Approximate Counting CSPs – Hunt for Galois Connection

Andrei A. Bulatov
Simon Fraser University

La Trobe University 2013

Constraint Satisfaction Problem

- Decision:
  Given a conjunctive formula
  $$R_1(x,y) \wedge R_2(z,x,x) \wedge \ldots$$
  decide if it is satisfiable

- Counting:
  Given a conjunctive formula
  $$R_1(x,y) \wedge R_2(z,x,x) \wedge \ldots$$
  find the number of satisfying assignments

- If all relations are from set Γ, the problem is denoted
  CSP(Γ), **#CSP(Γ)**

Examples: #SAT, Linear Equations

#3-SAT: = **#CSP(** $\mathcal{C}_3$ **)**

Instance: A propositional formula $\Phi = C_1 \wedge \ldots \wedge C_n$ in 3-CNF.

Objective: How many satisfying assignments are there?

#Linear Equations: = **#CSP(** F **)**

Instance: A system of linear equations

$$\begin{cases} a_{11}x_1 + \ldots + a_{1m}x_m = b_1 \\ \ldots \\ a_{n1}x_1 + \ldots + a_{nm}x_m = b_n \end{cases}$$

Objective: How many solutions are there?

CSP and Galois Connection

For a relation R an n-ary operation f on the same set is a polymorphism of R if for any $\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_n \in R$

$$f(\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_n) \in R$$

f is a polymorphism of Γ if it is a polymorphism of every relation from Γ

Theorem (Jeavons; 1998)
Let Γ and Δ be constraint languages on a set D such that every polymorphism of Γ is also polymorphism of Δ. Then **CSP(Δ)** is poly time reducible to **CSP(Γ)**.

More CSPs, More Galois Connections

CSP	usual polymorphisms
QCSP	surjective operations
CSPs with global const.	usual polymorphisms (almost)
Counting	usual polymorphisms
Optimization	multi-morphisms –
	fractional polymorphisms 1 –
	weighted polymorphisms

More CSPs, More Galois Connections

- Approximate optimization

  fractional polymorphisms 2

- QCSP + disjunction — surjective hyperoperations
- Approximate counting — ????

Outline

- How Galois connection works
- Galois connection for counting
- Partition functions and `Galois connection’

How It Works

- Suppose we have a CSP $R_1(z,x,x) \wedge R_2(x,y) \wedge \ldots$
  First rewrite it as follows:
  Is the sentence $\exists x,y,z \ldots R_1(z,x,x) \wedge R_2(x,y) \wedge \ldots$ true?
- Now suppose that R_2 has a `definition'

$$R_2(x,y) = \exists t\, Q(x,t) \wedge Q(t,y)$$

- Then we can substitute

$$\exists x,y,z \ldots R_1(z,x,x) \wedge (\exists t\, Q(x,t) \wedge Q(t,y)) \wedge \ldots$$

  and transform

$$\exists x,y,z,t \ldots R_1(z,x,x) \wedge Q(x,t) \wedge Q(t,y) \wedge \ldots$$

- Therefore $CSP(\{R_1, R_2, \ldots\}) \leq CSP(\{R_1, Q, R_3, \ldots\})$

How It Works 2

- Relation R is pp-definable in Γ if $R(\vec{x}) = \exists \vec{y}\, \Phi(\vec{x}, \vec{y})$, where Φ is a conjunctive formula using relations from Γ
- Δ is pp-definable in Γ if every its relation is.

Theorem (Jeavons; 1998)
Let Γ and Δ be constraint languages on a set D such that Δ is pp-definable in Γ. Then **CSP(Δ)** is poly time reducible to **CSP(Γ)**.

How It Works 3

Galois Correspondence (Geiger, 1968; BKKR, 1970; …)
Let Γ and Δ be constraint languages on a set D. Then Δ is pp-definable in Γ if and only if every polymorphism of Γ is also a polymorphism of Δ.

More Galois Connections

- conjunctive formulas (no quantifiers) — partial operations; works for any kind of CSP
- $\forall\exists\wedge$-formulas — surjective operations; works for QCSP
- $\exists\wedge\vee$-formulas — unary functions
- More complicated for fractional polymorphisms, multi-morphisms, etc.

