Piecewise linear homogeneous valued constraint satisfaction problems

Caterina Viola
joint work with Manuel Bodirsky and Marcello Mamino

TU Dresden
Institute for Algebra

Logic Colloquium 2018 - Udine
July 23-28, 2018

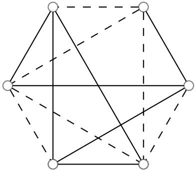
Example of (infinite domain) VCSP: Min Correlation Clustering

You are organising a big dinner party for all your facebook friends, V.

You can use an arbitrary number of tables each labelled with a color.

Friends: two people that are friends on facebook (solid);

Unfollowed: two people that unfollowed each other on facebook (dashed).



Example of (infinite domain) VCSP: Min Correlation Clustering

You are organising a big dinner party for all your facebook friends, V.

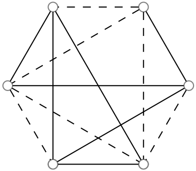
You can use an arbitrary number of tables each labelled with a color.

Friends: two people that are friends on facebook (solid);

Unfollowed: two people that unfollowed each other on facebook (dashed).

Task: assign a table to each person in such a way to minimise the sum of

- the number of friends that are not at the same table;
- the number of unfollowed that are at the same table.



Example of (infinite domain) VCSP: MIN CORRELATION CLUSTERING

You are organising a big dinner party for all your facebook friends, V.

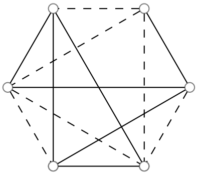
You can use an arbitrary number of tables each labelled with a color.

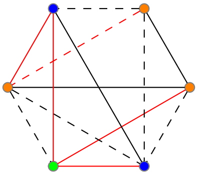
Friends: two people that are friends on facebook (solid);

Unfollowed: two people that unfollowed each other on facebook (dashed).

Task: assign a table to each person in such a way to minimise the sum of

- the number of friends that are not at the same table;
- the number of unfollowed that are at the same table.





Cost of this assignment=4

Example of (infinite domain) VCSP: Min Correlation Clustering

You are organising a big dinner party for all your facebook friends, V.

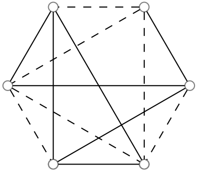
You can use an arbitrary number of tables each labelled with a color.

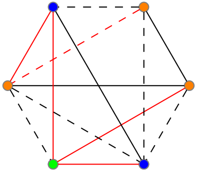
Friends: two people that are friends on facebook (solid);

Unfollowed: two people that unfollowed each other on facebook (dashed).

Task: assign a table to each person in such a way to minimise the sum of

- the number of friends that are not at the same table;
- the number of unfollowed that are at the same table.





Cost of this assignment=4

Min Correlation Clustering is NP-hard [Bansal - Blum - Chawla, 2004].

Valued constraint languages

Let D be a fixed set (called domain).

A cost function over D is a function $f\colon D^n \to \mathbb{Q} \cup \{+\infty\}$, for $n \in \mathbb{N}$.

Valued constraint languages

Let D be a fixed set (called domain).

A cost function over D is a function $f\colon D^n \to \mathbb{Q} \cup \{+\infty\}$, for $n \in \mathbb{N}$.
Here $f(x) = +\infty$ is interpreted as: f not defined on x.

Valued constraint languages

Let D be a fixed set (called domain).

A cost function over D is a function $f\colon D^n \to \mathbb{Q} \cup \{+\infty\}$, for $n \in \mathbb{N}$.
Here $f(x) = +\infty$ is interpreted as: f not defined on x.

Valued (constraint) language: a set Γ of cost functions over D.

Valued constraint languages

Let D be a fixed set (called domain).

A cost function over D is a function $f\colon D^n \to \mathbb{Q} \cup \{+\infty\}$, for $n \in \mathbb{N}$.
Here $f(x) = +\infty$ is interpreted as: f not defined on x.

Valued (constraint) language: a set Γ of cost functions over D.

