$. They also provided weak bounds on the capacity when it is not zero.

Prior Work

- **S. Halevy, *et al.***, *IEEE Trans. Information Theory*, 2004

  Used a constructive approach in which variable-rate bit-stuffing encoders are analyzed to provide the best yet known lower bounds on $\mathsf{cap}(S^{d,\infty})$ for $d > 1$.
- **M. Schwartz and A. Vardy**, *Proc. AAECC-16*, 2006

  Proved asymptotically-tight (as $k \to \infty$) lower and upper bounds on $\mathsf{cap}(S^{0,k})$ by using probabilistic tools.
- **S. Forchhammer and T. V. Laursen**, *Proc. ISIT06*, 2006

  Used random fields to approximate the capacity of the two-dimensional no-isolated-bit constraint.

Prior Work

- **R. J. Baxter**, *J. Physics*, 1980

  Gave an **exact** but non-rigorous solution to the capacity of **hexagonal** $(0,1)$-RLL.

$$\mathsf{cap}(S^{1,\infty}_{\mathrm{hex}}) = \log_2 \kappa_h \approx 0.480767622 \qquad \text{where} \qquad \kappa_h = \kappa_1\kappa_2\kappa_3\kappa_4$$

$$\kappa_1 = 4^{-1}3^{5/4}11^{-5/12}c^{-2} \qquad a = -\frac{124}{363}\sqrt[3]{11}$$

$$\kappa_2 = \left(1 - \sqrt{1-c} + \sqrt{2+c+2\sqrt{1+c+c^2}}\right)^2 \qquad b = \frac{2501}{11979}\sqrt{33}$$

$$\kappa_3 = \left(-1 - \sqrt{1-c} + \sqrt{2+c+2\sqrt{1+c+c^2}}\right)^2 \qquad c = \sqrt[3]{\frac{1}{4} + \frac{3}{8}a\left(\sqrt[3]{b+1} - \sqrt[3]{b-1}\right)}$$

$$\kappa_4 = \left(\sqrt{1-a} + \sqrt{2+a+2\sqrt{1+a+a^2}}\right)^{-1/2}$$

As Baxter notes: *"It is not mathematically rigorous, in that certain analyticity properties . . . are assumed, and the results . . . (which depend on assuming that various large-lattice limits can be interchanged) are used. However, I believe that these assumptions . . . are in fact correct."*

The Path-Cover Constraint

Definition

Given an undirected graph G, the Path-Cover Constraint is the set of all subsets of edges such that every vertex in the induced graph has degree either 1 or 2, i.e., a set of non-intersecting paths cover all the vertices of the graph.

The Path-Cover Constraint

Definition

Given an undirected graph G, the Path-Cover Constraint is the set of all subsets of edges such that every vertex in the induced graph has degree either 1 or 2, i.e., a set of non-intersecting paths cover all the vertices of the graph.

Observation

Equivalently, an assignment of 0's and 1's to edges such that every vertex “sees” exactly 1 or 2 incident edges assigned a 1.

The Path-Cover Constraint

Definition

Given an undirected graph G, the Path-Cover Constraint is the set of all subsets of edges such that every vertex in the induced graph has degree either 1 or 2, i.e., a set of non-intersecting paths cover all the vertices of the graph.

Observation

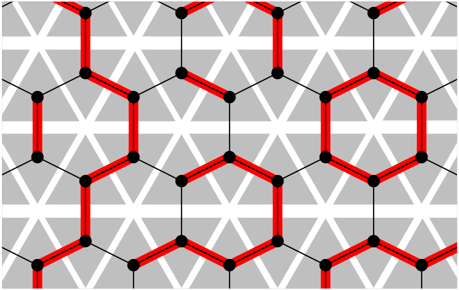
Equivalently, an assignment of 0's and 1's to edges such that every vertex "sees" exactly 1 or 2 incident edges assigned a 1.

Observation

If the graph G is a "one-dimensional" string graph, then the PC (Path Cover) constraint is exactly the $(0, 1)$-RLL constraint.