G.C. for Counting CSPs

- Suppose we have a #CSP $R_1(z,x,x) \wedge R_2(x,y) \wedge \ldots$
  We cannot rewrite as a sentence
  $$\exists x,y,z \ldots R_1(z,x,x) \wedge R_2(x,y) \wedge \ldots$$
- Suppose that R_2 has a `definition'
  $$R_2(x,y) = \exists t\, Q(x,t) \wedge Q(t,y)$$
- Then we substitute
  $$R_1(z,x,x) \wedge (\exists t\, Q(x,t) \wedge Q(t,y)) \wedge \ldots$$
  and transform
  $$\exists t \ldots R_1(z,x,x) \wedge Q(x,t) \wedge Q(t,y) \wedge \ldots$$
- This is not a counting CSP

G.C. for Counting CSPs 2

Theorem (B.,Dalmau; 2003)
Let Γ and Δ be constraint languages on a set D such that Δ is pp-definable in Γ. Then **#CSP(Δ)** is poly time reducible to **#CSP(Γ).**

Reduction is through interpolation

Approximate Counting

Algorithm Alg is an *ε-approximation* algorithm for a counting CSP if it is polynomial time and for any instance P of the problem it outputs number Alg(P) such that

$$\frac{|\#P - \mathrm{Alg}(P)|}{\#P} < \varepsilon$$

Alg is said to be an *FPRAS* if take P and ε as input and returns number Alg(P) satisfying the condition above in time polynomial in |P| and $1/\varepsilon$

AP-reductions are reductions that preserve approximation

AP-Reductions

Definitions by conjunctive formulas are AP-reductions
If R_2 has a definition $R(x, y, t) = Q(x, t) \wedge Q(t, y)$, then
$CSP(\{R\}) \leq CSP(\{Q\})$

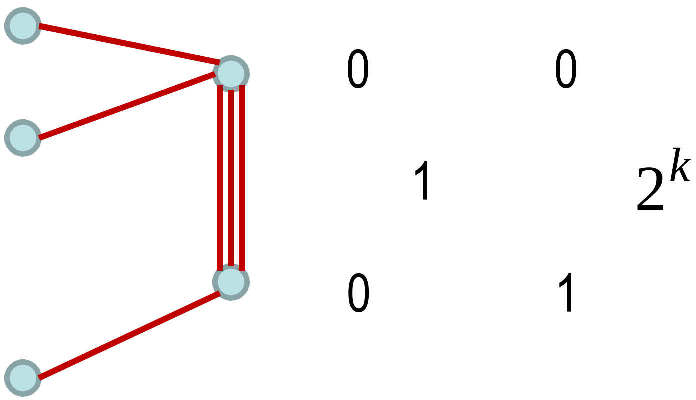
However, interpolation is not an NP-reduction
Moreover, the usual Galois connection does not work for AP-reductions

Example

Let R be the following ternary relation

$$R = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \quad \text{and} \quad Q = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

I claim $CSP(\{Q\}) \leq_{AP} CSP(\{R\})$



Algorithm:

- Insert k R's for each S
- Solve the problem
- Divide by $(2^k)^s$ and round down

Max-Quantifiers and AP-Reductions

Let $\Phi(x_1,\dots,x_n,y_1,\dots,y_m)$ be a conjunctive formula in language Γ, and for an assignment φ to $x_1,\dots,x_n$ let $\#\varphi$ denote the number of extensions of φ that satisfy Φ.

Let M be the maximum of these values.

We say that Φ is a *max-implementation* of R if

$$\varphi(\vec{x}) \in R \Leftrightarrow \#\varphi = M$$

Theorem (B., Hedayaty, 2011)
If there is a max-implementation of R in Γ, then **#CSP($\Gamma \cup$ {R})** is AP-reducible to **#CSP(Γ)**

Max-Co-Clones

We write $R(x_1,\ldots,x_n)=\exists_{\max} y_1,\ldots,y_m \Phi(x_1,\ldots,x_n,y_1,\ldots,y_m)$