Example (Valued language for Min Correlation Clustering)

$\Gamma = \{f_1, f_2\}$ where $f_1, f_2 \colon \mathbb{Q}^2 \to \{0, 1\}$ and

$$f_1(x,y) = \begin{cases} 0 & x = y \\ 1 & x \neq y \end{cases} \qquad f_2(x,y) = \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases}$$

Instances of VCSPs

Definition

An instance I of the valued constraint satisfaction problem for Γ, $VCSP(\Gamma)$, consists of

- a finite set of variables V,
- an expression ϕ of the form

$$\sum_{i=1}^{m} f_i(x_1^i, \ldots, x_{ar(f_i)}^i)$$

where $f_1, \ldots, f_m \in \Gamma$ and all the x_j^i are variables from V, and

- a value $u \in \mathbb{Q} \cup \{+\infty\}$.

Instances of VCSPs

Definition

An instance I of the valued constraint satisfaction problem for Γ, $VCSP(\Gamma)$, consists of

- a finite set of variables V,
- an expression ϕ of the form

$$\sum_{i=1}^{m} f_i(x_1^i, \ldots, x_{ar(f_i)}^i)$$

  where $f_1, \ldots, f_m \in \Gamma$ and all the x_j^i are variables from V, and
- a value $u \in \mathbb{Q} \cup \{+\infty\}$.

Task: decide whether there exists a map $\alpha \colon V \to D$ whose cost

$$\sum_{i=1}^{m} f_i^{\Gamma}(\alpha(x_1^i), \ldots, \alpha(x_{ar(f_i)}^i))$$

is finite, and if so, whether there is one whose cost is smaller than or equal to u.

Finite domain VCSPs

Theorem (Kolmogorov-Krokhin-Rolinek, 2015 + Bulatov-Zhuk, 2017)

The VCSP for a finite set of cost functions over a finite domain is either polynomial-time solvable or NP-hard.

Finite domain VCSPs

Theorem (Kolmogorov-Krokhin-Rolinek, 2015 + Bulatov-Zhuk, 2017)

The VCSP for a finite set of cost functions over a finite domain is either polynomial-time solvable or NP-hard.

This talk is about infinite domain VCSPs.

e.g.:

- MIN CORRELATION CLUSTERING,
- LINEAR PROGRAMMING.

.

Finite domain VCSPs

Theorem (Kolmogorov-Krokhin-Rolinek, 2015 + Bulatov-Zhuk, 2017)

The VCSP for a finite set of cost functions over a finite domain is either polynomial-time solvable or NP-hard.

This talk is about infinite domain VCSPs.

e.g.:

- Min Correlation Clustering,
- Linear Programming.

.

We will focus on a very natural class of infinite domain VCSPs.

A class of infinite domain VCSPs

A function is piecewise linear homogeneous (PLH) if it is first-order definable over $(\mathbb{Q}; <, 1, (c\cdot)_{c\in\mathbb{Q}})$, where $c\cdot$ denotes the map $x \mapsto cx$.

A class of infinite domain VCSPs

A function is piecewise linear homogeneous (PLH) if it is first-order definable over $(\mathbb{Q}; <, 1, (c\cdot)_{c \in \mathbb{Q}})$, where $c\cdot$ denotes the map $x \mapsto cx$.

A language Γ is PLH if every member of Γ is a PLH cost function.

A class of infinite domain VCSPs

A function is piecewise linear homogeneous (PLH) if it is first-order definable over $(\mathbb{Q}; <, 1, (c\cdot)_{c \in \mathbb{Q}})$, where $c\cdot$ denotes the map $x \mapsto cx$.

A language Γ is PLH if every member of Γ is a PLH cost function.

Example (Binary PLH function)

$$f(x,y) = \begin{cases} 2y & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \wedge y \geq 3 \\ -3y & \text{if } \underbrace{x \leq 0 \wedge y < 3}_{\text{“case distinctions”}}. \end{cases}$$

A class of infinite domain VCSPs

A function is piecewise linear homogeneous (PLH) if it is first-order definable over $(\mathbb{Q}; <, 1, (c\cdot)_{c\in\mathbb{Q}})$, where $c\cdot$ denotes the map $x \mapsto cx$.

A language Γ is PLH if every member of Γ is a PLH cost function.

Example (Binary PLH function)

$$f(x,y) = \begin{cases} 2y & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \wedge y \geq 3 \\ -3y & \text{if } \underbrace{x \leq 0 \wedge y < 3}_{\text{"case distinctions"}}. \end{cases}$$

Observation: every finite domain VCSP is equivalent to a PLH VCSP.

