# Fourier Representations for Probabilistic Inference

**Yexiang Xue**[1],

Stefano Ermon[2], Ronan Le Bras[1], Carla P. Gomes[1], Bart Selman[1]

[1] Computer Science Dept, Cornell University,

[2] Computer Science Dept, Stanford University.

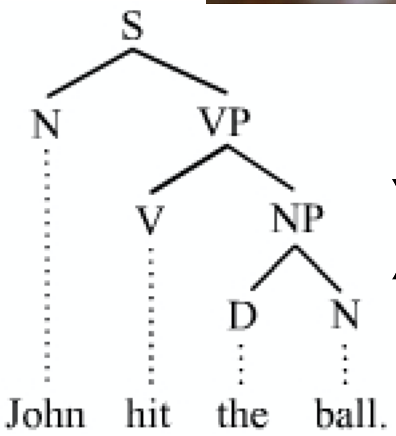
# High-dimensional Probabilistic Distributions in Machine Learning

- Modelling high-dimensional probabilistic distributions is ubiquitous in machine learning.

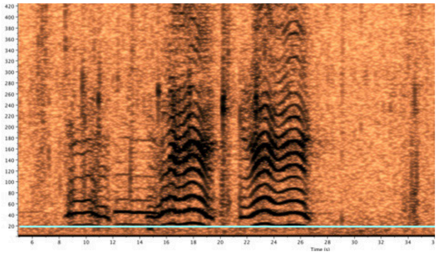
Pr(  , cat)>0.8?

Pr( 

, right turn)?

Pr( 

)>0.9?

Pr(  , elephant call)>0.6?

[1] https://www.youtube.com
[2] http://zackkanter.com/2015/01/23/how-ubers-autonomous-cars-will-destroy-10-million-jobs-by-2025
[3] http://www.birds.cornell.edu/brp/elephant/cyclotis/language/infrasound.html
[4] https://en.wikipedia.org/wiki/Parse_tree

# Compact Representations

- One of the most valuable insight for probabilistic inference is how to represent high dimensional probability distributions in a **compact form.**

- Many ramifications:

  Efficient **Inference** Algorithms

  **Machine Learning** with better predictive power and generalization

# Compact Representations

- Most compact representations are based on some form of **independence.**

Exploited in Graphical models, Mean Field Approximations, Tree Reweighted Approximations, Hidden Markov Models, and many more…

$$P(x_1 \dots, x_m, y_1 \dots, y_n) \approx P(x_1 \dots, x_m) \cdot P(y_1 \dots, y_n).$$

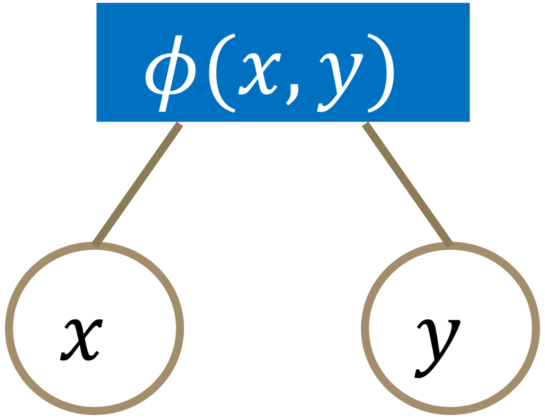
# Fourier Representation

- We propose the **Fourier representation of pseudo Boolean functions** as a **novel compact** way of representing high dimensional probability distributions.
- Fourier representations provide **a natural and well motivated way of approximating** high dimensional probability distributions.
  - I) **We show that for a general class of functions, most probability mass *concentrates on low degree coefficients in their Fourier spectrum.***
  - II) **Dropping higher degree coefficients leads to *good approximations measured in the L2 distance* (contrast with existing approaches such as minibucket and variational approaches).**
- Strong results obtained by applying Fourier representations for probabilistic inference. **Orders of magnitudes improvement for partition function** compared to competing approaches.

# Fourier Representation

Table Representation

| $x$ | $y$ | $\phi(\mathbf{x},\mathbf{y})$ |
|---|---|---|
| -1 | -1 | $\phi_1$ |
| -1 | 1 | $\phi_2$ |
| 1 | -1 | $\phi_3$ |
| 1 | 1 | $\phi_4$ |




$\boldsymbol{\phi}(\mathbf{x},\mathbf{y})$ is a function from $\{-1,1\}^2$ to **R+;**
$\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_4$ are real numbers.

Throughout this talk, we use:
-1 to represent **false**;
+1 to represent **true**.

# Fourier Representation

Table Representation

| $x$ | $y$ | $\phi(\mathrm{x},\mathrm{y})$ |
|---|---|---|
| -1 | -1 | $\phi_1$ |
| -1 | 1 | $\phi_2$ |
| 1 | -1 | $\phi_3$ |
| 1 | 1 | $\phi_4$ |

Interpolation

$$\phi(\mathrm{x},\mathrm{y}) = \frac{1-x}{2}\cdot\frac{1-y}{2}\cdot\phi_1 + \frac{1-x}{2}\cdot\frac{1+y}{2}\cdot\phi_2 + \frac{1+x}{2}\cdot\frac{1-y}{2}\cdot\phi_3 + \frac{1+x}{2}\cdot\frac{1+y}{2}\cdot\phi_4.$$

$\boldsymbol{\phi}(\mathbf{x},\mathbf{y})$ is a function from $\{-1,1\}^2$ to **R+;**
$\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_4$ are real numbers.

Throughout this talk, we use:
-1 to represent **false**;
+1 to represent **true**.