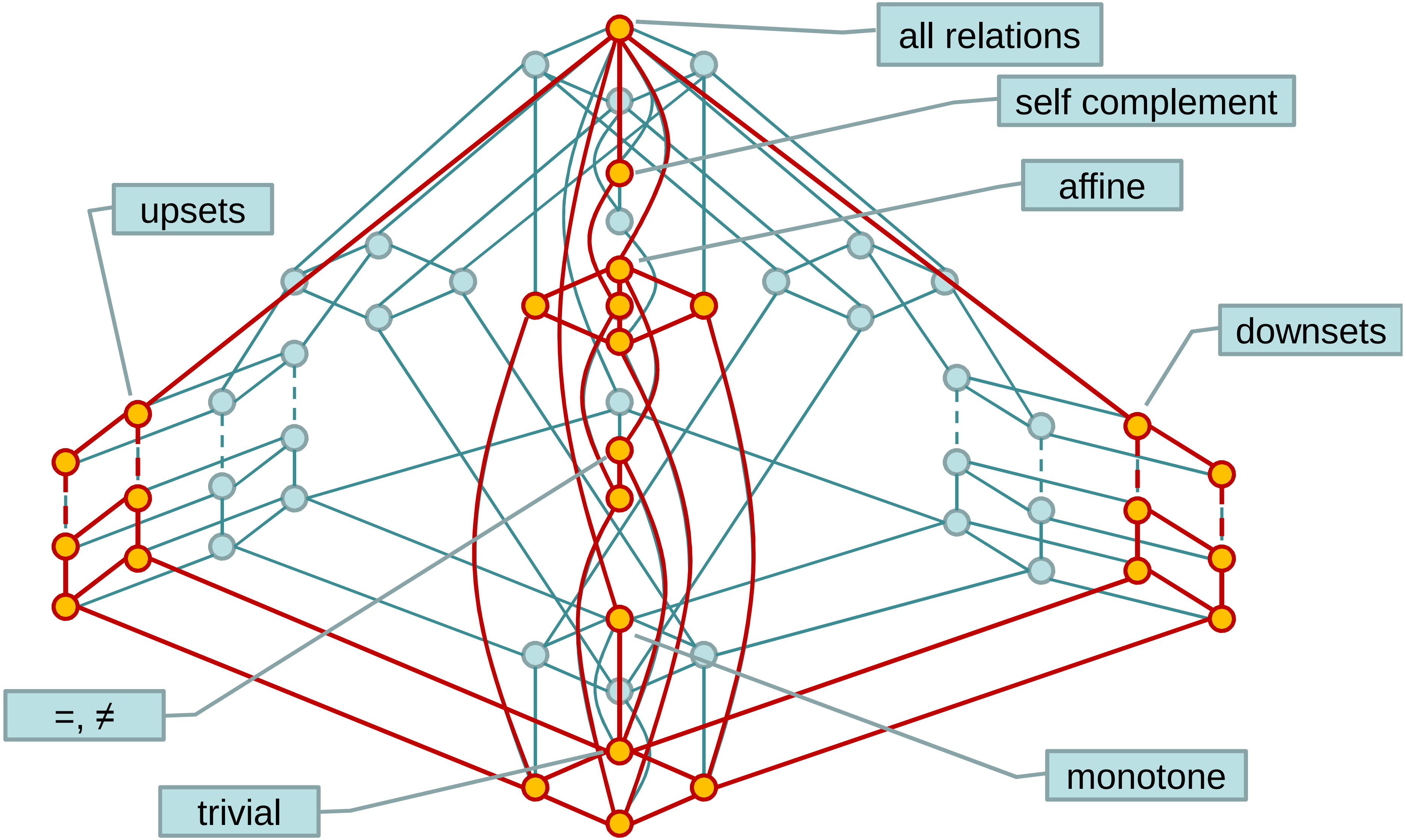
The max-co-clone $\langle\Gamma\rangle_{max}$ of a constraint language Γ consists of all relations that can be expressed using:

- Relations in Γ
- The equality relation
- Conjunction
- Max-quantification

Problem:

Find a Galois correspondence for max-co-clones

Co-Clones


all relations
self complement
affine
upsets
downsets
=, ≠
trivial
monotone

Boolean Approximate Counting

Theorem (Dyer,Goldberg,Jerrum, 2007)
Let Γ be a constraint language over {0,1}. Then either **#CSP(Γ)** is solvable in polynomial time; or it is as hard to approximate as **#BIS**; or it is as hard to approximate as **#SAT**.