Representing PLH cost functions

Lemma (Bodirsky-Mamino-V.,2018)

$(\mathbb{Q}; <, 1, (c\cdot)_{c\in\mathbb{Q}})$ *admits quantifier elimination.*

Consequence: every PLH cost function can be written in the form

$$f(x,y) = \begin{cases} \alpha_1 x + \beta_1 y & \chi_1(x,y) \\ \vdots & \vdots \\ \alpha_l x + \beta_l y & \chi_l(x,y) \end{cases}$$

where $\alpha_i \cdot \beta_i = 0$,

Representing PLH cost functions

Lemma (Bodirsky-Mamino-V.,2018)

$(\mathbb{Q}; <, 1, (c\cdot)_{c\in\mathbb{Q}})$ *admits quantifier elimination.*

Consequence: every PLH cost function can be written in the form

$$f(x,y) = \begin{cases} \alpha_1 x + \beta_1 y & \chi_1(x,y) \\ \vdots & \vdots \\ \alpha_l x + \beta_l y & \chi_l(x,y) \end{cases}$$

where $\alpha_i \cdot \beta_i = 0$, and $\chi_i(x,y)$ is a conjunction of atomic formulas, for all i.

Representing PLH cost functions

Lemma (Bodirsky-Mamino-V.,2018)

$(\mathbb{Q}; <, 1, (c\cdot)_{c\in\mathbb{Q}})$ *admits quantifier elimination.*

Consequence: every PLH cost function can be written in the form

$$f(x,y) = \begin{cases} \alpha_1 x + \beta_1 y & \chi_1(x,y) \\ \vdots & \vdots \\ \alpha_l x + \beta_l y & \chi_l(x,y) \end{cases}$$

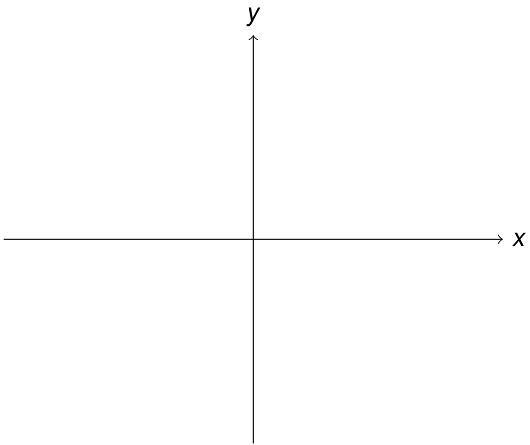
where $\alpha_i \cdot \beta_i = 0$, and $\chi_i(x,y)$ is a conjunction of atomic formulas, for all i.

Every atomic formula is equivalent to either

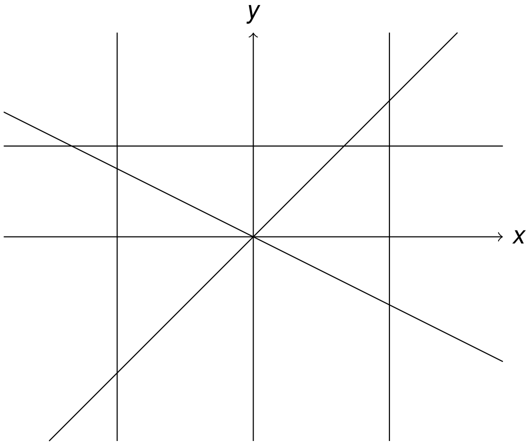
- $c_1 \cdot x \overset{<}{=} c_2 \cdot y$, or
- $c_1 \cdot x \overset{<}{=} c_2 \cdot 1$, or
- $c_1 \cdot 1 \overset{<}{=} c_2 \cdot y$,

for some $c_i \in \mathbb{Q}$.

