

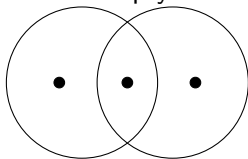
Satisfiability with Equivalences in Agreement, Part 1

Krasimir Georgiev

September 1, 2016

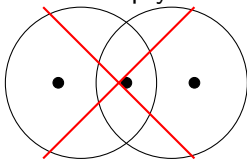
Agreement

- ▶ The sets A and B *intersect strictly* if $A \cap B$, $A \setminus B$ and $B \setminus A$ are nonempty.



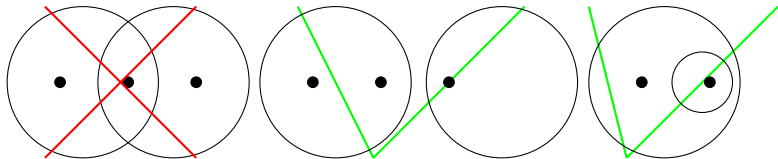
Agreement

- ▶ The sets A and B *intersect strictly* if $A \cap B$, $A \setminus B$ and $B \setminus A$ are nonempty.



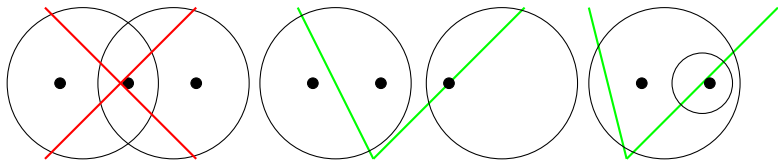
- ▶ What happens if we avoid these?

Agreement



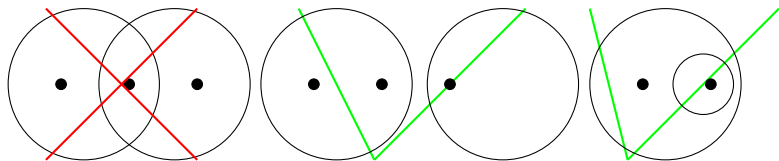
- ▶ Two equivalence relations G, E on A are *in local agreement* if their classes do not strictly intersect

Agreement



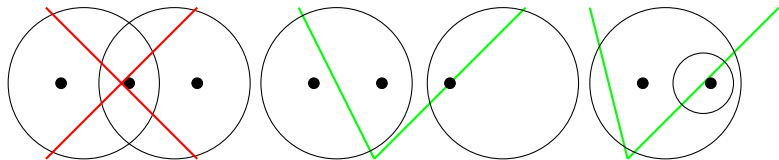
- ▶ Two equivalence relations G, E on A are *in local agreement* if their classes do not strictly intersect
- ▶ equivalently either $G[a] \subseteq E[a]$ or $E[a] \subseteq G[a]$ for every $a \in A$

Agreement



- ▶ Two equivalence relations G, E on A are *in local agreement* if their classes do not strictly intersect
- ▶ equivalently either $G[a] \subseteq E[a]$ or $E[a] \subseteq G[a]$ for every $a \in A$
- ▶ (colored) foam hierarchy: balloons in balloons

Agreement



- ▶ Two equivalence relations G, E on A are *in local agreement* if their classes do not strictly intersect
- ▶ equivalently either $G[a] \subseteq E[a]$ or $E[a] \subseteq G[a]$ for every $a \in A$
- ▶ (colored) foam hierarchy: balloons in balloons
- ▶ siam twin matryoshkas

Agreement

- *(colored) foam hierarchy: balloons in balloons*



image © 2014 Asia Citro, reproduced with permission from

<http://www.funathomewithkids.com/2014/07/magic-foaming-treasure-stars.html>

Agreement

- ▶ *(colored) foam hierarchy: balloons in balloons*



image © 2014 Asia Citro, reproduced with permission from

<http://www.funathomewithkids.com/2014/07/magic-foaming-treasure-stars.html>

- ▶ *siam twin matryoshkas*

This work

- ▶ define and characterize notions of agreement – ***refinement***, ***local agreement*** and ***global agreement***
- ▶ find the **computational complexity** of the **satisfiability** of the ***monadic*** and the ***two-variable*** fragment with respect to these notions

Overview

Setups

Equivalence Relations

Reductions

Monadic Logics

Setups

- ▶ use unary predicate symbols to “encode” data at elements of structures
- ▶ example: a permutation setup “encodes” a permutation at every element of a structure
- ▶ bits \rightarrow binary counters \rightarrow vectors \rightarrow permutations

Bit Setups

The set of *bits* is $\mathbb{B} = \{0, 1\}$. A *bit setup* is a predicate signature $B = \langle \mathbf{u} \rangle$ consisting of a single unary predicate symbol \mathbf{u} .