The PC Constraint on the Triangular Grid

We will examine the PC constraint over the two-dimensional triangular grid. An example of a PC constrained array:



Networks of Relations – Definitions

Definition

A network of relations is an undirected graph for which we associate with each vertex v a relation over $\deg(v)$ variables.

Definition

A satisfying assignment is an assignment of values to the edges, such that each vertex-relation is satisfied.

Networks of Relations – Definitions

Definition

A network of relations is an undirected graph for which we associate with each vertex v a relation over $\deg(v)$ variables.

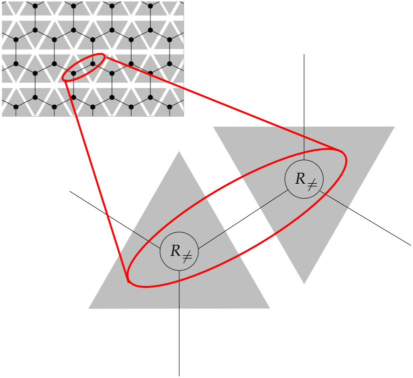
Definition

A satisfying assignment is an assignment of values to the edges, such that each vertex-relation is satisfied.

Motivation

We want to build a network of relations such that its satisfying assignments correspond to valid constrained arrays.

Step #1: Constraint $\rightarrow$ Network of Relations



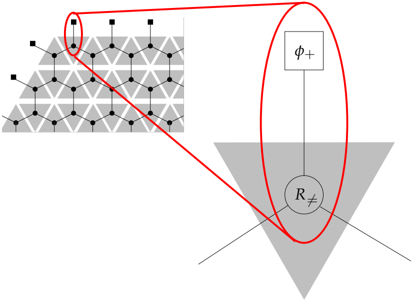
Definition

The relation $R_{\neq}$ is satisfied by all assignments *except* for the $(0,0,0)$ and $(1,1,1)$ assignments.

Observation

The satisfying assignments to the edges are exactly the valid PC-constrained arrays.

Step #1: Constraint $\rightarrow$ Network of Relations



Definition

The relation ϕ_+, the “accept all” relation, is satisfied by all assignments. It is used at the edges of the $n \times m$ array.

Observation

The satisfying assignments to the edges are exactly the valid PC-constrained arrays.

Holographic Reductions

Method

Under certain conditions a network of relations may be transformed into a weighted graph by replacing each relation vertex with a corresponding fixed gadget.

— **L. G. Valiant**, *FOCS 2004*.

Holographic Reductions

Method

Under certain conditions a network of relations may be transformed into a weighted graph by replacing each relation vertex with a corresponding fixed gadget.

— **L. G. Valiant**, *FOCS 2004*.

Motivation

The number of satisfying assignments of the original network of relations equals the weighted perfect matching of the resulting weighted graph.

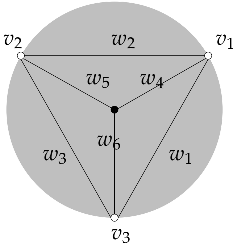
The Perfect Matching

Definition

Let $G = (V, E)$ be a graph. A perfect matching is a subset of edges $M \subseteq E$ such that every vertex $v \in V$ is incident to exactly one of the edges in M. The set of all perfect matchings will be denoted $\mathrm{PM}(G)$. We can now assign complex weights to the edges $w : E \to \mathbb{C}$, and define the weighted perfect matching of G to be

$$\mathrm{PerfMatch}(G) \stackrel{\mathrm{def}}{=} \sum_{M \in \mathrm{PM}(G)} \prod_{e \in M} w(e).$$

Matchgates and Matchgrids



Definition

A matchgate is a graph $G = (V, E, X, Y)$ with a set of input nodes $X \subseteq V$, and a set of output nodes $Y \subseteq V$, where X and Y are disjoint and $|X| + |Y|$ equals the number of variables in the original relation.

Definition

A matchgrid is a network of relations whose vertices have been replaced by appropriate matchgates, and every input vertex is incident to exactly one output vertex (and vice versa) by an edge from the original network.

Signatures of Relations

Definition

The signature of a relation over n variables is the length 2^n binary vector, indexed by all possible variable assignments, in which 0 stands for "not-satisfied" and 1 stands for "satisfied".