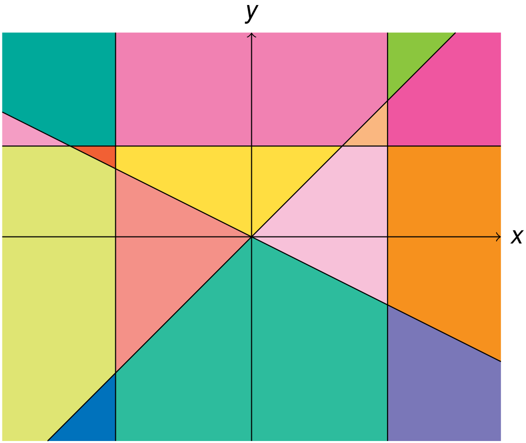
From a PLH VCSP to a finite domain VCSP: sampling



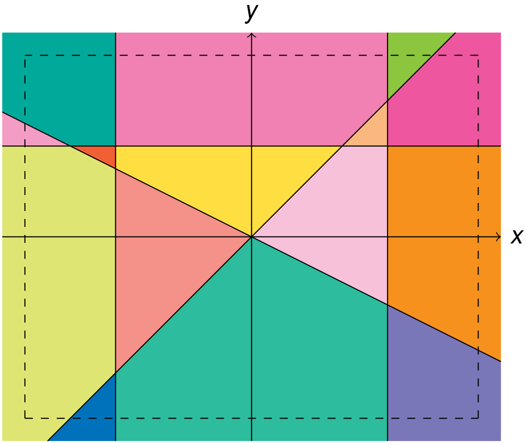
From a PLH VCSP to a finite domain VCSP: sampling



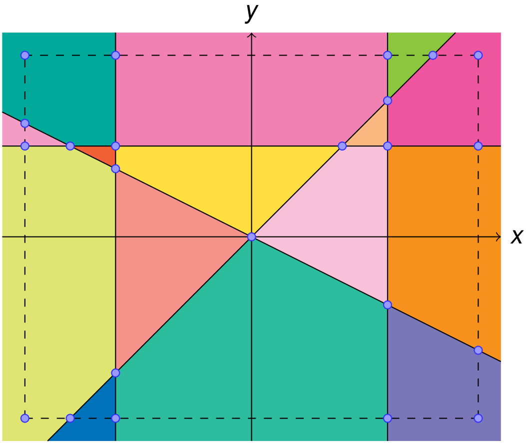
From a PLH VCSP to a finite domain VCSP: sampling



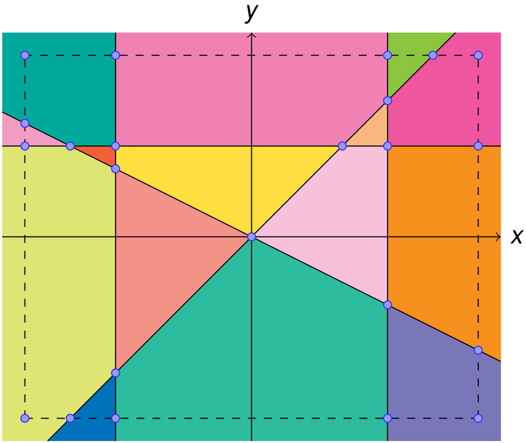
From a PLH VCSP to a finite domain VCSP: sampling



From a PLH VCSP to a finite domain VCSP: sampling



From a PLH VCSP to a finite domain VCSP: sampling



Observation: The representation of PLH cost functions suggests a "valued sampling approach".

Sampling

$\mathfrak{A}$: a structure with a finite relational signature τ.

Definition

A sampling algorithm for $\mathfrak{A}$ takes as input a positive integer *d* and computes a finite structure $\mathfrak{B}$ s.t. for every conjunction of τ-atomic formulas, χ, having at most *d* distinct free variables,

χ is satisfiable in $\mathfrak{A} \iff \chi$ is satisfiable in $\mathfrak{B}$.

Sampling

$\mathfrak{A}$: a structure with a finite relational signature τ.

Definition

A sampling algorithm for $\mathfrak{A}$ takes as input a positive integer *d* and computes a finite structure $\mathfrak{B}$ s.t. for every conjunction of τ-atomic formulas, χ, having at most *d* distinct free variables,

χ is satisfiable in $\mathfrak{A} \iff \chi$ is satisfiable in $\mathfrak{B}$.

A sampling algorithm is efficient if its running time is bounded by a polynomial in *d*.

Towards a valued sampling algorithm

We will adapt the sampling technique to “valued sampling” for VCSPs and find such an efficient sampling for PLH VCSPs.