- ▶ Given \mathfrak{A} , define the function $[\mathbf{u}:\text{data}]^{\mathfrak{A}} : A \rightarrow \mathbb{B}$ by:

$$[\mathbf{u}:\text{data}]^{\mathfrak{A}} a = \begin{cases} 1 & \text{if } \mathfrak{A} \models \mathbf{u}(a) \\ 0 & \text{otherwise.} \end{cases}$$

- ▶ If $d \in \mathbb{B}$, define the formula $[\mathbf{u}:\text{eq-}d](\mathbf{x})$ by:

$$[\mathbf{u}:\text{eq-}d](\mathbf{x}) = \begin{cases} \mathbf{u}(\mathbf{x}) & \text{if } d = 1 \\ \neg \mathbf{u}(\mathbf{x}) & \text{otherwise.} \end{cases}$$

- ▶ Property: $\mathfrak{A} \models [\mathbf{u}:\text{eq-}d](a)$ iff $[\mathbf{u}:\text{data}]^{\mathfrak{A}} a = d$.

Bit Setups Formulas

$$[u:eq](x, y) = u(x) \leftrightarrow u(y)$$

$$[u:eq-01](x, y) = \neg u(x) \wedge u(y)$$

$$[u:eq-10](x, y) = u(x) \wedge \neg u(y).$$

- ▶ $\mathfrak{A} \models [u:eq](a, b)$ iff $[u:data]^{\mathfrak{A}}a = [u:data]^{\mathfrak{A}}b$.
- ▶ $\mathfrak{A} \models [u:eq-01](a, b)$ iff $[u:data]^{\mathfrak{A}}a = 0$ and $[u:data]^{\mathfrak{A}}b = 1$.
- ▶ $\mathfrak{A} \models [u:eq-10](a, b)$ iff $[u:data]^{\mathfrak{A}}a = 1$ and $[u:data]^{\mathfrak{A}}b = 0$.

Counter Setups

The set of *t-bit numbers* is $\mathbb{B}_t = [0, 2^t - 1]$. A *t-bit counter setup* is a predicate signature $C = \langle \mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_t \rangle$ consisting of t unary predicate symbols.

- ▶ Given \mathfrak{A} , define the function $[C:\text{data}]^{\mathfrak{A}} : A \rightarrow \mathbb{B}_t$ that returns a t -bit number for any $a \in A$ by:

$$[C:\text{data}]^{\mathfrak{A}} a = \sum_{1 \leq i \leq t} 2^{i-1} [\mathbf{u}_i:\text{data}]^{\mathfrak{A}} a.$$

Counter Setups Formulas

We can define **small formulas** with the following properties:

- ▶ $\mathfrak{A} \models [\text{C:eq-}d](a)$ iff $[\text{C:data}]^{\mathfrak{A}} a = d$.
- ▶ $\mathfrak{A} \models [\text{C:eq}](a, b)$ iff $[\text{C:data}]^{\mathfrak{A}} a = [\text{C:data}]^{\mathfrak{A}} b$.
- ▶ $\mathfrak{A} \models [\text{C:less}](a, b)$ iff $[\text{C:data}]^{\mathfrak{A}} a < [\text{C:data}]^{\mathfrak{A}} b$.
- ▶ $\mathfrak{A} \models [\text{C:succ}](a, b)$ iff $[\text{C:data}]^{\mathfrak{A}} b = 1 + [\text{C:data}]^{\mathfrak{A}} a$.
- ▶ $\mathfrak{A} \models [\text{C:betw-}d\text{-}e](a)$ iff $d \leq [\text{C:data}]^{\mathfrak{A}} a \leq e$.

Vector Setups

The set of n -dimensional t -bit vectors is \mathbb{B}_t^n . An n -dimensional t -bit vector setup is a predicate signature $V = \langle \mathbf{u}_{11}, \mathbf{u}_{12}, \dots, \mathbf{u}_{nt} \rangle$ of (nt) distinct unary predicate symbols.

