Normal Factor Graphs, Linear Algebra and Probabilistic Modeling

Yongyi Mao
University of Ottawa

Workshop on
Counting, Inference, and Optimization on Graphs
Princeton University, NJ, USA

November 3, 2011

It started on day one of my PhD program ...

August 1, 2000 ...

- PhD advisor Frank Kschischang.

- "Codes on graphs: normal realizations" [Forney, 2001]
- Kschischang: "Go read it! "
- and went on a vacation ...

August 1, 2000 ...

- PhD advisor Frank Kschischang.

- "Codes on graphs: normal realizations" [Forney, 2001]
- Kschischang: "Go read it! "
- and went on a vacation ...

August 1, 2000 ...

- PhD advisor Frank Kschischang.

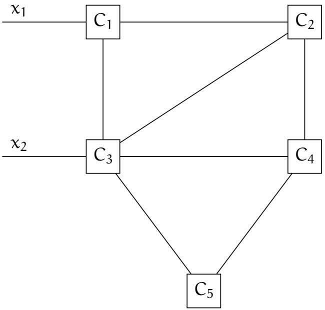
- "Codes on graphs: normal realizations" [Forney, 2001]
- Kschischang: "Go read it! "
- and went on a vacation ...

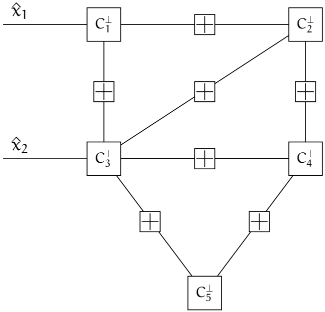
August 1, 2000 ...

- PhD advisor Frank Kschischang.

- "Codes on graphs: normal realizations" [Forney, 2001]
- Kschischang: "Go read it! "
- and went on a vacation ...

Normal graph duality





- Duality theorem: Dual normal graphs represent dual codes

One month later ...

- Kschischang:
  "Why do normal graphs have duality, but factor graphs do not?"
- Two weeks later ...
- Me: "We need ... Fourier ... we need convolution..."
- Given $f(x, y)$ and $g(y, z)$

$$f(x, y) * g(y, z) = \sum_{w} f(x, y - w) g(w, z)$$

One month later ...

- Kschischang:
  "Why do normal graphs have duality, but factor graphs do not?"
- Two weeks later ...
- Me: "We need ... Fourier ... we need convolution..."
- Given $f(x, y)$ and $g(y, z)$

$$f(x, y) * g(y, z) = \sum_{w} f(x, y - w) g(w, z)$$

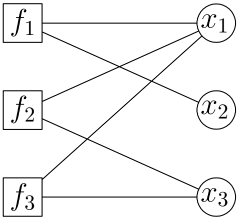
One month later ...

- Kschischang:
  "Why do normal graphs have duality, but factor graphs do not?"
- Two weeks later ...
- Me: "We need ... Fourier ... we need convolution..."
- Given $f(x, y)$ and $g(y, z)$

$$f(x, y) * g(y, z) = \sum_{w} f(x, y - w) g(w, z)$$

Convolutional factor graph

- bipartite graph: function nodes and variables nodes
- encodes a convolutional product



$$f_1(x_1, x_2) * f_2(x_1, x_3) * f_3(x_1, x_3)$$

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\hat{f}_i\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\widehat{f_i}\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\hat{f}_i\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\hat{f}_i\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\hat{f}_i\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\hat{f}_i\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

Factor graph duality

- Dual FGs: $(G, \{f_i\}, \times)$ and $(G, \{\hat{f_i}\}, *)$
  - $* \overset{\mathcal{F}}{\leftrightarrow} \times$
  - $\Rightarrow$ Dual FGs encode a FT pair
- Dual FGs of codes: $(G, \{\delta_{C_i}\}, \times)$ and $(G, \{\delta_{C_i^\perp}\}, *)$
  - $\delta_C \overset{\mathcal{F}}{\leftrightarrow} \delta_{C^\perp}$
  - $\Rightarrow$ Dual FGs represent dual codes
- Normal graph duality reinterpreted
- "On factor graphs and the Fourier transform" [Mao & Kschischang, ISIT2001, IT2005]

The reviewer ...

- Dave Forney

- 9 pages of review
- Forney: "It may take years for people to take it [convolutional FG] and run"

The reviewer ...

- Dave Forney

- 9 pages of review
- Forney: “It may take years for people to take it [convolutional FG] and run”

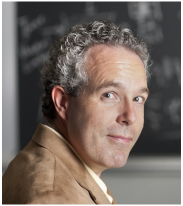
The reviewer ...

- Dave Forney

- 9 pages of review
- Forney: "It may take years for people to take it [convolutional FG] and run"

Going probabilistic ...

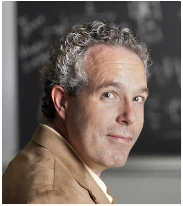
- Bredan Frey



- "Convolutional FGs as probabilistic models" [Mao, Kschischang & Frey, UAI 2004]
    - Each node: joint distribution of a collection of (latent) variables
    - All collections are independent
    - $\Rightarrow$ Convoltional CFG represents the joint distribution of observed variables that are linearly transformed from the latent variables

Going probabilistic ...

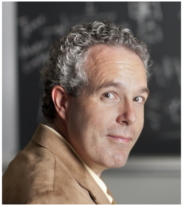
- Bredan Frey



- "Convolutional FGs as probabilistic models" [Mao, Kschischang & Frey, UAI 2004]
    - Each node: joint distribution of a collection of (latent) variables
    - All collections are independent
    - $\Rightarrow$ Convoltional CFG represents the joint distribution of observed variables that are linearly transformed from the latent variables

Going probabilistic ...

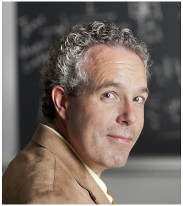
- Bredan Frey



- "Convolutional FGs as probabilistic models" [Mao, Kschischang & Frey, UAI 2004]
  - Each node: joint distribution of a collection of (latent) variables
  - All collections are independent
  - $\Rightarrow$ Convoltional CFG represents the joint distribution of observed variables that are linearly transformed from the latent variables

Going probabilistic ...