G.C. for Max-Quantification

Problem:

Find a Galois correspondence for max-co-clones

In the Boolean case every max-co-clone is a regular co-clone

It is not the case for larger sets

Every max-co-clone is a partial co-clone

Therefore G.C. should be with some class of partial functions

What class??

What We Would Like To Separate with G.C.

Bipartite #k-Coloring :

Input. A bipartite graph

Objective. Find the number of its k-colorings

It is believed that the hierarchy

FPRAS $\leq_{AP}$ Bip-3-Col $\leq_{AP}$ Bip-5-Col $\leq_{AP}$ Bip-7-Col $\leq_{AP} \ldots$

is strict

k-Existential Quantifiers

If Φ is a formula with free variables $x_1,\ldots,x_n$ and y, a tuple $a_1,\ldots,a_n$ satisfies

$$\Psi(x_1,\ldots,x_n) = \exists_k y \Phi(x_1,\ldots,x_n,y)$$

iff $\Phi(a_1,\ldots,a_n,b)$ is true for at least k values b.

$\exists_1 = \exists$

If |D| = k then $\exists_k = \forall$

The k-existential co-clone $\langle\Gamma\rangle_k$, k $\in \mathbb{N}$ of Γ consists of all relations definable from Γ and equality by conjunction and k-existential quantifier

K-Surjective Operations

A (partial) operation $f : D^n \to D$ is said to be *k-surjective* if for any $A_1, \ldots, A_n \subseteq D$ with $|A_1| = \ldots = |A_n| = k$ the set $f(A_1, \ldots, A_n)$ contains at least k elements

For a constraint language Γ and a set $k \in \mathbb{N}$, the set of all k-surjective polymorphisms of Γ is denoted by m(k)-Pol(Γ)

Theorem (B.,Hedayaty; 2012)

For any constraint language Γ, for any $k \in \mathbb{N}$,

$$Inv(m(k) - Pol(\Gamma)) = \langle \Gamma \rangle_k$$

Weighted #CSP

Γ is a set of functions $f\colon D \to \mathbb{R}$ (natural, real, complex)

Instance of #CSP(Γ):

$$R_1(x,y) \wedge R_2(z,x,x) \wedge \ldots \qquad \Rightarrow \quad f_1(x,y) \cdot f_2(z,x,x) \cdot \ldots$$

Given an instance I the weight of a mapping $\sigma\colon V \to D$ is computed as

$$w(\sigma) = f_1(\sigma(x),\sigma(y)) \cdot f_2(\sigma(z),\sigma(x),\sigma(x)) \cdot \ldots$$

Then

$$Z(I) = \sum_{\sigma:V \to D} w(\sigma)$$

Counting Problems: Fourier Coefficients

Let $f:\{0,1\}^n \to \mathbb{R}^+$ be a function and

$$S=\{i_1,\ldots,i_k\}\subseteq\{1,\ldots,n\}$$

Fourier coefficient $\hat{f}(S)$ is given by

$$\hat{f}(S)=\frac{1}{2^n}\sum_{x_1,\ldots,x_n\in\{0,1\}^n} f(x_1,\ldots,x_n)(-1)^{x_{i_1}+\cdots+x_{i_k}}$$

ω-Clones

We can define ω-clones similar to co-clones:

- $\wedge$ becomes product
- $\exists$ becomes summation
- also add a `trivial' operation of multiplying by a constant

$\langle \Gamma \rangle$ denotes the set of all functions that can be obtained from Γ applying the operations above

Limits: $R = \lim_{i \to \infty} R_i$ for $R_1, R_2, \ldots \in \langle \Gamma \rangle$

Functional Clones 2

A set of functions closed under multiplication by a constant, products, summation, and limits is said to be an ω-clone

The ω-clone generated by a set of functions Γ is denoted $\langle\Gamma\rangle_\omega$

Theorem (B.,Dyer,Goldberg,Jerrum; 2011)
If $\Gamma' \subseteq \langle\Gamma\rangle_\omega$ is finite then #CSP(Γ') $\leq_{AP}$#CSP(Γ)

Log Supermodular Functions

Question 2: Any `morphisms' for ω-clones?