- ▶ The *counter setup* $V(p)$ of V at position $p \in [1, n]$ is $V(p) = \langle \mathbf{u}_{p1}, \mathbf{u}_{p2}, \dots, \mathbf{u}_{pt} \rangle$.
- ▶ $[V:\text{data}]^{\mathfrak{A}} a = \left([V(1):\text{data}]^{\mathfrak{A}} a, [V(2):\text{data}]^{\mathfrak{A}} a, \dots, [V(n):\text{data}]^{\mathfrak{A}} a \right)$.

Vector Setups **Small Formulas**

- ▶ $\mathfrak{A} \models [V(pq):eq](a)$ iff $[V(p):data]^{\mathfrak{A}} a = [V(q):data]^{\mathfrak{A}} a$.
- ▶ $\mathfrak{A} \models [V(pq):less](a)$ iff $[V(p):data]^{\mathfrak{A}} a < [V(q):data]^{\mathfrak{A}} a$.
- ▶ $\mathfrak{A} \models [V(pq):succ](a)$ iff $[V(q):data]^{\mathfrak{A}} a = 1 + [V(p):data]^{\mathfrak{A}} a$.
- ▶ antilexicographic ordering, e.g. $(1, 1, 0) \prec (0, 0, 1)$:
 $\mathfrak{A} \models [V:less](a, b)$ iff $[V:data]^{\mathfrak{A}} a \prec [V:data]^{\mathfrak{A}} b$.

Permutation Setups

The set of permutations of $[1, n]$ is \mathbb{S}_n .

Encode an n -permutation $\nu \in \mathbb{S}_n$ by the n -dimensional t -bit vector $(\nu(1), \nu(2), \dots, \nu(n))$, where t is the bitsize of n .

An n -permutation setup $P = \langle \mathbf{u}_{11}, \mathbf{u}_{12}, \dots, \mathbf{u}_{nt} \rangle$ is just an n -dimensional t -bit vector.

The **small formula** $[P:\text{perm}] = [P:\text{betw-1-}n] \wedge [P:\text{alldiff}]$ asserts that the vector setup encodes exactly the permutations.

Agreement

Let E_1, E_2, \dots, E_n be a sequence of equivalence relations on A .
The sequence is in

- *refinement* if it forms a chain under inclusion, that is if
$$E_1 \subseteq E_2 \subseteq \dots \subseteq E_n$$

Agreement

Let E_1, E_2, \dots, E_n be a sequence of equivalence relations on A .
 The sequence is in

- ▶ *refinement* if it forms a chain under inclusion, that is if $E_1 \subseteq E_2 \subseteq \dots \subseteq E_n$
- ▶ *global agreement* if it can be rearranged into a chain under inclusion, that is if $E_{\nu(1)} \subseteq E_{\nu(2)} \subseteq \dots \subseteq E_{\nu(n)}$ for some permutation ν of $[1, n]$

Agreement

Let E_1, E_2, \dots, E_n be a sequence of equivalence relations on A .
 The sequence is in

- ▶ *refinement* if it forms a chain under inclusion, that is if $E_1 \subseteq E_2 \subseteq \dots \subseteq E_n$
- ▶ *global agreement* if it can be rearranged into a chain under inclusion, that is if $E_{\nu(1)} \subseteq E_{\nu(2)} \subseteq \dots \subseteq E_{\nu(n)}$ for some permutation ν of $[1, n]$
- ▶ *local agreement* if for every $a \in A$, the sequence of equivalence classes $E_1[a], E_2[a], \dots, E_n[a]$ can be rearranged into a chain under inclusion, that is if for every $a \in A$ there is some permutation $\nu(a)$ of $[1, n]$ such that $E_{\nu(a)(1)}[a] \subseteq E_{\nu(a)(2)}[a] \subseteq \dots \subseteq E_{\nu(a)(n)}[a]$

Agreement

Let E_1, E_2, \dots, E_n be a sequence of equivalence relations on A .
 The sequence is in

- ▶ *refinement* if it forms a chain under inclusion, that is if $E_1 \subseteq E_2 \subseteq \dots \subseteq E_n$
- ▶ *global agreement* if it can be rearranged into a chain under inclusion, that is if $E_{\nu(1)} \subseteq E_{\nu(2)} \subseteq \dots \subseteq E_{\nu(n)}$ for some permutation ν of $[1, n]$
- ▶ *local agreement* if for every $a \in A$, the sequence of equivalence classes $E_1[a], E_2[a], \dots, E_n[a]$ can be rearranged into a chain under inclusion, that is if for every $a \in A$ there is some permutation $\nu(a)$ of $[1, n]$ such that $E_{\nu(a)(1)}[a] \subseteq E_{\nu(a)(2)}[a] \subseteq \dots \subseteq E_{\nu(a)(n)}[a]$
- ▶ $\text{refinement} \implies \text{global agreement} \implies \text{local agreement}$