- Bredan Frey



- "Convolutional FGs as probabilistic models" [Mao, Kschischang & Frey, UAI 2004]
    - Each node: joint distribution of a collection of (latent) variables
    - All collections are independent
    - $\Rightarrow$ Convoltional CFG represents the joint distribution of observed variables that are linearly transformed from the latent variables

Linear characteristic graphical models

December 2010, email from Danny Bickson:

- "Inference with multivariate heavy-tails in linear models" [Bickson & Guestrin, NIPS2010]
  - modelling heavy-tail (stable) distributions
  - applications and inference algorithms
  - based on convolutional FG and duality

Linear characteristic graphical models

December 2010, email from Danny Bickson:

- "Inference with multivariate heavy-tails in linear models" [Bickson & Guestrin, NIPS2010]
  - modelling heavy-tail (stable) distributions
  - applications and inference algorithms
  - based on convolutional FG and duality

Linear characteristic graphical models

December 2010, email from Danny Bickson:

- "Inference with multivariate heavy-tails in linear models" [Bickson & Guestrin, NIPS2010]
  - modelling heavy-tail (stable) distributions
  - applications and inference algorithms
  - based on convolutional FG and duality

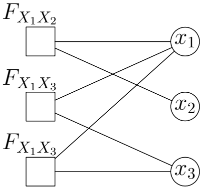
Linear characteristic graphical models

December 2010, email from Danny Bickson:

- "Inference with multivariate heavy-tails in linear models" [Bickson & Guestrin, NIPS2010]
    - modelling heavy-tail (stable) distributions
    - applications and inference algorithms
    - based on convolutional FG and duality

Missed opportunity ...

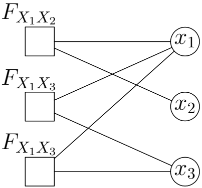
- Year 2005 ... met Frey in Toronto
- Frey: "How about a model for CDF?"
- "Cumulative distribution networks and derivative-sum-product algorithm" [Huang & Frey, UAI2008]



- CDN:
  - multiplicative FG
  - each function node: a CDF
  - $\Rightarrow$ the global function is a CDF
  - efficiently solves structured ranking problems

Missed opportunity ...

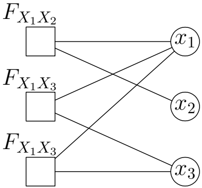
- Year 2005 ... met Frey in Toronto
- Frey: "How about a model for CDF?"
- "Cumulative distribution networks and derivative-sum-product algorithm" [Huang & Frey, UAI2008]



- CDN:
  - multiplicative FG
  - each function node: a CDF
  - $\Rightarrow$ the global function is a CDF
  - efficiently solves structured ranking problems

Missed opportunity ...

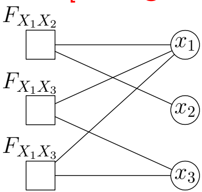
- Year 2005 ... met Frey in Toronto
- Frey: "How about a model for CDF?"
- "Cumulative distribution networks and derivative-sum-product algorithm" [Huang & Frey, UAI2008]



- CDN:
  - multiplicative FG
  - each function node: a CDF
  - $\Rightarrow$ the global function is a CDF
  - efficiently solves structured ranking problems

Missed opportunity ...

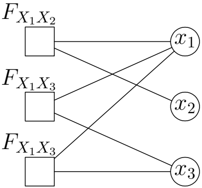
- Year 2005 ... met Frey in Toronto
- Frey: "How about a model for CDF?"
- "Cumulative distribution networks and derivative-sum-product algorithm" [Huang & Frey, UAI2008]



- CDN:
  - multiplicative FG
  - each function node: a CDF
  - $\Rightarrow$ the global function is a CDF
  - efficiently solves structured ranking problems

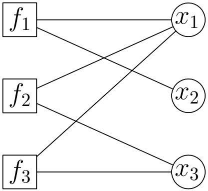
Missed opportunity ...

- Year 2005 ... met Frey in Toronto
- Frey: "How about a model for CDF?"
- "Cumulative distribution networks and derivative-sum-product algorithm" [Huang & Frey, UAI2008]



- CDN:
  - multiplicative FG
  - each function node: a CDF
  - $\Rightarrow$ the global function is a CDF
  - efficiently solves structured ranking problems

Conditional/marginal independence properties



- Multiplicative FG: conditional indepdence;
- Convolutional FG: marginal independence;
- CDN: marginal independence

Again, it is Frank Kschischang ...

- Early 2010 ...
- Kschischang: "Look at this!"
  - "Constrained coding as networks of relations" [Schwartz & Bruck 2008]
  - "Holographic algorithms" [Valiant 2004]
    - Counting problem
    - Holographic reduction: $\sum\prod \to \sum\prod$
    - Holant theorem
- Apparent "normal" structure
- Do something with holographic algorithm?
- My question: the magic of Holant theorem ...

Again, it is Frank Kschischang ...

- Early 2010 ...
- Kschischang: "Look at this!"
  - "Constrained coding as networks of relations" [Schwartz & Bruck 2008]
  - "Holographic algorithms" [Valiant 2004]
    - Counting problem
    - Holographic reduction: $\sum\prod \rightarrow \sum\prod$
    - Holant theorem
- Apparent "normal" structure
- Do something with holographic algorithm?
- My question: the magic of Holant theorem ...

Again, it is Frank Kschischang ...

- Early 2010 ...
- Kschischang: "Look at this!"
  - "Constrained coding as networks of relations" [Schwartz & Bruck 2008]
  - "Holographic algorithms" [Valiant 2004]
    - Counting problem
    - Holographic reduction: $\sum\prod \to \sum\prod$
    - Holant theorem
- Apparent "normal" structure
- Do something with holographic algorithm?
- My question: the magic of Holant theorem ...

Again, it is Frank Kschischang ...

- Early 2010 ...
- Kschischang: "Look at this!"
  - "Constrained coding as networks of relations" [Schwartz & Bruck 2008]
  - "Holographic algorithms" [Valiant 2004]
    - Counting problem
    - Holographic reduction: $\sum \prod \to \sum \prod$
    - Holant theorem
- Apparent "normal" structure
- Do something with holographic algorithm?
- My question: the magic of Holant theorem ...

Again, it is Frank Kschischang ...

- Early 2010 ...
- Kschischang: "Look at this!"
  - "Constrained coding as networks of relations" [Schwartz & Bruck 2008]
  - "Holographic algorithms" [Valiant 2004]
    - Counting problem
    - Holographic reduction: $\sum\prod \rightarrow \sum\prod$
    - Holant theorem
- Apparent "normal" structure
- Do something with holographic algorithm?
- My question: the magic of Holant theorem ...

Give it the student ...

- Ali Al-Bashebsheh

- Reformulated normal factor graphs
- Generalized Holant theorem
- Unified normal graph duality theorem and Holant theorem

Give it the student ...

- Ali Al-Bashebsheh

- Reformulated normal factor graphs
- Generalized Holant theorem
- Unified normal graph duality theorem and Holant theorem

Two concurrent submissions ...

- Editor Pascal Vontobel

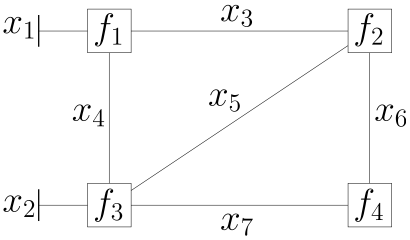
- “Normal factor graphs and holographic transformations " [Al-Bashebsheh & Mao 2011]
- “Codes on graphs: duality and MacWilliams identities” [Forney 2011]
- Extensive comments, rounds of rewriting

Two concurrent submissions ...