Towards a valued sampling algorithm

We will adapt the sampling technique to “valued sampling” for VCSPs and find such an efficient sampling for PLH VCSPs.

Problems:

- the regions of linearity need not be closed, in general;

Towards a valued sampling algorithm

We will adapt the sampling technique to “valued sampling” for VCSPs and find such an efficient sampling for PLH VCSPs.

Problems:

- the regions of linearity need not be closed, in general;
- the regions of linearity need not be bounded, in general;

Towards a valued sampling algorithm

We will adapt the sampling technique to “valued sampling” for VCSPs and find such an efficient sampling for PLH VCSPs.

Problems:

- the regions of linearity need not be closed, in general;
- the regions of linearity need not be bounded, in general;
- the sample might have super-polynomial size.

The standard trick of non-standard analysis

We interpret our cost functions in a new domain:

$$\mathbb{Q}^\star := \left\{ \sum_{i=-\infty}^{+\infty} a_i \epsilon^i \;\middle|\; a_i \in \mathbb{Q}, \text{ and } a_i \neq 0 \text{ for only finitely many negative values of } i \right\}$$

The standard trick of non-standard analysis

We interpret our cost functions in a new domain:

$$\mathbb{Q}^\star := \left\{ \sum_{i=-\infty}^{+\infty} a_i \epsilon^i \;\middle|\; a_i \in \mathbb{Q}, \text{ and } a_i \neq 0 \text{ for only finitely many negative values of } i \right\}$$

- the ring operations on $\mathbb{Q}^\star$ are defined, as usual, componentwise;

The standard trick of non-standard analysis

We interpret our cost functions in a new domain:

$$\mathbb{Q}^\star := \left\{ \sum_{i=-\infty}^{+\infty} a_i \epsilon^i \;\middle|\; a_i \in \mathbb{Q}, \text{ and } a_i \neq 0 \text{ for only finitely many negative values of } i \right\}$$

- the ring operations on $\mathbb{Q}^\star$ are defined, as usual, componentwise;
- the order is the lexicographic one induced by $0 < \epsilon << 1$.

The standard trick of non-standard analysis

We interpret our cost functions in a new domain:

$$\mathbb{Q}^\star := \left\{ \sum_{i=-\infty}^{+\infty} a_i \epsilon^i \;\middle|\; a_i \in \mathbb{Q}, \text{ and } a_i \neq 0 \text{ for only finitely many negative values of } i \right\}$$

- the ring operations on $\mathbb{Q}^\star$ are defined, as usual, componentwise;
- the order is the lexicographic one induced by $0 < \epsilon << 1$.

Remark: $\mathbb{Q}^\star$ is a ordered field containing $\mathbb{Q}$.

The standard trick of non-standard analysis

We interpret our cost functions in a new domain:

$$\mathbb{Q}^\star := \left\{ \sum_{i=-\infty}^{+\infty} a_i \epsilon^i \;\middle|\; a_i \in \mathbb{Q}, \text{ and } a_i \neq 0 \text{ for only finitely many negative values of } i \right\}$$

- the ring operations on $\mathbb{Q}^\star$ are defined, as usual, componentwise;
- the order is the lexicographic one induced by $0 < \epsilon << 1$.

Remark: $\mathbb{Q}^\star$ is a ordered field containing $\mathbb{Q}$.

For all $m < n$ we define the vector space

$$\mathbb{Q}^\star_{m,n} := \left\{ \sum_{i=m}^{n} a_i \epsilon^i \;\middle|\; a_i \in \mathbb{Q} \right\}$$

An efficient sampling for PLH cost functions

Theorem (Bodirsky-Mamino-V., 2018)

Let $\Phi(x_1, \ldots, x_d)$ *be a set of atomic PLH formulas. Let* $u, \alpha_1, \ldots, \alpha_d \in \mathbb{Q}$.