Intuition

Intuitively

- ▶ global agreement = refinement + a permutation

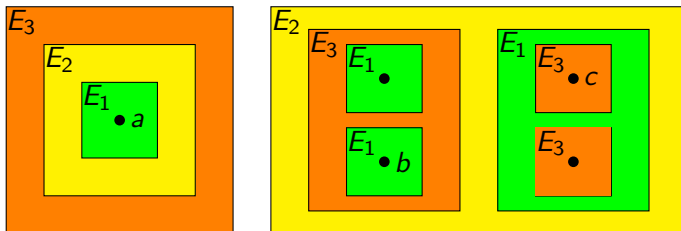
Intuition

Intuitively

- ▶ global agreement = refinement + a permutation
- ▶ local agreement = refinement + locally agreeing permutations

Example

Example of a sequence E_1, E_2, E_3 in local agreement:



Characterization

Lemma

The sequence E_1, E_2 of two equivalence relations on A is in local agreement iff $E_1 \cup E_2$ is an equivalence relation on A .

Theorem

The sequence E_1, E_2, \dots, E_n of equivalence relations on A is in local agreement iff the union of any nonempty subsequence is an equivalence relation on A , that is for any $m \in [1, n]$ and $1 \leq i_1 < i_2 < \dots < i_m \leq n$ we have that $E_{i_1} \cup E_{i_2} \cup \dots \cup E_{i_m}$ is an equivalence relation on A .

Levels

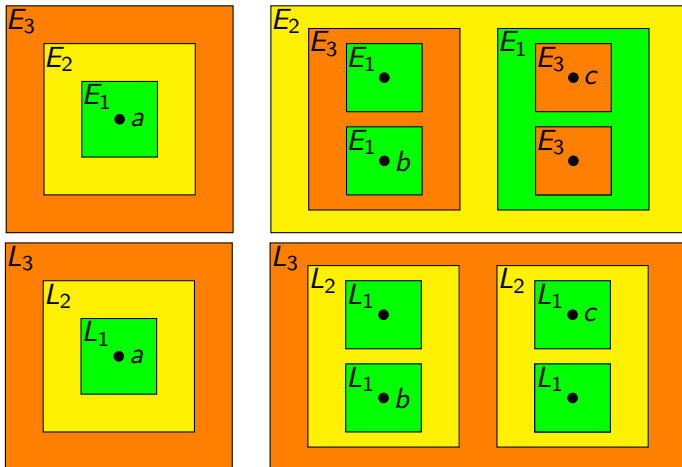
The *level sequence* L_1, L_2, \dots, L_n of the sequence E_1, E_2, \dots, E_n of equivalence relations on A in local agreement is defined by:

$$L_m = \bigcap \{ E_{i_1} \cup E_{i_2} \cup \dots \cup E_{i_m} \mid 1 \leq i_1 < i_2 < \dots < i_m \leq n \}.$$

Remark

The level sequence is a sequence of equivalence relations on A in refinement.

Example



Permutations

Lemma

Let E_1, E_2, \dots, E_n be a sequence of equivalence relations on A in local agreement having level sequence L_1, L_2, \dots, L_n . Suppose that $a \in A$ and that ν is any permutation witnessing the local agreement at a :

$$E_{\nu(1)}[a] \subseteq E_{\nu(2)}[a] \subseteq \dots \subseteq E_{\nu(n)}[a].$$

Then $L_k[a] = E_{\nu(k)}[a]$ for any $k \in [1, n]$.

Notation

$$\mathcal{L}_p^v eE_a$$

- \mathcal{L} , the *ground logic*, is the first-order logic with formal equality

Notation

$$\mathcal{L}_p^v \text{eE}_a$$

- ▶ \mathcal{L} , the *ground logic*, is the first-order logic with formal equality
- ▶ v , if given, bounds the number of allowed variables

Notation

$$\mathcal{L}_p^v e E_a$$

- ▶ \mathcal{L} , the *ground logic*, is the first-order logic with formal equality
- ▶ v , if given, bounds the number of allowed variables
- ▶ e , if given, specifies the number of builtin equivalence symbols