- Editor Pascal Vontobel

- “Normal factor graphs and holographic transformations " [Al-Bashebsheh & Mao 2011]
- “Codes on graphs: duality and MacWilliams identities” [Forney 2011]
- Extensive comments, rounds of rewriting

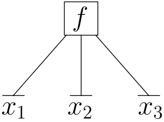
Normal factor graphs under new semantics ...

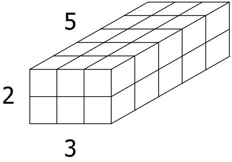


$$
\begin{aligned}
Z_{\mathcal{G}}(x_1, x_2) &:= \sum_{x_3,\ldots,x_7} f_1(x_1, x_3, x_4) f_2(x_3, x_5, x_6) f_3(x_2, x_4, x_5, x_7) f_4(x_6, x_7). \\
&:= \langle f_1, f_2, f_3, f_4 \rangle. \\
&:= \langle f_1(x_1, x_3, x_4), f_2(x_3, x_5, x_6), f_3(x_2, x_4, x_5, x_7), f_4(x_6, x_7). \rangle
\end{aligned}
$$

Functions as multidimensional arrays

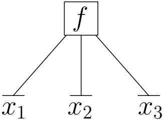
- $|\mathcal{X}_1| = 2, |\mathcal{X}_2| = 3$, and $|\mathcal{X}_3| = 5$.

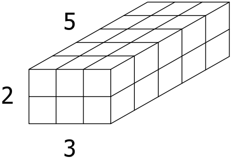


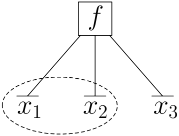


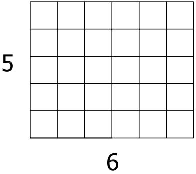
Functions as multidimensional arrays

- $|\mathcal{X}_1| = 2, |\mathcal{X}_2| = 3$, and $|\mathcal{X}_3| = 5$.



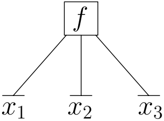


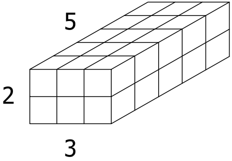


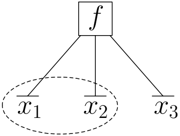


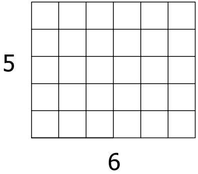
Functions as multidimensional arrays

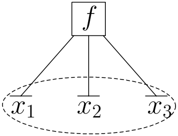
- $|\mathcal{X}_1| = 2, |\mathcal{X}_2| = 3$, and $|\mathcal{X}_3| = 5$.

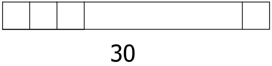












Example (Equality Indicator)

$$\delta(s, s') := \begin{cases} 1, & s = s' \\ 0, & \text{otherwise} \end{cases}$$

- $\delta(\cdot, \cdot)$: identity matrix

Example ("Transformer")

- Function $\Phi : \chi \times \chi \to \mathbb{C}$ is called a transformer if it corresponds to an invertible matrix.
- "Dual pair of transformers" Φ and $\hat{\Phi}$: an inverse pair of matrices
    - $\langle \Phi(s, t), \hat{\Phi}(t, s') \rangle = \delta(s, s')$

Example (Equality Indicator)

$$\delta(s, s') := \begin{cases} 1, & s = s' \\ 0, & \text{otherwise} \end{cases}$$

- $\delta(\cdot, \cdot)$: identity matrix

Example ("Transformer")

- Function $\Phi : \chi \times \chi \to \mathbb{C}$ is called a transformer if it corresponds to an invertible matrix.
- "Dual pair of transformers" Φ and $\widehat{\Phi}$: an inverse pair of matrices
  - $\langle \Phi(s, t), \widehat{\Phi}(t, s') \rangle = \delta(s, s')$

Example (Equality Indicator)

$$\delta(s, s') := \begin{cases} 1, & s = s' \\ 0, & \text{otherwise} \end{cases}$$

- $\delta(\cdot, \cdot)$: identity matrix

Example ("Transformer")

- Function $\Phi : \chi \times \chi \to \mathbb{C}$ is called a transformer if it corresponds to an invertible matrix.
- "Dual pair of transformers" Φ and $\widehat{\Phi}$: an inverse pair of matrices
  - $\langle \Phi(s, t), \widehat{\Phi}(t, s') \rangle = \delta(s, s')$

Example (Equality Indicator)

$$\delta(s,s') := \begin{cases} 1, & s = s' \\ 0, & \text{otherwise} \end{cases}$$

- $\delta(\cdot,\cdot)$: identity matrix

Example ("Transformer")

- Function $\Phi : \chi \times \chi \to \mathbb{C}$ is called a transformer if it corresponds to an invertible matrix.
- "Dual pair of transformers" Φ and $\widehat{\Phi}$: an inverse pair of matrices
  - $\langle \Phi(s,t), \widehat{\Phi}(t,s') \rangle = \delta(s,s')$

Example (Equality Indicator)

$$\delta(s,s') := \begin{cases} 1, & s = s' \\ 0, & \text{otherwise} \end{cases}$$

- $\delta(\cdot,\cdot)$: identity matrix

Example ("Transformer")

- Function $\Phi : \chi \times \chi \to \mathbb{C}$ is called a transformer if it corresponds to an invertible matrix.
- "Dual pair of transformers" Φ and $\widehat{\Phi}$: an inverse pair of matrices
  - $\langle \Phi(s,t), \widehat{\Phi}(t,s') \rangle = \delta(s,s')$

Functions as linear maps

$\langle f, g \rangle$:

- $\langle f(s), g(s) \rangle$: vector-vector dot product
- $\langle f(s), g(s, t) \rangle$: vector-matrix product
- $\langle f(s, t), g(t, u) \rangle$: matrix-matrix product
- $\langle f(s), g(t) \rangle$: vector outer product, matrix Kronecker product, tensor product etc

NFGs are linear algebraic expressions written graphically.

Functions as linear maps

$\langle f, g \rangle$:

- $\langle f(s), g(s) \rangle$: vector-vector dot product
- $\langle f(s), g(s,t) \rangle$: vector-matrix product
- $\langle f(s,t), g(t,u) \rangle$: matrix-matrix product
- $\langle f(s), g(t) \rangle$: vector outer product, matrix Kronecker product, tensor product etc

NFGs are linear algebraic expressions written graphically.