Example

The signatures for $R_{\neq}$ and ϕ_+ are:

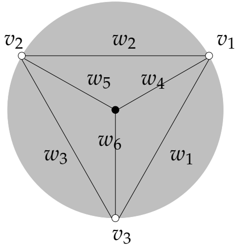
x_1	x_2	x_3	$R_{\neq}$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

x_1	ϕ_+
0	1
1	1

Signatures of Matchgates

Definition

The interaction of the matchgate with the outside world is given by a $2^{|X|} \times 2^{|Y|}$ matrix, called the signature of the matchgate: for every $Z \subseteq X \cup Y$ there is an entry containing $\mathrm{PerfMatch}(G - Z)$.



$$
\begin{array}{cc}
\text{index} & \text{signature} \\
\begin{array}{c} (0,0,0) \\ (0,0,1) \\ (0,1,0) \\ (0,1,1) \\ (1,0,0) \\ (1,0,1) \\ (1,1,0) \\ (1,1,1) \end{array} &
\begin{pmatrix} w_1 w_5 + w_2 w_6 + w_3 w_4 \\ 0 \\ 0 \\ w_4 \\ 0 \\ w_5 \\ w_6 \\ 0 \end{pmatrix}
\end{array}
$$

Signatures of Matchgates

Observation

The signature of a generator (a matchgate with only output nodes) is a column vector, while the signature of a recognizer (a matchgate with only input nodes) is a row vector.

Signatures of Matchgates

Observation

The signature of a generator (a matchgate with only output nodes) is a column vector, while the signature of a recognizer (a matchgate with only input nodes) is a row vector.

Observation

Half the entries of the signature of a matchgate are guaranteed to be zero (depending on the parity of the index and the parity of the vertex set).

Signatures of Matchgates

Observation

The signature of a generator (a matchgate with only output nodes) is a column vector, while the signature of a recognizer (a matchgate with only input nodes) is a row vector.

Observation

Half the entries of the signature of a matchgate are guaranteed to be zero (depending on the parity of the index and the parity of the vertex set).

Problem

Will we be able to build matchgates for $R_{\neq}$ and ϕ_{+}?

Change of Bases

Definition

A basis is an ordered set of vectors. The standard basis is defined as $\mathbf{b} = [(1,0),(0,1)]$. Let $\boldsymbol{\beta} = [\mathsf{n},\mathsf{p}] = [(\mathsf{n}_0,\mathsf{n}_1),(\mathsf{p}_0,\mathsf{p}_1)]$ be some basis. We define the basis translation matrix as

$$T_{\beta} \stackrel{\text{def}}{=} \begin{pmatrix} \mathsf{n}_0 & \mathsf{n}_1 \\ \mathsf{p}_0 & \mathsf{p}_1 \end{pmatrix}.$$

Let Γ be some matchgate with n input/output vertices.

$$\text{sig}_{\beta}(\Gamma) \cdot T_{\beta}^{\otimes n} = \text{sig}_{\mathbf{b}}(\Gamma) \qquad \text{for } \Gamma \text{ a generator} \qquad (1)$$

$$T_{\beta}^{\otimes n} \cdot \text{sig}_{\mathbf{b}}(\Gamma) = \text{sig}_{\beta}(\Gamma) \qquad \text{for } \Gamma \text{ a recognizer} \qquad (2)$$

Change of Bases (Cont.)

Goal

Find a basis such that all matchgates are realizable.

Change of Bases (Cont.)

Goal

Find a basis such that all matchgates are realizable.