A function $f : \{0,1\}^n \rightarrow \mathbb{R}^+$ is said to be log supermodular if for any $\vec{x}, \vec{y} \in \{0,1\}^n$

$$f(\vec{x}) \cdot f(\vec{y}) \leq f(\vec{x} \wedge \vec{y}) \cdot f(\vec{x} \vee \vec{y})$$

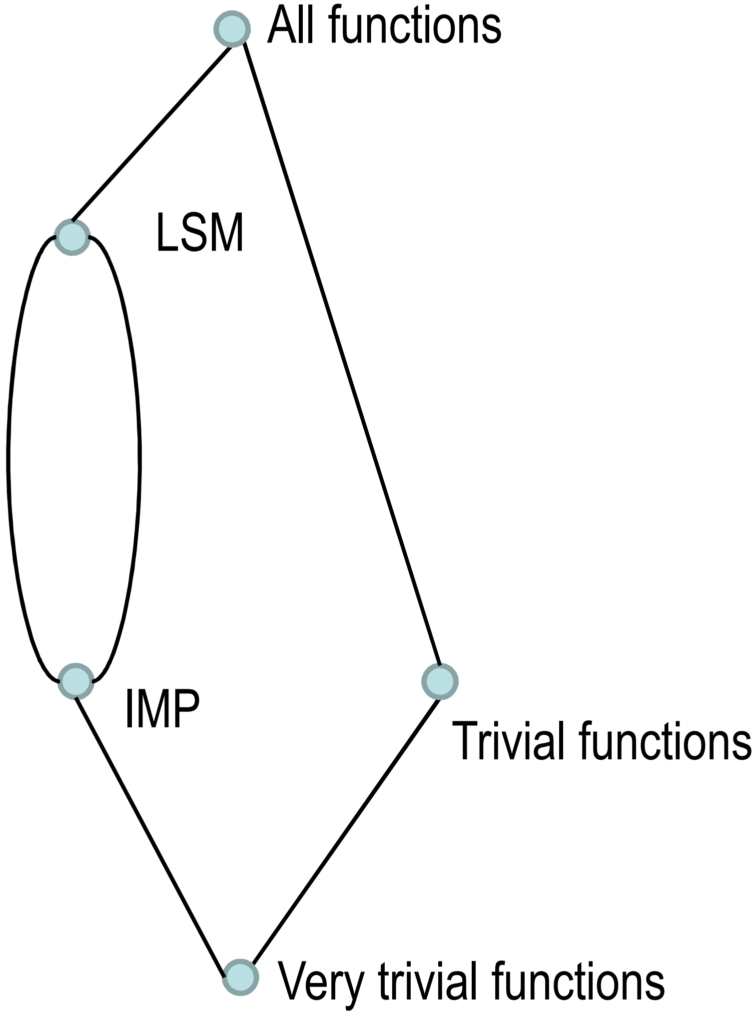
Lemma

LSM is an ω-clone

Log Supermodular Functions 2

Theorem

Let Γ be a set of functions $f:\{0,1\}^n \to \mathbb{R}^+$ containing all nonnegative unary functions. Then either $\langle\Gamma\rangle_\omega$ is trivial, or it is LSM, or it is the set of all functions on {0,1}



Multi Morphisms

A k-tuple of k-ary operations $(g_1,\ldots,g_k),\ g_i:\{0,1\}^k \to \{0,1\},$ is said to be a multimorphism of function f if for any $x_{11},\ldots,x_{1n},\ldots,x_{kn} \in \{0,1\}$

$$f(x_{11},\ldots,x_{1n})\cdot\ldots\cdot f(x_{k1},\ldots,x_{kn})$$

$$\le f(g_1(x_{11},\ldots,x_{k1}),\ldots,g_1(x_{1n},\ldots,x_{kn}))\times\ldots$$

$$\times f(g_1(x_{11},\ldots,x_{k1}),\ldots,g_1(x_{1n},\ldots,x_{kn}))$$

Log supermodularity is equivalent to $(\wedge,\vee)$ multimorphism

Do multimorphisms preserve products, summation, and limit?