Functions as linear maps

$\langle f, g \rangle$:

- $\langle f(s), g(s) \rangle$: vector-vector dot product
- $\langle f(s), g(s,t) \rangle$: vector-matrix product
- $\langle f(s,t), g(t,u) \rangle$: matrix-matrix product
- $\langle f(s), g(t) \rangle$: vector outer product, matrix Kronecker product, tensor product etc

NFGs are linear algebraic expressions written graphically.

Functions as linear maps

$\langle f, g \rangle$:

- $\langle f(s), g(s) \rangle$: vector-vector dot product
- $\langle f(s), g(s, t) \rangle$: vector-matrix product
- $\langle f(s, t), g(t, u) \rangle$: matrix-matrix product
- $\langle f(s), g(t) \rangle$: vector outer product, matrix Kronecker product, tensor product etc

NFGs are linear algebraic expressions written graphically.

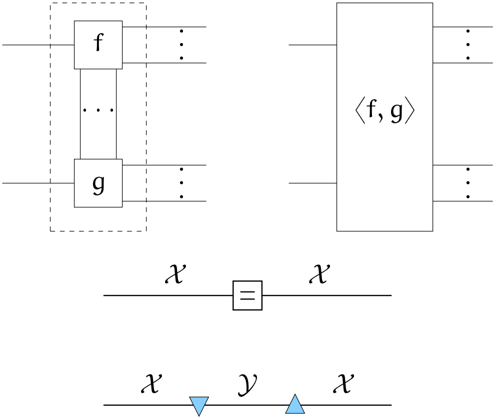
Functions as linear maps

$\langle f, g \rangle$:

- $\langle f(s), g(s) \rangle$: vector-vector dot product
- $\langle f(s), g(s,t) \rangle$: vector-matrix product
- $\langle f(s,t), g(t,u) \rangle$: matrix-matrix product
- $\langle f(s), g(t) \rangle$: vector outer product, matrix Kronecker product, tensor product etc

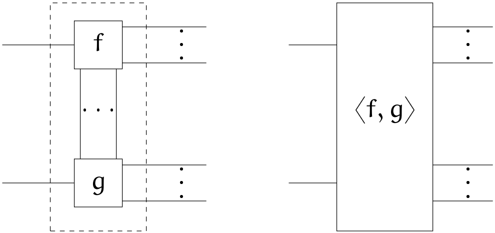
NFGs are linear algebraic expressions written graphically.

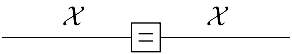
Opening/closing the box [Vontobel, Loeliger, 2002]

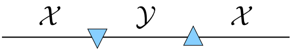


Opening/closing the box is a powerful technique

Opening/closing the box [Vontobel, Loeliger, 2002]

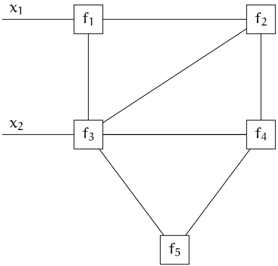


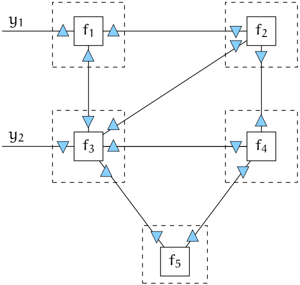




Opening/closing the box is a powerful technique

Holographic transformation





$Z_{\mathcal{G}}$ $\qquad$ $Z_{\mathcal{G}^H}$

Theorem (Generalized Holant Theorem)

$Z_{\mathcal{G}^H}$ is "externally transformed" version of $Z_{\mathcal{G}}$.

Generalized Holant Theorem

- When external transformers are $\delta(\cdot, \cdot)$
  - reduces to Valiant's Holant Theorem
- When all transformers are Fourier kernels
  - reduces to General NFG Duality Theorem
  - when each function is a subgroup-indicator function
    - reduces to Forney's Normal Graph Duality Theorem

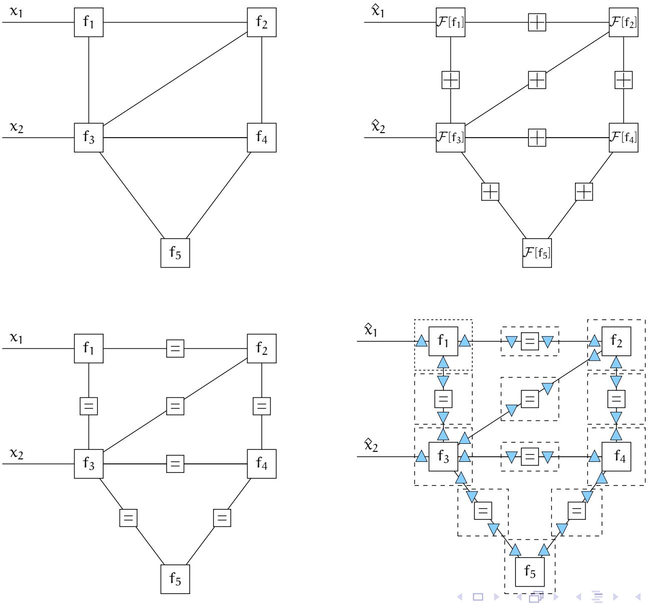
Generalized Holant Theorem

- When external transformers are $\delta(\cdot, \cdot)$
  - reduces to Valiant's Holant Theorem
- When all transformers are Fourier kernels
  - reduces to General NFG Duality Theorem
  - when each function is a subgroup-indicator function
    - reduces to Forney's Normal Graph Duality Theorem

Generalized Holant Theorem

- When external transformers are $\delta(\cdot, \cdot)$
  - reduces to Valiant's Holant Theorem
- When all transformers are Fourier kernels
  - reduces to General NFG Duality Theorem
  - when each function is a subgroup-indicator function
    - reduces to Forney's Normal Graph Duality Theorem

General NFG Duality Theorem



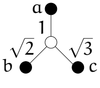
Holographic algorithm: PerfMatch(H)

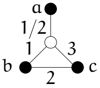
- H: weighted graph
- PerfMatch(H)

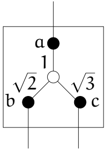
$$\pi(H) := \sum_{M \in Q(H)} \prod_{e \in M} w(e),$$

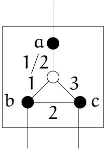
where Q(H) the set of all perfect matchings of H.

- If H is planar, $\pi(H)$ is poly. solvable by FKT algorithm
- Matchgate (G, U), where G is a weighted graph, $U \subseteq V(G)$.
- Signature of matchgate (G, U) is a function $\mu(x_U)$ ...