An efficient sampling for PLH cost functions

Theorem (Bodirsky-Mamino-V., 2018)

Let $\Phi(x_1, \ldots, x_d)$ be a set of atomic PLH formulas. Let $u, \alpha_1, \ldots, \alpha_d \in \mathbb{Q}$. Then there exists a finite

$$C \subset \mathbb{Q}^{\star}_{-1,4}$$

An efficient sampling for PLH cost functions

Theorem (Bodirsky-Mamino-V., 2018)

Let $\Phi(x_1, \ldots, x_d)$ *be a set of atomic PLH formulas. Let* $u, \alpha_1, \ldots, \alpha_d \in \mathbb{Q}$*. Then there exists a finite*

$$C \subset \mathbb{Q}^{\star}_{-1,4}$$

such that TFAE:

1. *the formulas in* Φ *are simultaneously satisfiable in* $\mathbb{Q}$ *by a point* $(x_1, \ldots, x_d) \in \mathbb{Q}^d$ *s.t.* $\sum \alpha_i x_i \leq u$
2. *the formulas in* Φ *are simultaneously satisfiable in* $\mathbb{Q}^{\star}$ *by a point* $(x'_1, \ldots, x'_d) \in C^d$ *s.t.* $\sum \alpha_i x'_i \leq u$*.*

An efficient sampling for PLH cost functions

Theorem (Bodirsky-Mamino-V., 2018)

Let $\Phi(x_1, \ldots, x_d)$ be a set of atomic PLH formulas. Let $u, \alpha_1, \ldots, \alpha_d \in \mathbb{Q}$. Then there exists a finite

$$C \subset \mathbb{Q}^{\star}_{-1,4}$$

such that TFAE:

1. *the formulas in Φ are simultaneously satisfiable in $\mathbb{Q}$ by a point $(x_1, \ldots, x_d) \in \mathbb{Q}^d$ s.t. $\sum \alpha_i x_i \leq u$*
2. *the formulas in Φ are simultaneously satisfiable in $\mathbb{Q}^{\star}$ by a point $(x'_1, \ldots, x'_d) \in C^d$ s.t. $\sum \alpha_i x'_i \leq u$.*

Remarks:

- $|C|$ is polynomial in the size of the instance;

An efficient sampling for PLH cost functions

Theorem (Bodirsky-Mamino-V., 2018)

Let $\Phi(x_1, \ldots, x_d)$ be a set of atomic PLH formulas. Let $u, \alpha_1, \ldots, \alpha_d \in \mathbb{Q}$. Then there exists a finite

$$C \subset \mathbb{Q}^{\star}_{-1,4}$$

such that TFAE:

1. *the formulas in Φ are simultaneously satisfiable in $\mathbb{Q}$ by a point $(x_1, \ldots, x_d) \in \mathbb{Q}^d$ s.t. $\sum \alpha_i x_i \leq u$*
2. *the formulas in Φ are simultaneously satisfiable in $\mathbb{Q}^\star$ by a point $(x'_1, \ldots, x'_d) \in C^d$ s.t. $\sum \alpha_i x'_i \leq u$.*

Remarks:

- $|C|$ is polynomial in the size of the instance;
- C is computable in polynomial time.

An efficient sampling for PLH cost functions

Theorem (Bodirsky-Mamino-V., 2018)

Let $\Phi(x_1, \ldots, x_d)$ be a set of atomic PLH formulas. Let $u, \alpha_1, \ldots, \alpha_d \in \mathbb{Q}$. Then there exists a finite

$$C \subset \mathbb{Q}^{\star}_{-1,4}$$

such that TFAE:

1. *the formulas in Φ are simultaneously satisfiable in $\mathbb{Q}$ by a point $(x_1, \ldots, x_d) \in \mathbb{Q}^d$ s.t. $\sum \alpha_i x_i \leq u$*
2. *the formulas in Φ are simultaneously satisfiable in $\mathbb{Q}^{\star}$ by a point $(x'_1, \ldots, x'_d) \in C^d$ s.t. $\sum \alpha_i x'_i \leq u$.*

Remarks:

- $|C|$ is polynomial in the size of the instance;
- C is computable in polynomial time.

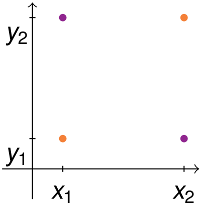
Consequence: there exists an efficient “valued sampling algorithm” for valued PLH languages.

Submodular PLH VCSPs

A function $f\colon \mathbb{Q}^n \to \mathbb{Q} \cup \{+\infty\}$ is submodular if, for all $x, y \in \mathbb{Q}^n$

$$f(x) + f(y) \geq f(\min(x, y)) + f(\max(x, y))$$

where min and max are applied componentwise.