# Fourier Representation

Table Representation

| $x$ | $y$ | $\phi(x,y)$ |
|---|---|---|
| -1 | -1 | $\phi_1$ |
| -1 | 1 | $\phi_2$ |
| 1 | -1 | $\phi_3$ |
| 1 | 1 | $\phi_4$ |

$\boldsymbol{\phi}(\mathbf{x},\mathbf{y})$ is a function from $\{-1,1\}^2$ to **R+;**
$\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_4$ are real numbers.

Throughout this talk, we use:
-1 to represent **false**;
+1 to represent **true**.

Interpolation

$$\phi(x,y) = \frac{1-x}{2}\cdot\frac{1-y}{2}\cdot\phi_1 + \frac{1-x}{2}\cdot\frac{1+y}{2}\cdot\phi_2 + \frac{1+x}{2}\cdot\frac{1-y}{2}\cdot\phi_3 + \frac{1+x}{2}\cdot\frac{1+y}{2}\cdot\phi_4.$$

**Blue part evaluates to 1**

Green parts evaluate to 0.

# Fourier Representation

Table Representation

| $x$ | $y$ | $\phi(x,y)$ |
|---|---|---|
| -1 | -1 | $\phi_1$ |
| -1 | 1 | $\phi_2$ |
| 1 | -1 | $\phi_3$ |
| 1 | 1 | $\phi_4$ |

Interpolation

$$\phi(x,y) = \frac{1-x}{2}\cdot\frac{1-y}{2}\cdot\phi_1 + \frac{1-x}{2}\cdot\frac{1+y}{2}\cdot\phi_2 + \frac{1+x}{2}\cdot\frac{1-y}{2}\cdot\phi_3 + \frac{1+x}{2}\cdot\frac{1+y}{2}\cdot\phi_4.$$

Rearranging Terms

$$\phi(x,y) = {}^1\!/_4(\ \phi_1+\phi_2+\phi_3+\phi_4) + {}^1\!/_4(-\phi_1-\phi_2+\phi_3+\phi_4)\,x + {}^1\!/_4(-\phi_1+\phi_2-\phi_3+\phi_4)y + {}^1\!/_4(\ \phi_1-\phi_2-\phi_3+\phi_4)\,xy.$$

# Fourier Representation

Table Representation

| $x$ | $y$ | $\phi(x,y)$ |
|---|---|---|
| -1 | -1 | $\phi_1$ |
| -1 | 1 | $\phi_2$ |
| 1 | -1 | $\phi_3$ |
| 1 | 1 | $\phi_4$ |

Interpolation

$$\phi(x,y) = \frac{1-x}{2}\cdot\frac{1-y}{2}\cdot\phi_1 + \frac{1-x}{2}\cdot\frac{1+y}{2}\cdot\phi_2 + \frac{1+x}{2}\cdot\frac{1-y}{2}\cdot\phi_3 + \frac{1+x}{2}\cdot\frac{1+y}{2}\cdot\phi_4.$$

**Fourier Representation**

Rearranging Terms

$$\phi(x,y) = {}^1\!/_4(\ \phi_1 + \phi_2 + \phi_3 + \phi_4) + {}^1\!/_4(-\phi_1 - \phi_2 + \phi_3 + \phi_4)\,x + {}^1\!/_4(-\phi_1 + \phi_2 - \phi_3 + \phi_4)\,y + {}^1\!/_4(\ \phi_1 - \phi_2 - \phi_3 + \phi_4)\,xy$$

# Formal Definition

**Thm 1:** *(Hadamard-Fourier Transformation) Every weighted function* $f:\{-1,1\}^n \to \boldsymbol{R}$ *can be uniquely expressed as a multilinear polynomial:*

$$f(\boldsymbol{x}) = \sum_{S\subseteq[n]} c_S \prod_{i\in S} x_i = \sum_{S\subseteq[n]} \widehat{f(S)}\, \chi_S(\boldsymbol{x}).$$

*Where each* $c_S \in \boldsymbol{R}$. $[n]$ *is the power set of* $\{1, \dots, n\}$. *This polynomial is referred to as the Hadamard-Fourier expansion of* $f$.

Following standard notation:

$\prod_{i\in S} x_i \Leftarrow \chi_S(\boldsymbol{x})$ (Basis Function, also parity functions).

$c_S \Leftarrow \widehat{f(S)}$ (coefficients)

$\widehat{f(S)}$ is a degree-$k$ coefficient iff $|S| = k$.

# Formal Definition

**Thm 1:** *(Hadamard-Fourier Transformation) Every weighted function* $f\colon\{-1,1\}^n \to \boldsymbol{R}$ *can be uniquely expressed as a multilinear polynomial:*

$$f(\boldsymbol{x}) = \sum_{S\subseteq[n]} \boxed{c_S} \boxed{\prod_{i\in S} x_i} = \sum_{S\subseteq[n]} \boxed{\widehat{f(S)}} \boxed{\chi_S(\boldsymbol{x})}.$$

Basis Function
(also Parity functions)
Coefficients

*Where each* $c_S \in \boldsymbol{R}$. $[n]$ *is the power set of* $\{1,\ldots,n\}$. *This polynomial is referred to as the Hadamard-Fourier expansion of* $f$.

Following standard notation:

$\prod_{i\in S} x_i \Leftarrow \chi_S(\boldsymbol{x})$ (Basis Function, also parity functions).

$c_S \Leftarrow \widehat{f(S)}$ (coefficients)

$\widehat{f(S)}$ is a degree-$k$ coefficient iff $|S| = k$.