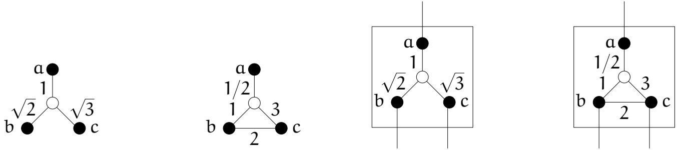
Holographic algorithm: PerfMatch(H)

- H: weighted graph
- PerfMatch(H)

$$\pi(H) := \sum_{M \in Q(H)} \prod_{e \in M} w(e),$$

where Q(H) the set of all perfect matchings of H.

- If H is planar, $\pi(H)$ is poly. solvable by FKT algorithm
- Matchgate (G, U), where G is a weighted graph, $U \subseteq V(G)$.
- Signature of matchgate (G, U) is a function $\mu(x_U)$ ...



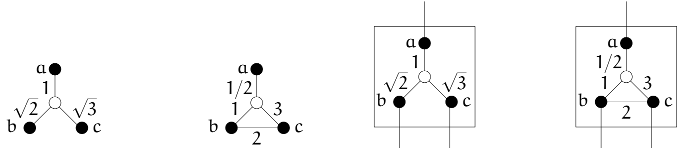
Holographic algorithm: PerfMatch(H)

- H: weighted graph
- PerfMatch(H)

$$\pi(H) := \sum_{M \in Q(H)} \prod_{e \in M} w(e),$$

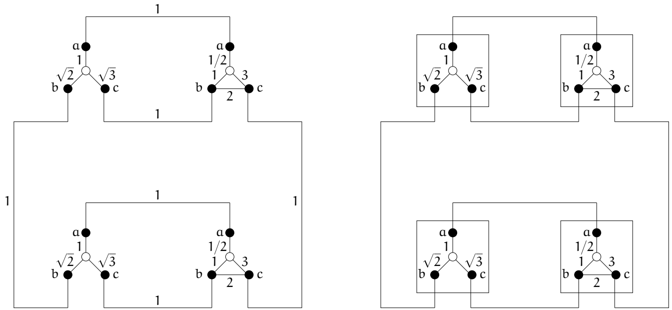
where Q(H) the set of all perfect matchings of H.

- If H is planar, $\pi(H)$ is poly. solvable by FKT algorithm
- Matchgate (G, U), where G is a weighted graph, $U \subseteq V(G)$.
- Signature of matchgate (G, U) is a function $\mu(x_U)$ ...



Holographic algorithm: a graph-theoretic property

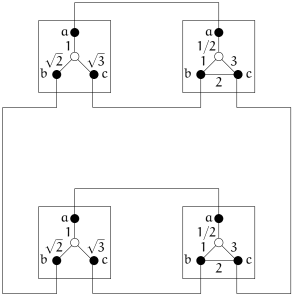
Lemma: $\pi(H) = \sum_{x_{\mathcal{E}}} \prod_{i=1}^{m} \mu_i(x_{\mathcal{E}(i)})$.



Holographic algorithm

$$\text{Solve } Z := \sum_{x_{\mathcal{E}}} \prod_v f_v(x_{\mathcal{E}(v)})$$

$$\Downarrow$$



$$\Downarrow$$

FKT

Don't forget the mathematicians ...

- Vontobel pointed to Trace Diagram, Birdtracks ...
- "Unshackling linear algebra from linear notation" [Peterson 2009]
  "Group Theory: Birdtracks, Lie's, and Exceptional Group", [Cvitanovic, 2008]
- Trace diagram is contained in NFG framework
  - Trace diagrams are NFGs with degree 1, 2, or $|\chi|$
- "NFGs: a diagrammatic approach to linear algebra" [Al-Bashebsheh, Mao & Vontobel ISIT2011]

Don't forget the mathematicians ...

- Vontobel pointed to Trace Diagram, Birdtracks ...
- "Unshackling linear algebra from linear notation" [Peterson 2009]
  "Group Theory: Birdtracks, Lie's, and Exceptional Group", [Cvitanovic, 2008]
- Trace diagram is contained in NFG framework
  - Trace diagrams are NFGs with degree 1, 2, or $|\chi|$
- "NFGs: a diagrammatic approach to linear algebra" [Al-Bashebsheh, Mao & Vontobel ISIT2011]

Don't forget the mathematicians ...

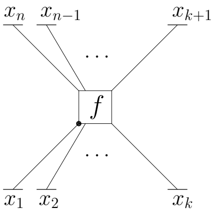
- Vontobel pointed to Trace Diagram, Birdtracks ...
- "Unshackling linear algebra from linear notation" [Peterson 2009]
  "Group Theory: Birdtracks, Lie's, and Exceptional Group", [Cvitanovic, 2008]
- Trace diagram is contained in NFG framework
  - Trace diagrams are NFGs with degree 1, 2, or $|\chi|$
- "NFGs: a diagrammatic approach to linear algebra" [Al-Bashebsheh, Mao & Vontobel ISIT2011]

Don't forget the mathematicians ...

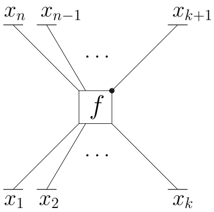
- Vontobel pointed to Trace Diagram, Birdtracks ...
- "Unshackling linear algebra from linear notation" [Peterson 2009]
  "Group Theory: Birdtracks, Lie's, and Exceptional Group", [Cvitanovic, 2008]
- Trace diagram is contained in NFG framework
  - Trace diagrams are NFGs with degree 1, 2, or $|\chi|$
- "NFGs: a diagrammatic approach to linear algebra" [Al-Bashebsheh, Mao & Vontobel ISIT2011]

Ciliated function nodes

- Draw the edges of f counter-clockwise, with first edge ciliated



$f(x_1, \dots, x_n)$

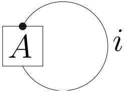


$f(x_{k+1}, \dots, x_n, x_1, \dots, x_k)$

Ciliated function nodes

$\mathsf{A} \cdot \mathsf{B}$ $\quad$ $\mathsf{A} \cdot \mathsf{B}^\mathsf{T}$ $\quad$ $\mathsf{A}^\mathsf{T} \cdot \mathsf{B}^\mathsf{T}$ $\quad$ $\mathsf{A}^\mathsf{T} \cdot \mathsf{B}$

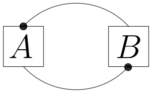
Trace



$$Z = \sum_i A(i,i) = \mathrm{tr}(A)$$

$\mathrm{tr}(A \cdot B) = \mathrm{tr}(B \cdot A)$

Proof:



Read the graph in two ways:

$Z = \mathrm{tr}(A \cdot B)$ and $Z = \mathrm{tr}(B \cdot A)$. $\square$

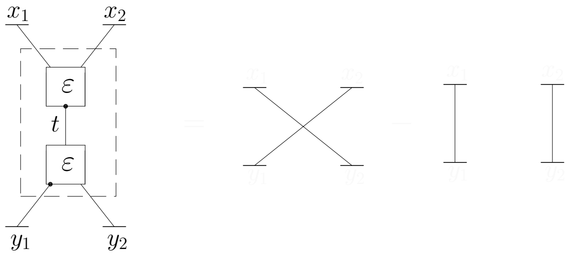
Levi-Civita symbol

- S_n: the permutation group on $\{1, \ldots, n\}$, $n = |\chi|$.
- Levi-Civita symbol, $\varepsilon : \{1, \ldots, n\}^n \to \mathbb{C}$ s.t.