Multi Morphisms 2

No example of a multimorphism that does not
Threshold multimorphism $(g_1,\ldots,g_k)$:

$g_i(x_1,\ldots,x_k)=1$ iff at least i arguments equal 1

Preserves products, summation, limit, because
f has a threshold multimorphism iff it is log supermodular

No direct proof

Fourier Coefficients Revisited

Let $f : \{0,1\}^n \to \mathbb{R}$ be a function and

$$S = \{i_1, \ldots, i_k\} \subseteq \{1, \ldots, n\}$$

Fourier coefficient $\hat{f}(S)$ is given by

$$\hat{f}(S) = \frac{1}{2^n} \sum_{x_1,\ldots,x_n \in \{0,1\}^n} f(x_1, \ldots, x_n) \cdot (-1)^{x_{i_1} + \cdots + x_{i_k}}$$

Let $\vec{y}_S$ denote tuple with $y_i = 1$ if i $\in$ S and $y_i = 0$ otherwise

Fourier transform of f is function $\hat{f}$ given by $\hat{f}(\vec{y}_S) = \hat{f}(S)$

Fourier Coefficients vs Products and Summation

Summation: Let $f(x_1,\ldots,x_n,y)$ be a function and

$$g(x_1,\ldots,x_n) = f(x_1,\ldots,x_n,0) + f(x_1,\ldots,x_n,1)$$

Then $\hat{g}(\vec{x}) = \hat{f}(\vec{x},0)$

Product:

Convolution Theorem. $\widehat{fh}(\vec{x}) = \sum_{\vec{w}\in\{0,1\}^n} \hat{f}(\vec{w}) \cdot \hat{g}(\vec{x} + \vec{w})$

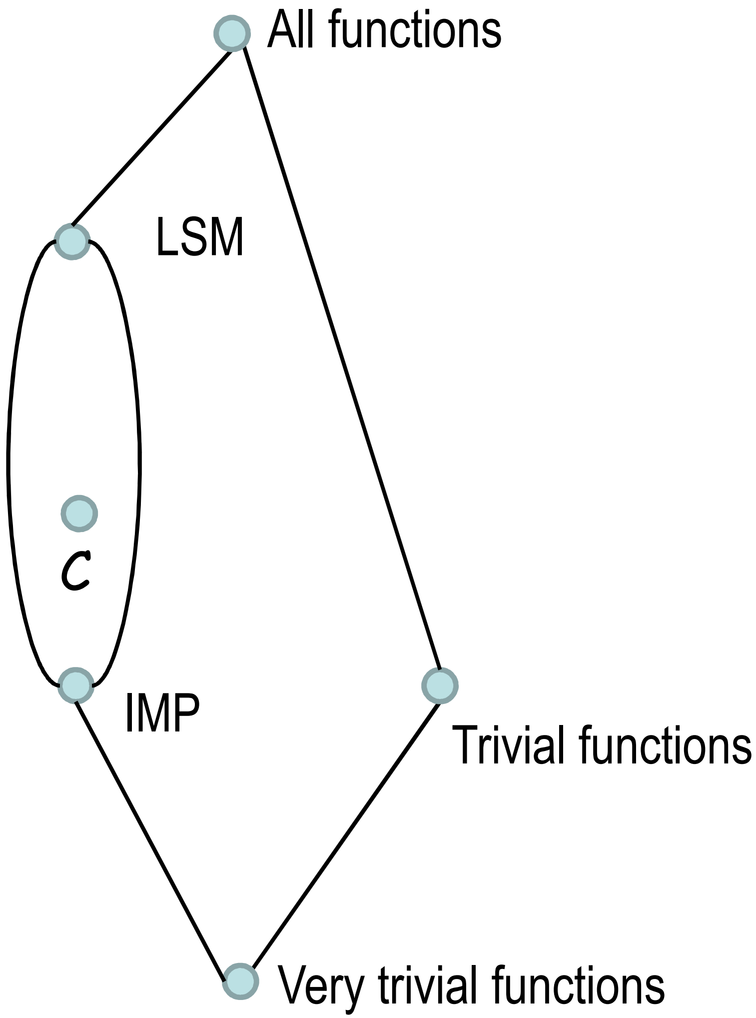
ω-Clones and Fourier Transform

Let $\bar{f}(\vec{x}) = f(\neg\vec{x})$ and

$$f^{*}(\vec{x}) = f(\vec{x})\,\bar{f}(\vec{x})$$

Let C be the set of functions f such that $\widehat{f^{*}}$ is nonnegative

Theorem (McQuillan,Dyer; 2011)
C is a ω-clone



Thank you!