# Example Fourier Representation

Degree 1 term

$$\boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) = {}^{1}\!/_{4}(\ \phi_1+\phi_2+\phi_3+\phi_4) \ + {}^{1}\!/_{4}(-\phi_1-\phi_2+\phi_3+\phi_4)\,\boldsymbol{x} + {}^{1}\!/_{4}(-\phi_1+\phi_2-\phi_3+\phi_4)\boldsymbol{y} + {}^{1}\!/_{4}(\ \phi_1-\phi_2-\phi_3+\phi_4)\,\boldsymbol{xy}$$

Degree 2 term

Coefficients
(real numbers)

**Basis Functions,**
**(Also parity functions,**
**e.g., the product "*xy*" evaluates to -1 iff**
**an odd number of variables is set to "-1")**

# Why Fourier Representation?

**Value Representation**

$\boldsymbol{n}$ **binary variables → table of length $2^n$**

| $\boldsymbol{x}$ | $\boldsymbol{f(x)}$ |
|---|---|
| -1, -1, …-1,-1 | f(-1,-1,…,-1,-1) |
| -1,-1,…,-1,+1 | f(-1,-1,…,-1,+1) |
| -1,-1,…,+1,-1 | f(-1,-1,…,+1,-1) |
| … | … |
| … | … |
| … | … |
| +1,+1,…,+1,+1 | f(+1,+1,…,+1,+1) |

**Fourier Representation**

$$f(\boldsymbol{x}) = \sum_{S \subseteq [n]} \widehat{f(S)}\, \chi_S(\boldsymbol{x}).$$

**expression with $2^n$ terms**

## *Why Fourier?*

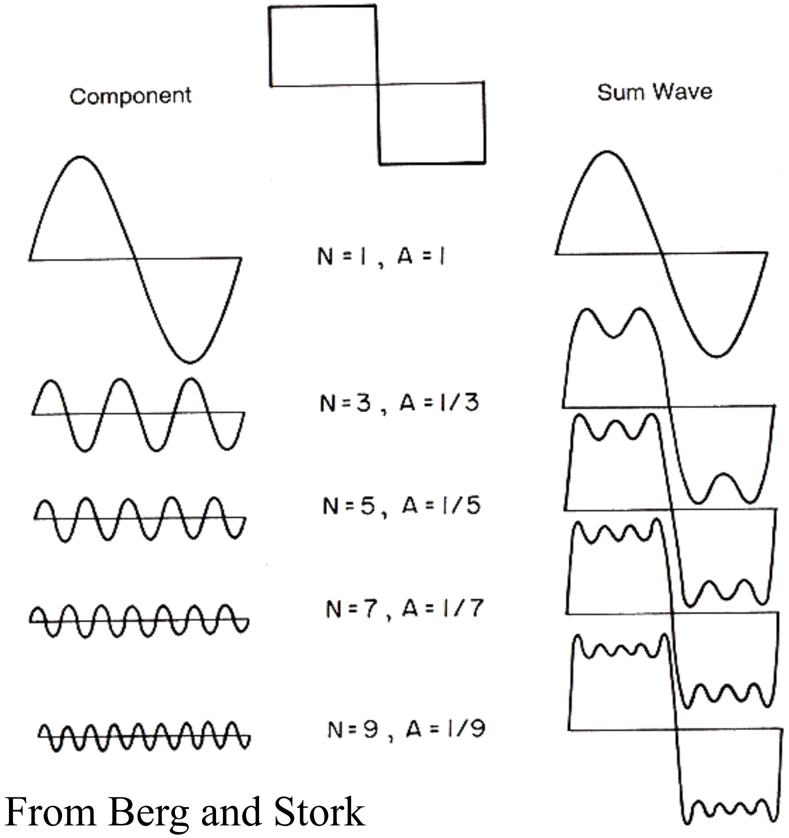
***Good properties for many distributions.***

a) **Often only a few lower-order terms needed**
b) **Can also truncate and get good approximation (L2 norm)**

# Weight Concentration

**Continuous Case**: signals decompose into components with different frequencies. High frequency components are often close to zero for a wide class of functions.