Example

We choose the basis $\beta = [\mathsf{n}, \mathsf{p}] = [(1,1),(1,-1)]$. Indeed:

$$(0,1,1,1,1,1,1,0) \cdot T_{\beta}^{\otimes 3} = (6,0,0,-2,0,-2,-2,0),$$

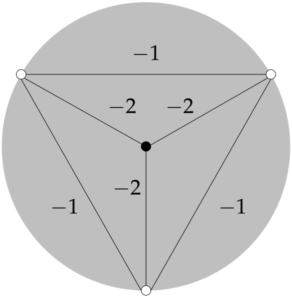
and the generator matchgate is realizable since

$$(w_1 w_5 + w_2 w_6 + w_3 w_4, 0, 0, w_4, 0, w_5, w_6, 0) = \\ = (6,0,0,-2,0,-2,-2,0).$$

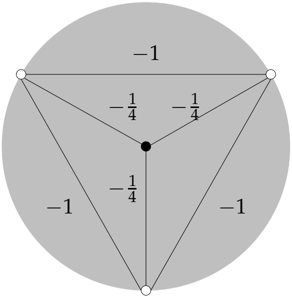
Change of Bases (Cont.)

Goal

Find a basis such that all matchgates are realizable.

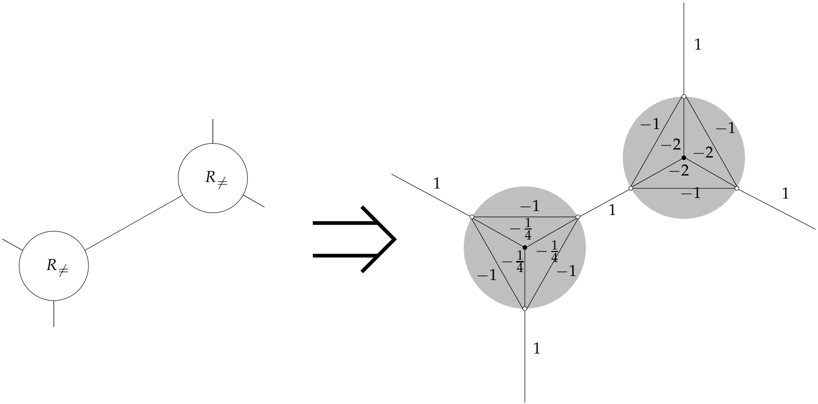


Generator for $R_{\neq}$



Recognizer for $R_{\neq}$

Step #2: Network of Relations → Weighted Graph



The Holant Theorem

Definition

Given some $x \in \{\mathsf{n}, \mathsf{p}\}^{\otimes n}$, we associate with it an index vector by substituting 0 for n and 1 for p. For example, with $\mathsf{n} \otimes \mathsf{p} \otimes \mathsf{n}$ we associate the index vector $(0, 1, 0)$. For Γ a generator (recognizer), we define $\mathrm{valG}_{\beta}(\Gamma, x)$ (we define $\mathrm{valR}_{\beta}(\Gamma, x)$) to be the entry in $\mathrm{sig}_{\beta}(\Gamma)$ with the index associated with x.

The Holant Theorem

Definition

Given some $x \in \{\mathsf{n}, \mathsf{p}\}^{\otimes n}$, we associate with it an index vector by substituting 0 for n and 1 for p. For example, with $\mathsf{n} \otimes \mathsf{p} \otimes \mathsf{n}$ we associate the index vector $(0, 1, 0)$. For Γ a generator (recognizer), we define $\mathrm{valG}_{\beta}(\Gamma, x)$ (we define $\mathrm{valR}_{\beta}(\Gamma, x)$) to be the entry in $\mathrm{sig}_{\beta}(\Gamma)$ with the index associated with x.

Definition

For a matchgrid $\mathcal{M}$ with f edges between matchgates,

$$\mathrm{Holant}(\mathcal{M}) = \sum_{x \in \beta^{\otimes f}} \left(\prod_{1 \leqslant j \leqslant g} \mathrm{valG}_{\beta}(B_j, x) \right) \left(\prod_{1 \leqslant i \leqslant r} \mathrm{valR}_{\beta}(A_i, x) \right)$$

The Holant Theorem

Definition

For a matchgrid $\mathcal{M}$ with f edges between matchgates,

$$\mathrm{Holant}(\mathcal{M}) = \sum_{x \in \beta^{\otimes f}} \left(\prod_{1 \leqslant j \leqslant g} \mathrm{valG}_{\beta}(B_j, x) \right) \left(\prod_{1 \leqslant i \leqslant r} \mathrm{valR}_{\beta}(A_i, x) \right)$$

Observation

Under the standard basis, $\mathrm{Holant}(\mathcal{M})$ is $\mathrm{PerfMatch}(G)$. Under our chosen basis β, $\mathrm{Holant}(\mathcal{M})$ is the number of satisfying assignments to the network of relations since valG and valR query the signatures of the relations.