$$\varepsilon(x_1, \ldots, x_n) = \begin{cases} \operatorname{sgn} \begin{pmatrix} 1 & \cdots & n \\ x_1 & \cdots & x_n \end{pmatrix}, & \begin{pmatrix} 1 & \cdots & n \\ x_1 & \cdots & x_n \end{pmatrix} \in S_n \\ 0, & \text{otherwise} \end{cases}$$

- Contraction ($n = 3$).

$$\sum_t \varepsilon(y_1, y_2, t)\varepsilon(t, x_2, x_1) = \delta(x_1, y_2)\delta(x_2, y_1) - \delta(x_1, y_1)\delta(x_2, y_2).$$



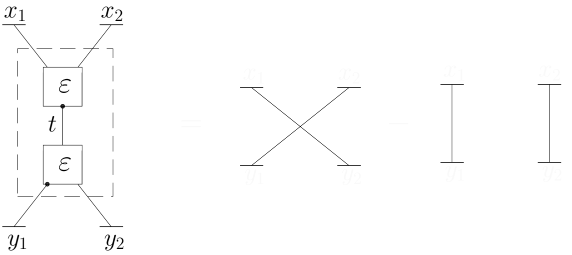
Levi-Civita symbol

- S_n: the permutation group on $\{1,\ldots,n\}$, $n=|\chi|$.
- Levi-Civita symbol, $\varepsilon:\{1,\ldots,n\}^n\to\mathbb{C}$ s.t.

$$\varepsilon(x_1,\ldots,x_n)=\begin{cases}\operatorname{sgn}\begin{pmatrix}1&\cdots&n\\x_1&\cdots&x_n\end{pmatrix}, & \begin{pmatrix}1&\cdots&n\\x_1&\cdots&x_n\end{pmatrix}\in S_n\\ 0, & \text{otherwise}\end{cases}$$

- Contraction ($n=3$).

$$\sum_t \varepsilon(y_1,y_2,t)\varepsilon(t,x_2,x_1)=\delta(x_1,y_2)\delta(x_2,y_1)-\delta(x_1,y_1)\delta(x_2,y_2).$$



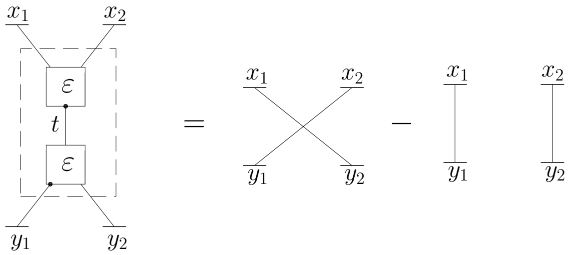
Levi-Civita symbol

- S_n: the permutation group on $\{1, \ldots, n\}$, $n = |\chi|$.
- Levi-Civita symbol, $\varepsilon : \{1, \ldots, n\}^n \to \mathbb{C}$ s.t.

$$\varepsilon(x_1, \ldots, x_n) = \begin{cases} \operatorname{sgn} \begin{pmatrix} 1 & \cdots & n \\ x_1 & \cdots & x_n \end{pmatrix}, & \begin{pmatrix} 1 & \cdots & n \\ x_1 & \cdots & x_n \end{pmatrix} \in S_n \\ 0, & \text{otherwise} \end{cases}$$

- Contraction $(n = 3)$.

$$\sum_t \varepsilon(y_1, y_2, t)\varepsilon(t, x_2, x_1) = \delta(x_1, y_2)\delta(x_2, y_1) - \delta(x_1, y_1)\delta(x_2, y_2).$$

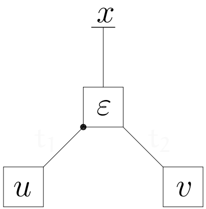


Vector cross product

- $u, v : \{1,2,3\} \to \mathbb{F} \Leftrightarrow u, v \in \mathbb{F}^3$.

- Cross Product.

$$Z(x) = \sum_{t_1,t_2} u(t_1)v(t_2)\varepsilon(t_1,t_2,x).$$



$$Z(1) = u(2)v(3) - u(3)v(2)$$
$$Z(2) = u(3)v(1) - u(1)v(3)$$
$$Z(3) = u(1)v(2) - u(2)v(1)$$

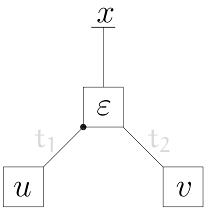
$$Z = u \times v.$$

Vector cross product

- $u, v : \{1, 2, 3\} \to \mathbb{F} \Leftrightarrow u, v \in \mathbb{F}^3$.

- Cross Product.

$$Z(x) = \sum_{t_1, t_2} u(t_1) v(t_2) \varepsilon(t_1, t_2, x).$$



$$Z(1) = u(2)v(3) - u(3)v(2)$$
$$Z(2) = u(3)v(1) - u(1)v(3)$$
$$Z(3) = u(1)v(2) - u(2)v(1)$$

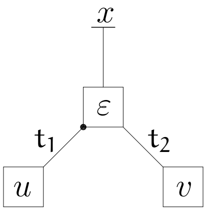
$$Z = u \times v.$$

Vector cross product

- $u, v : \{1, 2, 3\} \to \mathbb{F} \Leftrightarrow u, v \in \mathbb{F}^3$.

- Cross Product.

$$Z(x) = \sum_{t_1, t_2} u(t_1) v(t_2) \varepsilon(t_1, t_2, x).$$



$$Z(1) = u(2)v(3) - u(3)v(2)$$
$$Z(2) = u(3)v(1) - u(1)v(3)$$
$$Z(3) = u(1)v(2) - u(2)v(1)$$

$$Z = u \times v.$$

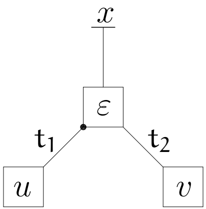
Vector cross product

- $u, v : \{1, 2, 3\} \to \mathbb{F} \Leftrightarrow u, v \in \mathbb{F}^3$.

- Cross Product.

$$Z(x) = \sum_{t_1, t_2} u(t_1) v(t_2) \varepsilon(t_1, t_2, x).$$

$$Z(1) = u(2)v(3) - u(3)v(2)$$
$$Z(2) = u(3)v(1) - u(1)v(3)$$
$$Z(3) = u(1)v(2) - u(2)v(1)$$



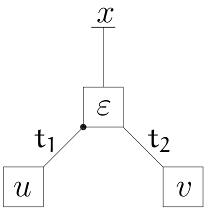
$Z = u \times v$.