High Freq Comps are smaller

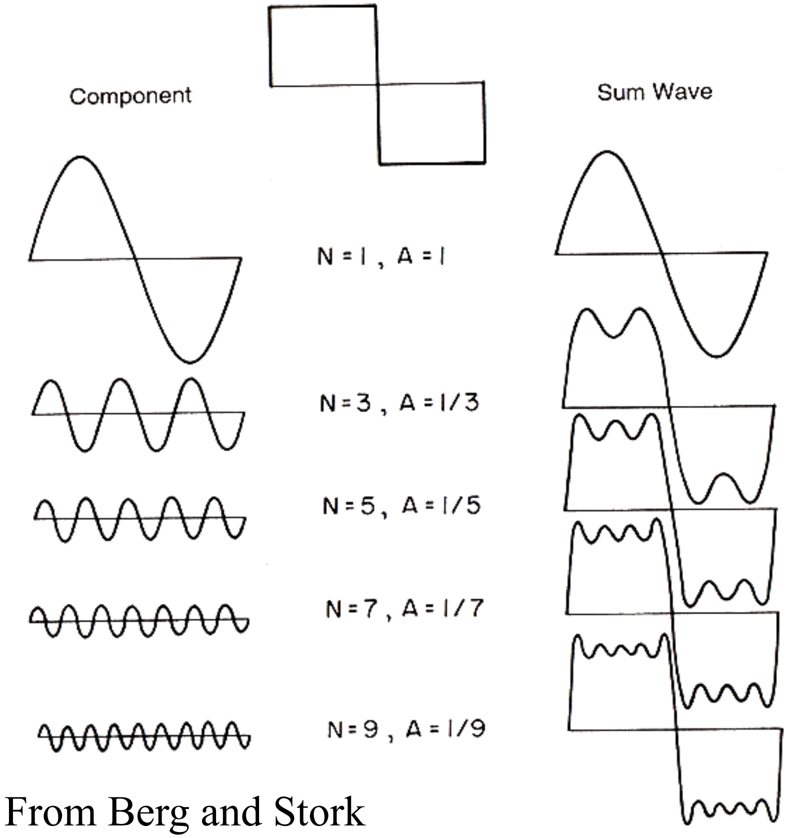



From Berg and Stork

# Weight Concentration

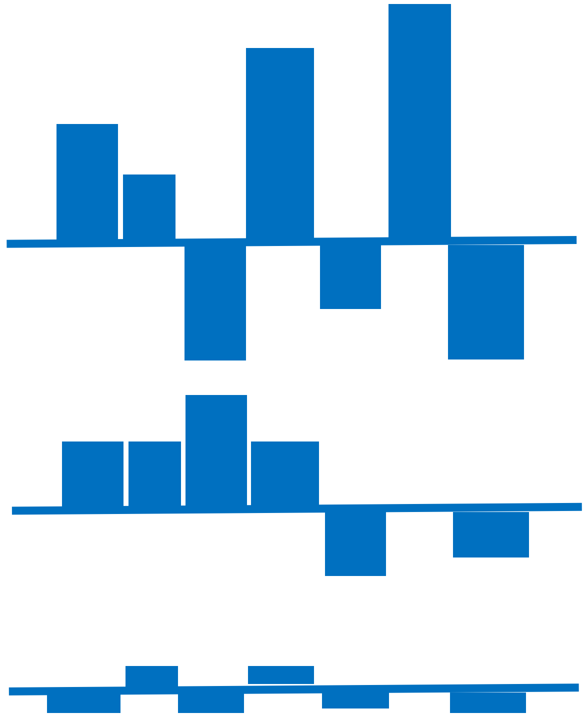
**Continuous Case**: signals decompose into components with different frequencies. High frequency components are often close to zero for a wide class of functions.

High Freq Comps are smaller



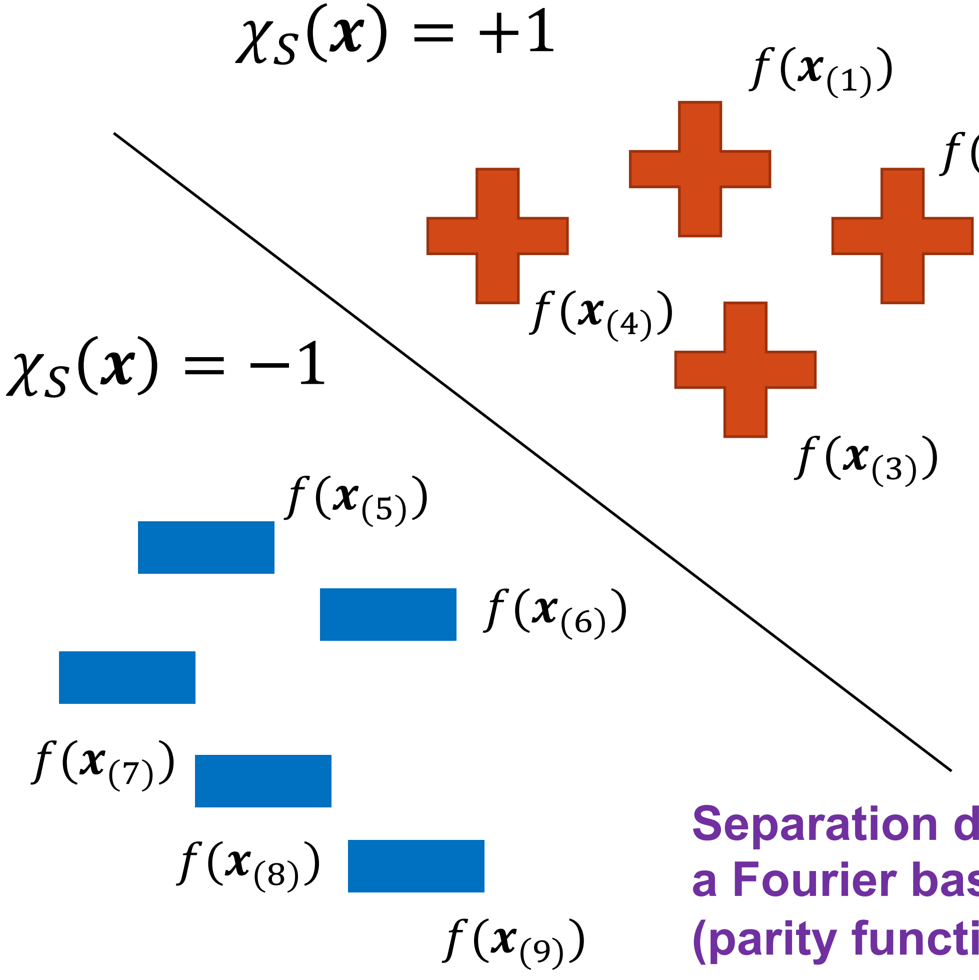

From Berg and Stork

**Discrete Case**: decompose the function into the sum of parity functions. Weights are **concentrated on low degree Fourier coefficients.**



# Weight Concentration on Lower-Order Terms

*Fourier exposes small group of variable interactions.*




**Separation defined by a Fourier basis function (parity function)**

- Let $f(\boldsymbol{x}) = \sum_{S\subseteq[n]} \widehat{f(S)}\, \chi_S(\boldsymbol{x})$ be the Fourier expansion of function $f$.