The Holant Theorem

Observation

Under the standard basis, $\mathrm{Holant}(\mathcal{M})$ is $\mathrm{PerfMatch}(G)$. Under our chosen basis β, $\mathrm{Holant}(\mathcal{M})$ is the number of satisfying assignments to the network of relations since valG and valR query the signatures of the relations.

Theorem

For any matchgrid $\mathcal{M}$ over any basis β, if $\mathcal{M}$ has weighted graph G then

$$\mathrm{Holant}(\mathcal{M}) = \mathrm{PerfMatch}(G).$$

Calculating the Perfect Matching

Problem

How do we calculate $\mathrm{PerfMatch}(G)$?

Calculating the Perfect Matching

Definition

A canonical partition, π, of $\{1, \dots, n\}$ is a list of pairs $\mid p_1 p_2 \mid p_3 p_4 \mid \dots \mid p_{n-1} p_n \mid$ such that $p_1 < p_2$, $p_3 < p_4$, up until $p_{n-1} < p_n$, and $p_1 < p_3 < \cdots < p_{n-1}$.

Calculating the Perfect Matching

Definition

A canonical partition, π, of $\{1,\dots,n\}$ is a list of pairs $\mid p_1 p_2 \mid p_3 p_4 \mid \dots \mid p_{n-1} p_n \mid$ such that $p_1 < p_2$, $p_3 < p_4$, up until $p_{n-1} < p_n$, and $p_1 < p_3 < \dots < p_{n-1}$.

Observation

With a_{p_i,p_j} being the weight of the edge $p_i \to p_j$,

$$\mathrm{PerfMatch}(G) = \sum_{\pi} a_{p_1,p_2} a_{p_3,p_4} \dots a_{p_{n-1},p_n}.$$

Calculating the Perfect Matching

Definition

A canonical partition, π, of $\{1,\ldots,n\}$ is a list of pairs $\mid p_1 p_2 \mid p_3 p_4 \mid \ldots \mid p_{n-1} p_n \mid$ such that $p_1 < p_2$, $p_3 < p_4$, up until $p_{n-1} < p_n$, and $p_1 < p_3 < \cdots < p_{n-1}$.

Observation

With a_{p_i,p_j} being the weight of the edge $p_i \to p_j$,

$$\mathrm{PerfMatch}(G) = \sum_{\pi} a_{p_1,p_2} a_{p_3,p_4} \ldots a_{p_{n-1},p_n}.$$

Does this look familiar?

$$\sum_{\pi} \mathrm{sgn}(\pi) a_{p_1,p_2} a_{p_3,p_4} \ldots a_{p_{n-1},p_n}.$$

Some Algebra

Definition

Let $A = (a_{i,j})$ be the part above the main diagonal of an $n \times n$ matrix. Then the Pfaffian of A is defined as

$$\mathrm{Pf}(A) = \sum_{\pi} \mathrm{sgn}(\pi) a_{p_1,p_2} a_{p_3,p_4} \ldots a_{p_{n-1},p_n}.$$

Some Algebra

Definition

Let $A = (a_{i,j})$ be the part above the main diagonal of an $n \times n$ matrix. Then the Pfaffian of A is defined as

$$\mathrm{Pf}(A) = \sum_{\pi} \mathrm{sgn}(\pi) a_{p_1,p_2} a_{p_3,p_4} \dots a_{p_{n-1},p_n}.$$

Theorem

If we complete A to be an $n \times n$ anti-symmetric matrix then we get $(\mathrm{Pf}(A))^2 = \det(A)$.

Some Algebra

Definition

Let $A = (a_{i,j})$ be the part above the main diagonal of an $n \times n$ matrix. Then the Pfaffian of A is defined as

$$\mathrm{Pf}(A) = \sum_{\pi} \mathrm{sgn}(\pi) a_{p_1,p_2} a_{p_3,p_4} \dots a_{p_{n-1},p_n}.$$

Theorem

If we complete A to be an $n \times n$ anti-symmetric matrix then we get $(\mathrm{Pf}(A))^2 = \det(A)$.