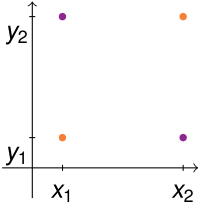


Submodular PLH VCSPs

A function $f : \mathbb{Q}^n \to \mathbb{Q} \cup \{+\infty\}$ is submodular if, for all $x, y \in \mathbb{Q}^n$

$$f(x) + f(y) \geq f(\min(x, y)) + f(\max(x, y))$$

where min and max are applied componentwise.



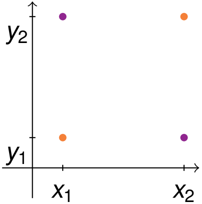
A language Γ is submodular if every element of Γ is submodular.

Submodular PLH VCSPs

A function $f\colon \mathbb{Q}^n \to \mathbb{Q} \cup \{+\infty\}$ is submodular if, for all $x, y \in \mathbb{Q}^n$

$$f(x) + f(y) \geq f(\min(x, y)) + f(\max(x, y))$$

where min and max are applied componentwise.



A language Γ is submodular if every element of Γ is submodular.

Theorem (Bodirsky-Mamino-V., 2018)

Let Γ be a finite language of submodular PLH cost functions. Then VCSP(Γ) is solvable in polynomial-time.

Maximal tractability

Let $\mathcal{V}$ be a class of valued constraint languages over a fixed domain D and let Γ be a language of $\mathcal{V}$, Γ is maximally tractable within $\mathcal{V}$ if

- $VCSP(\Gamma') \in P$ for every finite sublanguage $\Gamma' \subset \Gamma$; and
- for every $f \in \mathcal{V}$ such that $f \notin \Gamma$, there exists a finite sublanguage $\Delta \subseteq \Gamma$ such that $VCSP(\Delta \cup \{f\})$ is NP-hard.

Maximal tractability

Let $\mathcal{V}$ be a class of valued constraint languages over a fixed domain D and let Γ be a language of $\mathcal{V}$, Γ is maximally tractable within $\mathcal{V}$ if

- $VCSP(\Gamma') \in P$ for every finite sublanguage $\Gamma' \subset \Gamma$; and
- for every $f \in \mathcal{V}$ such that $f \notin \Gamma$, there exists a finite sublanguage $\Delta \subseteq \Gamma$ such that $VCSP(\Delta \cup \{f\})$ is NP-hard.

Theorem (Bodirsky-Mamino-V., 2018)

The valued language consisting of all submodular PLH cost functions is maximally tractable within the class of PLH valued languages.

Two possibilities:

$\mathbb{Q}^\star$

Two possibilities:

1. interpret our PLH functions in the domain $\mathbb{Q}^\star$,

$\mathbb{Q}^\star$

Two possibilities:

1. interpret our PLH functions in the domain $\mathbb{Q}^\star$,
2. substitute a suitably small rational value of ϵ in the formal expression of C and map the problem to $\mathbb{Q}$.

$\mathbb{Q}^\star$

Two possibilities:

1. interpret our PLH functions in the domain $\mathbb{Q}^\star$,
2. substitute a suitably small rational value of ϵ in the formal expression of C and map the problem to $\mathbb{Q}$.

In the first case we have to transfer the known approaches for $\mathbb{Q}$ to the new domain $\mathbb{Q}^\star$.
In the second case we can use them (after having computed a suitable ϵ).

Two possibilities:

1. interpret our PLH functions in the domain $\mathbb{Q}^\star$,
2. substitute a suitably small rational value of ϵ in the formal expression of C and map the problem to $\mathbb{Q}$.

In the first case we have to transfer the known approaches for $\mathbb{Q}$ to the new domain $\mathbb{Q}^\star$.
In the second case we can use them (after having computed a suitable ϵ).

For submodular PLH VCSPs we chose the first way.

Two possibilities:

1. interpret our PLH functions in the domain $\mathbb{Q}^\star$,
2. substitute a suitably small rational value of ϵ in the formal expression of C and map the problem to $\mathbb{Q}$.

In the first case we have to transfer the known approaches for $\mathbb{Q}$ to the new domain $\mathbb{Q}^\star$.
In the second case we can use them (after having computed a suitable ϵ).

For submodular PLH VCSPs we chose the first way.

Thank you