**Thm:** $\widehat{f(S)} = \mathbf{E}_{x:\{-1,1\}^n}\left(f(\boldsymbol{x})\, \chi_S(\boldsymbol{x})\right)$

$$= \frac{1}{2^n} \sum_{x:\chi_S(x)=+1} f(\boldsymbol{x}) \; - \frac{1}{2^n} \sum_{x:\chi_S(x)=-1} f(\boldsymbol{x}) .$$

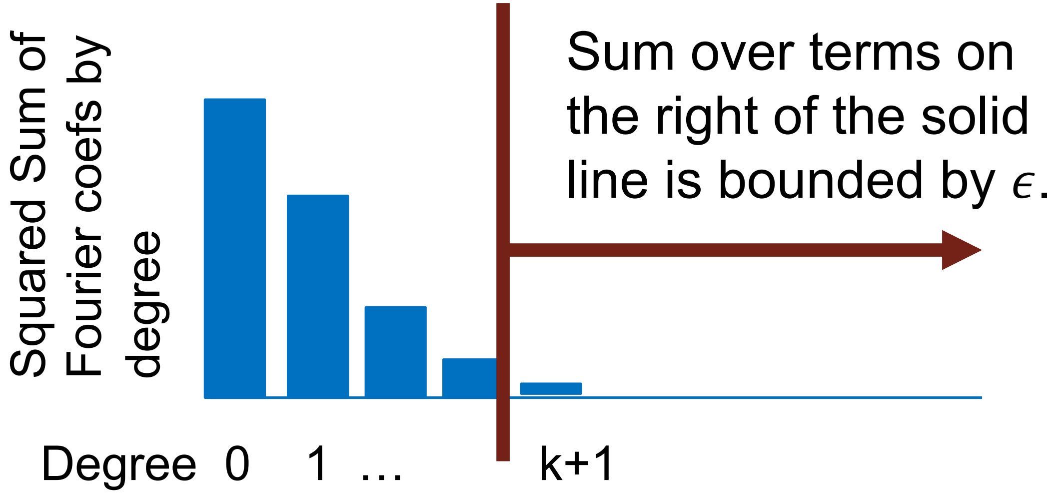
**I.e., Fourier coefficient $\widehat{f(S)}$ is the difference between the sum of the values of $f(\boldsymbol{x})$ where $\chi_S(\boldsymbol{x})$ evaluate to +1 against those where $\chi_S(\boldsymbol{x})$ evaluate to -1.**

- In most functions, high order correlations are rare. Therefore, the weight difference of the Fourier coefficients corresponding to long parity functions are usually very small.

***Long parity constraints cut space roughly evenly. Gives near zero Fourier coefficient. (Aside: Long is useful for counting! (Ermon))***

# Approximation with Bounded Degree

**Defn 1:** The Fourier spectrum of $f$ is **$\epsilon$-concentrated on degree up to $k$** if and only if the squared sum of coefficients whose degrees are larger than $k$ (tail weight) is bounded by $\epsilon$, i.e., $W_{>k}[f] = \sum_{S \subseteq [n], |S|>k} \widehat{f(S)}^2 < \epsilon$.



# Tail Bound Leads to Good Approximation

**Key property: When Fourier spectrum is $\epsilon$-concentrated, we can drop higher-order terms and get a good guaranteed bound on the accuracy of the approximation in terms of the L2 distance.**

**We will use this during probabilistic inference.**

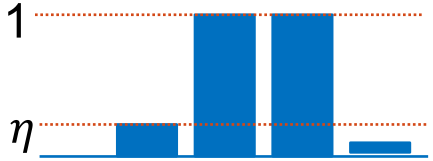
Formal: Suppose $f = \sum_{S\subseteq[n]} \widehat{f(S)}\, \chi_S(\boldsymbol{x})$ is $\epsilon$-concentrated on degree up to $k$, then $f$ can be approximated by its Fourier coefficients up to degree $k$: $f_{\leq k} = \sum_{S\subseteq[n],|S|\leq k} \widehat{f(S)}\, \chi_S(\boldsymbol{x})$, with the difference in L2 distance bounded by:

$$\mathbf{E}_{\boldsymbol{x}:\{-1,1\}^{\mathrm{n}}}\left((f - f_{\leq k})^2\right) = \sum_{S\subseteq[n],|S|>k} \widehat{f(S)}^2 < \epsilon.$$

# *Many distributions have bounded degree Fourier spectrum*

**Formally:**

**Defn :** Suppose $f(\boldsymbol{x}): \{-1,1\}^{\mathrm{n}} \rightarrow \mathbf{R}^{+}$ is a weighted function, we say $f(\boldsymbol{x})$ has **bounded width *w*** iff the number of variables in the domain of $f$ is no more than $w$. We say $f(\boldsymbol{x})$ is **contractive with gap $\mathbf{1}-\boldsymbol{\eta}$** $(0 \leq \eta < 1)$ iff (1) $f(\boldsymbol{x}) \leq 1$; (2) $\max_{\mathbf{x}} f(\boldsymbol{x}) = 1$; (3) if $f(\boldsymbol{x}_{\mathbf{0}}) < 1$, then $f(\boldsymbol{x}_{\mathbf{0}}) \leq \eta$.




**Key Thm**: *(Xue et al. 2016)* Suppose $f(\boldsymbol{x}) = \prod_{i=1}^{m} f_i(\boldsymbol{x}_{\boldsymbol{i}})$, in which every $f_i$ is a contractive function with width $w$ and gap $1-\eta$, then $f$'s Fourier spectrum is $\epsilon$-concentrated on degree up to $O\left(w \log\left(\frac{1}{\epsilon}\right) \log_{\eta} \epsilon\right)$ when $\eta > 0$ and $O\left(w \log\left(\frac{1}{\epsilon}\right)\right)$ when $\eta = 0$.