Observation

Without $\mathrm{sgn}(\pi)$, the Pfaffian becomes the Hafnian, which is to the permanent as the Pfaffian is to the determinant.

The Fisher-Kasteleyn-Temperley Method

Problem

- The Hafnian and the permanent are notoriously hard to handle, but give the perfect matching exactly.
- The Pfaffian and determinant are easy to handle, but count some of the summands with the wrong sign.

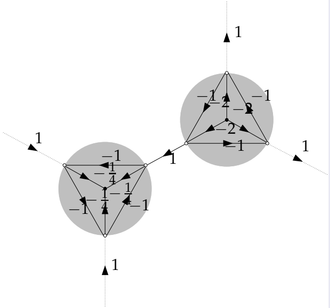
The Fisher-Kasteleyn-Temperley Method

Method

- The weighted perfect matching of a graph may be calculated (up to a sign) as the square root of the determinant of its anti-symmetric adjacency matrix.
- The signs of the entries in the matrix are determined by a Pfaffian orientation of the graph. Every planar graph has a Pfaffian orientation.

— **H. N. V. Temperley and M. E. Fisher**, *Phil. Mag.*, 1960.
— **P. W. Kasteleyn**, *Physica*, 1961.

Step #3: Pfaffian Orientation



Method

For a planar graph, an orientation of the edges such that every clockwise walk on a face has an *odd* number of edges agreeing, is a Pfaffian orientation. Set

$$a_{i,j} = \begin{cases} 0 & \text{no edge} \\ w(e_{i,j}) & \text{if } i \to j \\ -w(e_{i,j}) & \text{if } j \to i \end{cases}$$

and then $\mathrm{Pf}(A) = \mathrm{PerfMatch}(G)$.

— **P. W. Kasteleyn**, *Physica*, 1961.

An Exact Solution – The Determinant

An $n \times n$ array of basic blocks has the following anti-symmetric adjacency matrix:

$$
\begin{aligned}
A = I_n \otimes I_n \otimes B + I_n \otimes U_n \otimes \Delta_{6,2} - I_n \otimes U_n^T \otimes \Delta_{6,2}^T \\
+ U_n \otimes I_n \otimes \Delta_{7,3} - U_n^T \otimes I_n \otimes \Delta_{7,3}^T.
\end{aligned}
$$

An Exact Solution – The Determinant

An $n \times n$ array of basic blocks has the following anti-symmetric adjacency matrix:

$$A = I_n \otimes I_n \otimes B + I_n \otimes U_n \otimes \Delta_{6,2} - I_n \otimes U_n^T \otimes \Delta_{6,2}^T + U_n \otimes I_n \otimes \Delta_{7,3} - U_n^T \otimes I_n \otimes \Delta_{7,3}^T.$$

Takes care of the basic block. I_n is the $n \times n$ identity matrix and

$$B = \begin{pmatrix} 0 & -1 & 1 & -\frac{1}{4} & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & -\frac{1}{4} & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & -\frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 & 1 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 2 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 & -2 & -2 & -2 & 0 \end{pmatrix}.$$

An Exact Solution – The Determinant

An $n \times n$ array of basic blocks has the following anti-symmetric adjacency matrix:

$$A = I_n \otimes I_n \otimes B + I_n \otimes U_n \otimes \Delta_{6,2} - I_n \otimes U_n^T \otimes \Delta_{6,2}^T$$
$$+ U_n \otimes I_n \otimes \Delta_{7,3} - U_n^T \otimes I_n \otimes \Delta_{7,3}^T.$$

Takes care of edges between blocks in the same row.
$\Delta_{i,j}$ (of the same dimensions as B) which is all zeroes except for position (i, j) which is 1. Also

$$U_n \stackrel{\text{def}}{=} \begin{pmatrix} 0 & 1 & & & & \\ & 0 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & & 0 & 1 \\ & & & & & 0 \end{pmatrix}$$