Vector cross product

- $u, v : \{1, 2, 3\} \to \mathbb{F} \Leftrightarrow u, v \in \mathbb{F}^3$.

- Cross Product.

$$Z(x) = \sum_{t_1, t_2} u(t_1) v(t_2) \varepsilon(t_1, t_2, x).$$



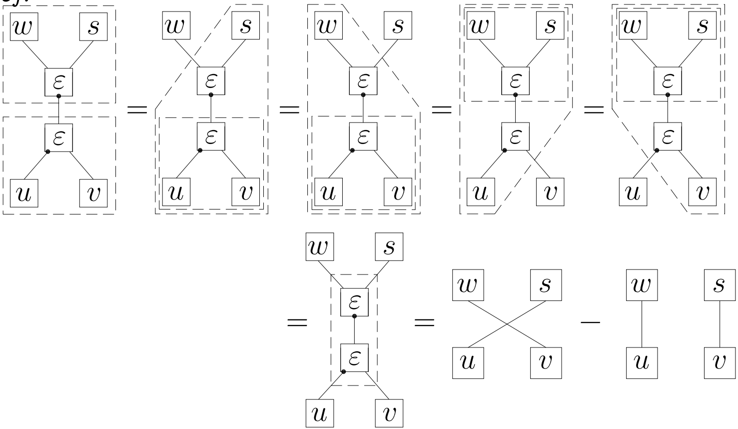
$$Z(1) = u(2)v(3) - u(3)v(2)$$
$$Z(2) = u(3)v(1) - u(1)v(3)$$
$$Z(3) = u(1)v(2) - u(2)v(1)$$

$$Z = u \times v.$$

Cross product identities

$$
\begin{aligned}
(u \times v) \cdot (s \times w) &= ((u \times v) \times s) \cdot w = (w \times (u \times v)) \cdot s \\
&= ((s \times w) \times u) \cdot v = (v \times (s \times w)) \cdot u \\
&= (u \cdot s)(v \cdot w) - (u \cdot w)(v \cdot s).
\end{aligned}
$$

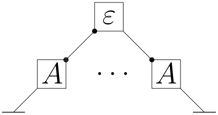
Proof:



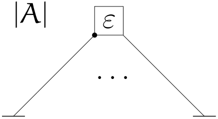
Determinant

$$|A| := \sum_{\sigma \in S_n} \mathrm{sgn}(\sigma) \prod_{j=1}^{n} A(j, \sigma(j)).$$

Lemma:

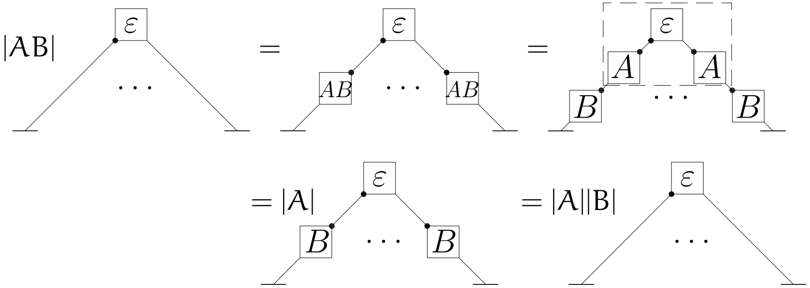


$= \ |A|$



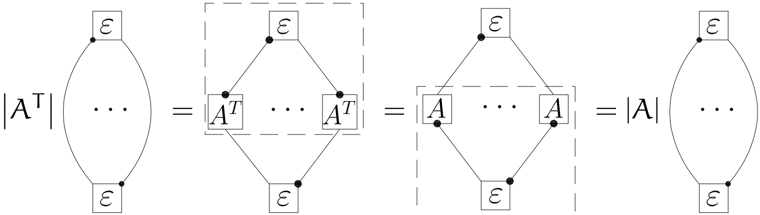
$|A \cdot B| = |A||B|$

Proof:



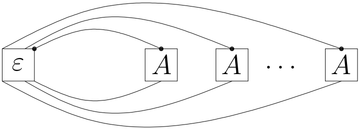
$|A^{\mathsf{T}}| = |A|$

Proof:



Pfaffian

- A is a $2n \times 2n$ skew-symmetric matrix.
- $\mathrm{Pf}(A) = \frac{1}{2^n n!} \sum_{\sigma \in S_{2n}} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} A\big(\sigma(2i-1), \sigma(2i)\big).$

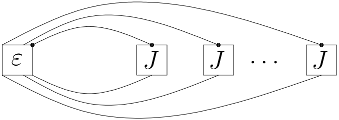


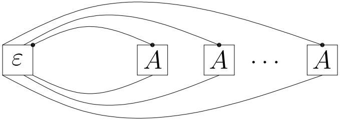
- $Z = n! \cdot 2^n \cdot \mathrm{Pf}(A)$
- Affirms a conjecture of Peterson [Peterson, 2009].

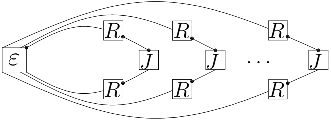
Pfaffian and Determinant

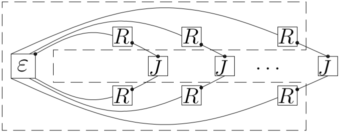
$J := \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix}$, $A := R^T \cdot J \cdot R$ for some R. Then $\mathrm{Pf}(A)^2 = |A|$.

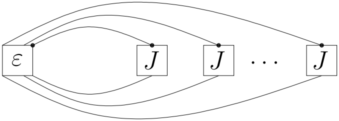
Proof: $\mathrm{Pf}(J) = \pm 1$

$\pm\mathrm{Pf}(A)$ 

$=$ 

$=$ 

$=$ 

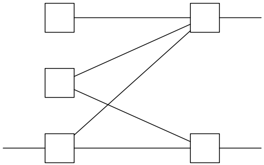
$= |R|$ 

$\pm\mathrm{Pf}(A) = |R| \Rightarrow \mathrm{Pf}(A)^2 = |R|^2 = |A|.$

Going probabilistic ...

"NFGs as probabilistic model"
[Al-Bashebsheh & Mao, 2011, soon available on arxiv]

- Internal edges: latent variabels
- External edges: observed variables
- Exterior function: joint distribution of observed variables (up to scale)

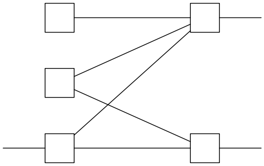


- Function:
  - compatibility/constraints/interaction protential
  - source of randomness, i.e., distribution
- Bipartite normal: WLOG

Going probabilistic ...