**Prove by extending *Hastad's Switching Lemma for CNF/DNF* to the weighted case. Clever use of "random restrictions."**

# Variable Elimination in the Fourier Domain

- Inference in Probabilistic Graphical Models

  Compute the probability of evidence, the partition function, etc…

$$Z = \sum_{x_1,\ldots,x_n} f(\boldsymbol{x}) = \sum_{x_1,\ldots,x_n} \prod_{i=1}^{K} \psi_i(\boldsymbol{x}_{S_i}).$$

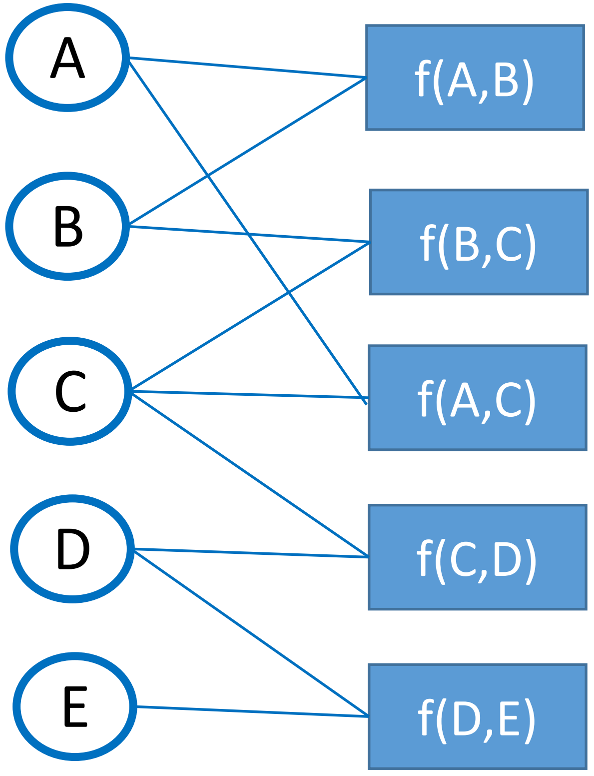
- Variable Elimination

  Eliminate (sum out) variables based on a fixed variable order.

  **Simple algorithm in Probabilistic Inference *(Inference 101)***

- Represent intermediate result in Fourier domain. Keep a fixed number of coefficients.
- With weight concentration theory, the truncated Fourier representation approximates the original probability table with small L2 loss.

# Example of the Variable Elimination process

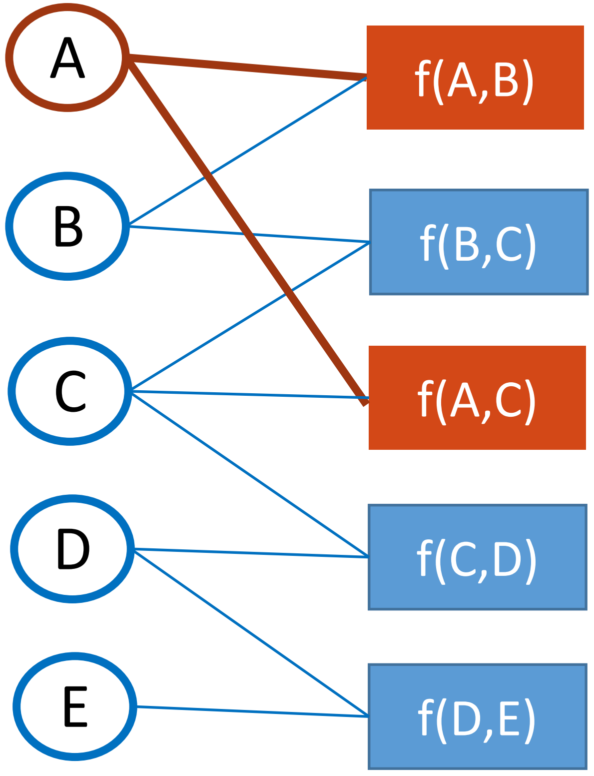



- Compute the problem:

$$\sum_{A,B,C,D,E} f(A,B)f(B,C)f(A,C)f(C,D)f(D,E)$$

- Fix a variable elimination order: $A, B, C, D, E$.

# Example of the Variable Elimination process




- Compute the problem:

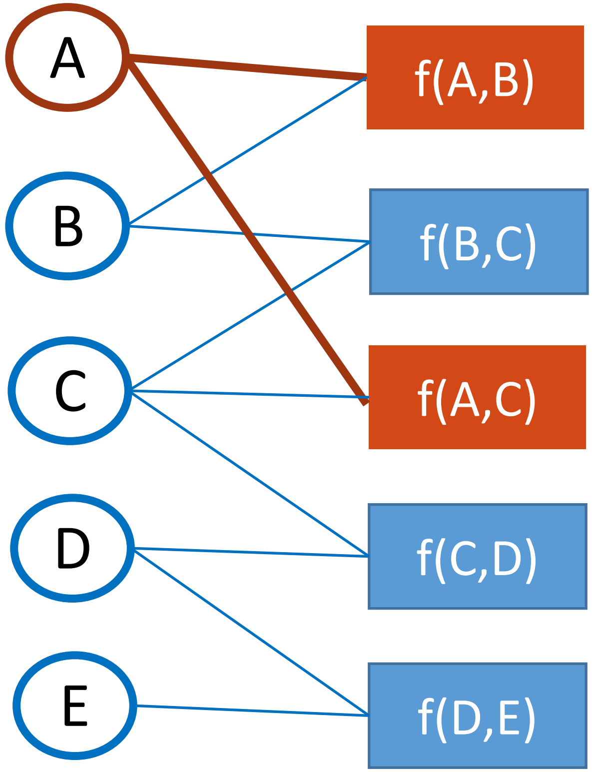
$$\sum_{A,B,C,D,E} f(A,B)f(B,C)f(A,C)f(C,D)f(D,E)$$

- Fix a variable elimination order: $A, B, C, D, E$.
- Eliminate $A$:

$$\sum_{B,C,D,E} \left( \sum_{A} f(A,B)f(A,C) \right) f(B,C)f(C,D)f(D,E)$$

$g(B,C)$

# Example of the Variable Elimination process




- Compute the problem:

$$\sum_{A,B,C,D,E} f(A,B)f(B,C)f(A,C)f(C,D)f(D,E)$$

- Fix a variable elimination order: $A, B, C, D, E$.
- Eliminate $A$:

$$\sum_{B,C,D,E} g(B,C)f(B,C)f(C,D)f(D,E)$$

- Then eliminating $B, C, D, E$.
- Involve sum and multiplication operator.