An Exact Solution – The Determinant

An $n \times n$ array of basic blocks has the following anti-symmetric adjacency matrix:

$$\begin{aligned} A = I_n \otimes I_n \otimes B + I_n \otimes U_n \otimes \Delta_{6,2} - I_n \otimes U_n^T \otimes \Delta_{6,2}^T \\ + U_n \otimes I_n \otimes \Delta_{7,3} - U_n^T \otimes I_n \otimes \Delta_{7,3}^T. \end{aligned}$$

Takes care of edges between blocks in the same column.
$\Delta_{i,j}$ (of the same dimensions as B) which is all zeroes except for position (i, j) which is 1. Also

$$U_n \stackrel{\text{def}}{=} \begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & \ddots & \ddots & \\ & & & 0 & 1 \\ & & & & 0 \end{pmatrix}$$

An Exact Solution – The Determinant

An $n \times n$ array of basic blocks has the following anti-symmetric adjacency matrix:

$$\begin{aligned} A = I_n \otimes I_n \otimes B + I_n \otimes U_n \otimes \Delta_{6,2} - I_n \otimes U_n^T \otimes \Delta_{6,2}^T \\ + U_n \otimes I_n \otimes \Delta_{7,3} - U_n^T \otimes I_n \otimes \Delta_{7,3}^T. \end{aligned}$$

Since each basic block stores 3 bit positions (edges), the capacity is

$$\mathrm{cap} = \lim_{n\to\infty} \frac{\log_2 \sqrt{\det(A)}}{3n^2}.$$

An Exact Solution – The Determinant

An $n \times n$ array of basic blocks has the following anti-symmetric adjacency matrix:

$$\begin{aligned} A = I_n \otimes I_n \otimes B + I_n \otimes U_n \otimes \Delta_{6,2} - I_n \otimes U_n^T \otimes \Delta_{6,2}^T \\ + U_n \otimes I_n \otimes \Delta_{7,3} - U_n^T \otimes I_n \otimes \Delta_{7,3}^T. \end{aligned}$$

Since each basic block stores 3 bit positions (edges), the capacity is

$$\mathrm{cap} = \lim_{n \to \infty} \frac{\log_2 \sqrt{\det(A)}}{3n^2}.$$

Observation

The matrix A is a 2-level Toeplitz matrix.

Spectral Distribution of Toeplitz Matrices

Definition

Let us denote $Q \stackrel{\text{def}}{=} [-\pi, \pi]$. For natural numbers $p, k \in \mathbb{N}$, let an integrable p-variate function $f : Q^p \to \mathbb{C}^{k\times k}$ and a multi-index $n = (n_1, \ldots, n_p)$, $n_i \geqslant 1$ be given. The p-level Toeplitz matrix $T_n(f)$ is defined as

$$T_n(f) \stackrel{\text{def}}{=} \sum_{j_1=-n_1+1}^{n_1-1} \cdots \sum_{j_p=-n_p+1}^{n_p-1} J_{n_1}^{(j_1)} \otimes \cdots \otimes J_{n_p}^{(j_p)} \otimes a_{j_1,\ldots,j_p}(f)$$

where $J_m^{(l)}$ denotes the matrix of order m whose i, j entry equals 1 if $j - i = l$ and equals zero otherwise, and where

$$a_{j_1,\ldots,j_p}(f) \stackrel{\text{def}}{=} \frac{1}{(2\pi)^p} \int_{Q^p} f(\phi) e^{-\mathbf{i}(j_1\phi_1+\cdots+j_p\phi_p)} d\phi$$

is a matrix in $\mathbb{C}^{k\times k}$ and $\mathbf{i} = \sqrt{-1}$.

Spectral Distribution of Toeplitz Matrices

Theorem (Tilli, 98)

If $f : Q^p \to \mathbb{C}^{k\times k}$ is an integrable Hermitian matrix-valued function, then for any function F, uniformly continuous and bounded over $\mathbb{R}$ it holds

$$\lim_{n\to\infty} \frac{1}{n_1 \dots n_p} \sum_{j=1}^{kn_1 \dots n_p} F[\lambda_j(T_n(f))] =$$

$$= \frac{1}{(2\pi)^p} \int_{Q^p} \sum_{j=1}^{k} F[\lambda_j(f(\phi))] d\phi$$

where $\lambda_j(M)$ denotes the j-th eigenvalue of M.