"NFGs as probabilistic model"
[Al-Bashebsheh & Mao, 2011, soon available on arxiv]

- Internal edges: latent variabels
- External edges: observed variables
- Exterior function: joint distribution of observed variables (up to scale)

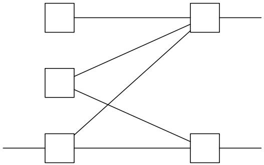


- Function:
  - compatibility/constraints/interaction protential
  - source of randomness, i.e., distribution
- Bipartite normal: WLOG

Going probabilistic ...

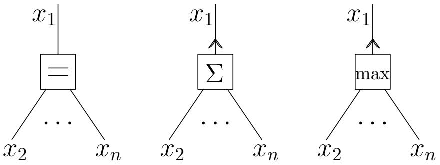
"NFGs as probabilistic model"
[Al-Bashebsheh & Mao, 2011, soon available on arxiv]

- Internal edges: latent variabels
- External edges: observed variables
- Exterior function: joint distribution of observed variables (up to scale)

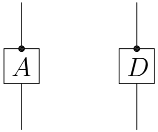


- Function:
  - compatibility/constraints/interaction protential
  - source of randomness, i.e., distribution
- Bipartite normal: WLOG

Special functions



"Cumulus" and "difference" transformers

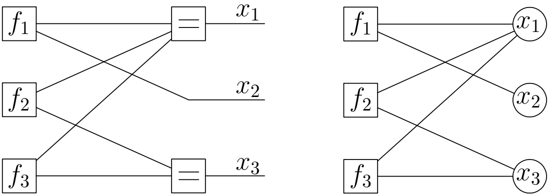


- χ: an ordered set (integers) with "$<$" well defined.
- $A(x, x') = [x' \leqslant x]$
- E.g. $|\chi| = 2$, $A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$;

  $|\chi| = 3$, $A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
- $\langle f_X(x'), A(x, x') \rangle = F_X(x)$.
- $D := A^{-1}$, e.g.$D = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$ for $|\chi|$=3.

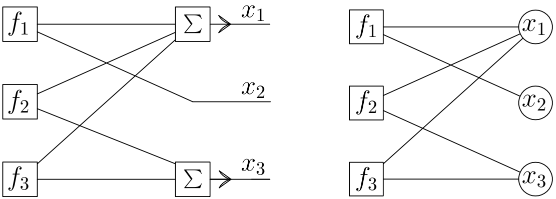
Bipartite NFG reduces to FG

- One partitiion: arbitrary potential functions
- The other partition: equality indicators
- $\Rightarrow$ NFG model reduces to FG model



Bipartite NFG reduces to convolutional FG

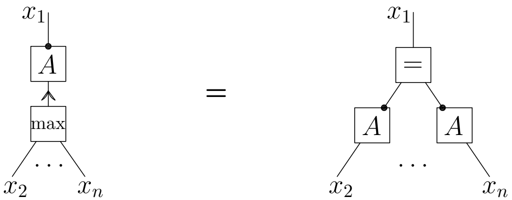
- One side: distributions of indep. collections of latent RVs
- The other side: sum indicators
- No RVs from the same collection connect to the same indicator
- $\Rightarrow$ NFG model reduces to convolutional FG model



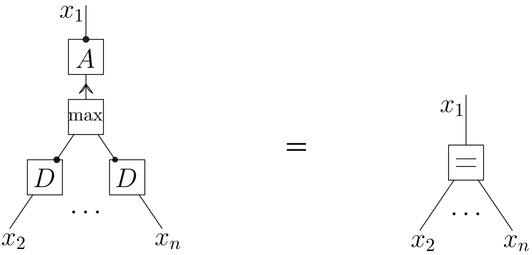
$$\sum_{u,v} f(x,u)g(v,y)[w=u+v] = \sum_{u} f(x,u)g(w-u,y) = f(x,w) * g(w,y)$$

CDN is a holographic transformed NFG

Lemma

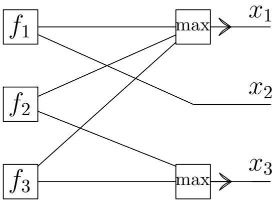


Corollary

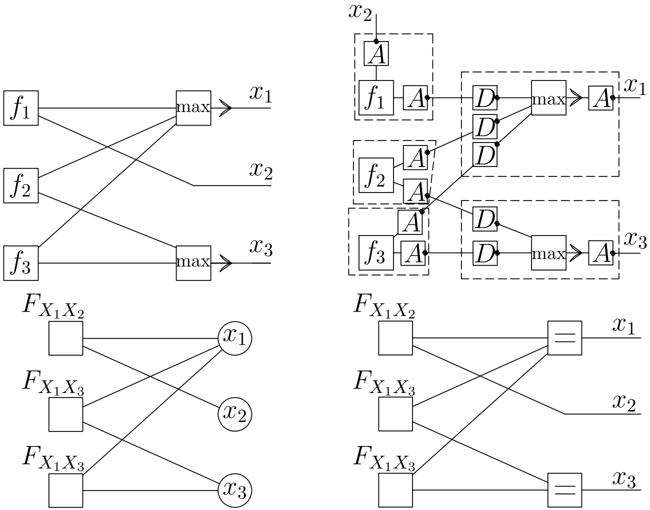


CDN is a holographic transformed NFG

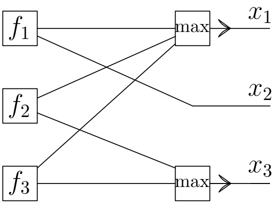
- One side: distributions of indep. collections of latent RVs
- The other side: max indicators
- No RVs from the same collection connect to the same indicator

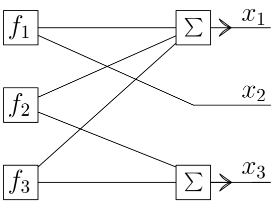


CDN is a holographic transformed NFG



The independence coincidence of Convolutional FG and CDN





- Both are generative models
- In fact, the independence property holds when changing indicator to conditional distributions
- Changing the right-side functions gives rise to a family of infinite models ... what applications?

The End

NFGs are linear algebraic expressions written graphically

Opening/closing the box is a powerful technique

NFGs are potential tools for inference

Outlook: continuous alphabets

The End

NFGs are linear algebraic expressions written graphically

Opening/closing the box is a powerful technique

NFGs are potential tools for inference

Outlook: continuous alphabets

The End

NFGs are linear algebraic expressions written graphically

Opening/closing the box is a powerful technique

NFGs are potential tools for inference

Outlook: continuous alphabets

The End

NFGs are linear algebraic expressions written graphically

Opening/closing the box is a powerful technique

NFGs are potential tools for inference

Outlook: continuous alphabets

The End

NFGs are linear algebraic expressions written graphically

Opening/closing the box is a powerful technique

NFGs are potential tools for inference

Outlook: continuous alphabets