# Truncate Fourier Rep with L2 Guarantee

- Conduct variable elimination in the Fourier domain. The size of the intermediate probability tables are exponential in the treewidth.
- **When the probability table becomes too large, keep a fixed number of Fourier coefficients based on:**

  **(i) lowest degree;**

  **(ii) maximal absolute values.**
- Because of weight concentration result on the Fourier representation, **removing high degree coefficients preserves approximation guarantee in L2-distance.**

  - If the tail weights are bounded by $\epsilon$, then the average difference between the approximated function and the original one in L2-distance is also bounded by $\epsilon$.

# Results on UAI Inference Challenge

| Category | #ins | Minibucket | Fourier (max coef) | Fourier (min deg) | BP | MCMC | HAK |
|---|---|---|---|---|---|---|---|
| bn2o-30-* | 18 | 3.91 | $1.21 \cdot 10^{-2}$ | $1.36 \cdot 10^{-2}$ | $0.94 \cdot 10^{-2}$ | 0.34 | $\mathbf{8.3 \cdot 10^{-4}}$ |
| grids2/50-* | 72 | 5.12 | $\mathbf{3.67 \cdot 10^{-6}}$ | $7.81 \cdot 10^{-6}$ | $1.53 \cdot 10^{-2}$ | – | $1.53 \cdot 10^{-2}$ |
| grids2/75-* | 103 | 18.34 | $\mathbf{5.41 \cdot 10^{-4}}$ | $6.87 \cdot 10^{-4}$ | $2.94 \cdot 10^{-2}$ | – | $2.94 \cdot 10^{-2}$ |
| grids2/90-* | 105 | 26.16 | $\mathbf{2.23 \cdot 10^{-3}}$ | $5.71 \cdot 10^{-3}$ | $5.59 \cdot 10^{-2}$ | – | $5.22 \cdot 10^{-2}$ |
| blockmap_05* | 48 | $1.25 \cdot 10^{-6}$ | $\mathbf{4.34 \cdot 10^{-9}}$ | $\mathbf{4.34 \cdot 10^{-9}}$ | 0.11 | – | $8.73 \cdot 10^{-9}$ |
| students_03* | 16 | $2.85 \cdot 10^{-6}$ | $\mathbf{1.67 \cdot 10^{-7}}$ | $\mathbf{1.67 \cdot 10^{-7}}$ | 2.20 | – | $3.17 \cdot 10^{-6}$ |
| mastermind_03* | 48 | 7.83 | 0.47 | 0.36 | 27.69 | – | $\mathbf{4.35 \cdot 10^{-5}}$ |
| mastermind_04* | 32 | 12.30 | $\mathbf{3.63 \cdot 10^{-7}}$ | $\mathbf{3.63 \cdot 10^{-7}}$ | 20.59 | – | $4.03 \cdot 10^{-5}$ |
| mastermind_05* | 16 | 4.06 | $\mathbf{2.56 \cdot 10^{-7}}$ | $\mathbf{2.56 \cdot 10^{-7}}$ | 22.47 | – | $3.02 \cdot 10^{-5}$ |
| mastermind_06* | 16 | 22.34 | $\mathbf{3.89 \cdot 10^{-7}}$ | $\mathbf{3.89 \cdot 10^{-7}}$ | 17.18 | – | $4.5 \cdot 10^{-5}$ |
| mastermind_10* | 16 | 275.82 | 5.63 | 2.98 | 26.32 | – | **0.14** |

*Table 2.* The comparsion of various inference algorithms on several categories in UAI 2010 Inference Challenge. The median differences in log partition function $|\log_{10} Z_{\text{approx}} - \log_{10} Z_{\text{true}}|$ averaged over benchmarks in each category are shown. Fourier VE algorithms outperform Belief Propagation, MCMC and Minibucket Algorithm. #ins is the number of instances in each category.

- Both Fourier and Minibucket retain a message size of one million.
- Ground truth computed by Ace in 2h & 8G memory.

Both based on variable elimination, Minibucket approximate messages assuming independence.