Spectral Distribution of Toeplitz Matrices

Theorem (Tilli, 98)

If $f : Q^p \to \mathbb{C}^{k\times k}$ is an integrable Hermitian matrix-valued function, then for any function F, uniformly continuous and bounded over $\mathbb{R}$ it holds

$$\lim_{n\to\infty} \frac{1}{n_1 \dots n_p} \sum_{j=1}^{kn_1 \dots n_p} F[\lambda_j(T_n(f))] = \\ = \frac{1}{(2\pi)^p} \int_{Q^p} \sum_{j=1}^{k} F[\lambda_j(f(\phi))] d\phi$$

where $\lambda_j(M)$ denotes the j-th eigenvalue of M.

An Exact Solution

Observation

- $\frac{1}{n}\log_2 \det(A) = \frac{1}{n}\sum \log_2 \lambda_i(A)$.
- $\mathbf{i}A = T_n(f)$ where we define

$$f(\phi_1,\phi_2) = \mathbf{i}[B + e^{\mathbf{i}\phi_1}\Delta_{6,2} - e^{-\mathbf{i}\phi_1}\Delta_{6,2}^T + e^{\mathbf{i}\phi_2}\Delta_{7,3} - e^{-\mathbf{i}\phi_2}\Delta_{7,3}^T].$$

An Exact Solution

Observation

- $\frac{1}{n}\log_2 \det(A) = \frac{1}{n}\sum \log_2 \lambda_i(A)$.
- $\mathbf{i}A = T_n(f)$ where we define

$$f(\phi_1,\phi_2) = \mathbf{i}[B + e^{\mathbf{i}\phi_1}\Delta_{6,2} - e^{-\mathbf{i}\phi_1}\Delta_{6,2}^T + e^{\mathbf{i}\phi_2}\Delta_{7,3} - e^{-\mathbf{i}\phi_2}\Delta_{7,3}^T].$$

The Solution

$$\begin{aligned}\text{cap} = \frac{1}{24\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \log_2 |21 - 4\cos\phi_1 - 4\cos\phi_2 \\ - 4\cos(\phi_1-\phi_2)|d\phi_1 d\phi_2 \\ = 0.72399217\ldots\end{aligned}$$

Result Summary

- A general approach to the problem of determining the capacity of two-dimensional constraints. We do not know the expressive power of the method.
- Generalization to non-planar graphs: we do not care about the exponential number of summands since we are interested in the capacity, but we find it difficult to find the dominant one.
- Extension to generalized relations: relations are no longer either satisfied or unsatisfied, but rather have a “degree” of satisfaction. For example, we can efficiently count $(0, 1)$-RLL with equal amount of horizontal and vertical violations, but again, we find it difficult to find the dominant summand.

More News

Since publication of this work, Louidor and Marcus (*IEEE Trans. IT*, 2010) used ad-hoc arguments to calculate the capacity of:

- 2-Charge-Constrained arrays: The alphabet is $\{+1, -1\}$, and the sum of every $1 \times \ell$ or $\ell \times 1$ window is between -2 and 2. The capacity is $\frac{1}{4}$.
- ODD-Constrained arrays: The alphabet is $\{0, 1\}$, and is an odd number of 0's between adjacent 1's in rows and columns. The capacity is $\frac{1}{2}$.

Some Interesting Open Problems...

What is the capacity of two-dimensional...

- (d,k)-RLL? (d,∞)-RLL? $(0,k)$-RLL? $(0,1)$-RLL (hard-square entropy constant)?
  Application: Magnetic and optimal storage devices
- c-Charge-Constraint?
  Application: Magnetic storage devices
- No-isolated-bit constraint? No-isolated 1's constraint?
  Application: Optical and phase-change memory devices
- No oriented cycle in the grid graph?
  Application: Flash memory devices

Thank You