# Results on UAI Inference Challenge

| Category | #ins | Minibucket | Fourier (max coef) | Fourier (min deg) | BP | MCMC | HAK |
|---|---|---|---|---|---|---|---|
| bn2o-30-* | 18 | 3.91 | $1.21 \cdot 10^{-2}$ | $1.36 \cdot 10^{-2}$ | $0.94 \cdot 10^{-2}$ | 0.34 | $\mathbf{8.3 \cdot 10^{-4}}$ |
| grids2/50-* | 72 | 5.12 | $\mathbf{3.67 \cdot 10^{-6}}$ | $7.81 \cdot 10^{-6}$ | $1.53 \cdot 10^{-2}$ | – | $1.53 \cdot 10^{-2}$ |
| grids2/75-* | 103 | 18.34 | $\mathbf{5.41 \cdot 10^{-4}}$ | $6.87 \cdot 10^{-4}$ | $2.94 \cdot 10^{-2}$ | – | $2.94 \cdot 10^{-2}$ |
| grids2/90-* | 105 | 26.16 | $\mathbf{2.23 \cdot 10^{-3}}$ | $5.71 \cdot 10^{-3}$ | $5.59 \cdot 10^{-2}$ | – | $5.22 \cdot 10^{-2}$ |
| blockmap_05* | 48 | $1.25 \cdot 10^{-6}$ | $\mathbf{4.34 \cdot 10^{-9}}$ | $\mathbf{4.34 \cdot 10^{-9}}$ | 0.11 | – | $8.73 \cdot 10^{-9}$ |
| students_03* | 16 | $2.85 \cdot 10^{-6}$ | $\mathbf{1.67 \cdot 10^{-7}}$ | $\mathbf{1.67 \cdot 10^{-7}}$ | 2.20 | – | $3.17 \cdot 10^{-6}$ |
| mastermind_03* | 48 | 7.83 | 0.47 | 0.36 | 27.69 | – | $\mathbf{4.35 \cdot 10^{-5}}$ |
| mastermind_04* | 32 | 12.30 | $\mathbf{3.63 \cdot 10^{-7}}$ | $\mathbf{3.63 \cdot 10^{-7}}$ | 20.59 | – | $4.03 \cdot 10^{-5}$ |
| mastermind_05* | 16 | 4.06 | $\mathbf{2.56 \cdot 10^{-7}}$ | $\mathbf{2.56 \cdot 10^{-7}}$ | 22.47 | – | $3.02 \cdot 10^{-5}$ |
| mastermind_06* | 16 | 22.34 | $\mathbf{3.89 \cdot 10^{-7}}$ | $\mathbf{3.89 \cdot 10^{-7}}$ | 17.18 | – | $4.5 \cdot 10^{-5}$ |
| mastermind_10* | 16 | 275.82 | 5.63 | 2.98 | 26.32 | – | **0.14** |

*Table 2.* The comparsion of various inference algorithms on several categories in UAI 2010 Inference Challenge. The median differences in log partition function $|\log_{10} Z\text{approx} - \log_{10} Z\text{true}|$ averaged over benchmarks in each category are shown. Fourier VE algorithms outperform Belief Propagation, MCMC and Minibucket Algorithm. #ins is the number of instances in each category.

BP is belief propagation, a variational method. MCMC is based on sampling.

# Results on UAI Inference Challenge

| Category | #ins | Minibucket | Fourier (max coef) | Fourier (min deg) | BP | MCMC | HAK |
|---|---|---|---|---|---|---|---|
| bn2o-30-* | 18 | 3.91 | $1.21 \cdot 10^{-2}$ | $1.36 \cdot 10^{-2}$ | $0.94 \cdot 10^{-2}$ | 0.34 | $\mathbf{8.3 \cdot 10^{-4}}$ |
| grids2/50-* | 72 | 5.12 | $\mathbf{3.67 \cdot 10^{-6}}$ | $7.81 \cdot 10^{-6}$ | $1.53 \cdot 10^{-2}$ | – | $1.53 \cdot 10^{-2}$ |
| grids2/75-* | 103 | 18.34 | $\mathbf{5.41 \cdot 10^{-4}}$ | $6.87 \cdot 10^{-4}$ | $2.94 \cdot 10^{-2}$ | – | $2.94 \cdot 10^{-2}$ |
| grids2/90-* | 105 | 26.16 | $\mathbf{2.23 \cdot 10^{-3}}$ | $5.71 \cdot 10^{-3}$ | $5.59 \cdot 10^{-2}$ | – | $5.22 \cdot 10^{-2}$ |
| blockmap_05* | 48 | $1.25 \cdot 10^{-6}$ | $\mathbf{4.34 \cdot 10^{-9}}$ | $\mathbf{4.34 \cdot 10^{-9}}$ | 0.11 | – | $8.73 \cdot 10^{-9}$ |
| students_03* | 16 | $2.85 \cdot 10^{-6}$ | $\mathbf{1.67 \cdot 10^{-7}}$ | $\mathbf{1.67 \cdot 10^{-7}}$ | 2.20 | – | $3.17 \cdot 10^{-6}$ |
| mastermind_03* | 48 | 7.83 | 0.47 | 0.36 | 27.69 | – | $\mathbf{4.35 \cdot 10^{-5}}$ |
| mastermind_04* | 32 | 12.30 | $\mathbf{3.63 \cdot 10^{-7}}$ | $\mathbf{3.63 \cdot 10^{-7}}$ | 20.59 | – | $4.03 \cdot 10^{-5}$ |
| mastermind_05* | 16 | 4.06 | $\mathbf{2.56 \cdot 10^{-7}}$ | $\mathbf{2.56 \cdot 10^{-7}}$ | 22.47 | – | $3.02 \cdot 10^{-5}$ |
| mastermind_06* | 16 | 22.34 | $\mathbf{3.89 \cdot 10^{-7}}$ | $\mathbf{3.89 \cdot 10^{-7}}$ | 17.18 | – | $4.5 \cdot 10^{-5}$ |
| mastermind_10* | 16 | 275.82 | 5.63 | 2.98 | 26.32 | – | **0.14** |

*Table 2.* The comparsion of various inference algorithms on several categories in UAI 2010 Inference Challenge. The median differences in log partition function $|\log_{10} Z\text{approx} - \log_{10} Z_{\text{true}}|$ averaged over benchmarks in each category are shown. Fourier VE algorithms outperform Belief Propagation, MCMC and Minibucket Algorithm. #ins is the number of instances in each category.

HAK is an award winning solver, based on a portfolio of several state-of-the-art solvers.

# Conclusion

- We explore **the Fourier Representation,** a **novel compact representation** of high-dimensional probability distributions.

- The Fourier Representation provides **a natural and well motivated way of approximating probability distributions.**
  - Most probability mass **concentrates on low degree coefficients**.
  - Drop high degree coefficients leads to **good approximation guarantee in L2-norm**, novel bound contrasting with existing methods such as minibucket and variational approaches.

- Strong empirical results when applied to probabilistic inference.
  - **Orders of magnitudes better** than competing methods.