Notation

$$\mathcal{L}_p^v e E_a$$

- ▶ \mathcal{L} , the *ground logic*, is the first-order logic with formal equality
- ▶ v , if given, bounds the number of allowed variables
- ▶ e , if given, specifies the number of builtin equivalence symbols
- ▶ $a \in \{\text{refine, global, local}\}$, if given, specifies the agreement condition between the equivalence symbols

Notation

$$\mathcal{L}_p^v e E_a$$

- ▶ \mathcal{L} , the *ground logic*, is the first-order logic with formal equality
- ▶ v , if given, bounds the number of allowed variables
- ▶ e , if given, specifies the number of builtin equivalence symbols
- ▶ $a \in \{\text{refine, global, local}\}$, if given, specifies the agreement condition between the equivalence symbols
- ▶ if p is not given, the signature consists of arbitrary many unary and binary predicate symbols

Notation

$$\mathcal{L}_p^v e E_a$$

- ▶ \mathcal{L} , the *ground logic*, is the first-order logic with formal equality
- ▶ v , if given, bounds the number of allowed variables
- ▶ e , if given, specifies the number of builtin equivalence symbols
- ▶ $a \in \{\text{refine, global, local}\}$, if given, specifies the agreement condition between the equivalence symbols
- ▶ if p is not given, the signature consists of arbitrary many unary and binary predicate symbols
- ▶ if $p = 0$, the signature consists of only constantly many unary predicate symbols

Notation

$$\mathcal{L}_p^v e E_a$$

- ▶ \mathcal{L} , the *ground logic*, is the first-order logic with formal equality
- ▶ v , if given, bounds the number of allowed variables
- ▶ e , if given, specifies the number of builtin equivalence symbols
- ▶ $a \in \{\text{refine, global, local}\}$, if given, specifies the agreement condition between the equivalence symbols
- ▶ if p is not given, the signature consists of arbitrary many unary and binary predicate symbols
- ▶ if $p = 0$, the signature consists of only constantly many unary predicate symbols
- ▶ if $p = 1$, the signature consists of arbitrary many unary predicate symbols

Examples

- ▶ \mathcal{L}_01E is the logic of a single equivalence
- ▶ \mathcal{L}_1 is the monadic fragment
- ▶ $\mathcal{L}^22E_{\text{local}}$ is the two-variable logic featuring unary and binary predicate symbols in addition to two builtin equivalence symbols in local agreement

Reduction Strategy

To reduce $(\text{FIN-})\text{SAT-}\mathcal{L}eE_{\text{local}}$ to $(\text{FIN-})\text{SAT-}\mathcal{L}eE_{\text{refine}}$,

- ▶ look at the levels
- ▶ encode a permutation witnessing the local agreement in a permutation setup
- ▶ define formulas that recover the original equivalences from the levels and the permutations
- ▶ not every combination of levels and permutations defines local agreement \implies need constraint on permutations

Characteristic Permutations

Consider an $\mathcal{L}eE_{\text{local}}$ -signature Σ containing $E = \langle \mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_e \rangle$.
 Let \mathfrak{A} be a Σ -structure, $E_i = \mathbf{e}_i^{\mathfrak{A}}$ and $a \in A$.

The *characteristic permutation* ν at a is the antilexicographically smallest permutation of $[1, e]$ satisfying:

$$E_{\nu(1)}[a] \subseteq E_{\nu(2)}[a] \subseteq \dots \subseteq E_{\nu(e)}[a].$$

Collect the characteristic permutations in $[\Sigma:\text{chperm}]^{\mathfrak{A}} : A \rightarrow S_e$.

Local Agreement of Permutations

Remark

Let L_1, L_2, \dots, L_e be the levels of E_i , $a, b \in A$, $\alpha = [\Sigma:\text{chperm}]^{\mathfrak{A}} a$
 and $\beta = [\Sigma:\text{chperm}]^{\mathfrak{A}} b$.

If $(a, b) \in L_k$, then $\alpha(k) = \beta(k)$.

That is, if a and b are connected at level k , then their
 characteristic permutations agree at position k .

- ▶ This doesn't hold in general for any set of witnessing permutations

Levels and Permutations

Let $L = \langle l_1, l_2, \dots, l_e \rangle + P$ consist of the builtin equivalence symbols l_i (we intend to interpret them as the levels) together with the permutation setup P (intended to encode the characteristic permutations).

The formula

$$[L:\text{fixperm}] = \forall \mathbf{x} \forall \mathbf{y} \bigwedge_{1 \leq k \leq e} (l_k(\mathbf{x}, \mathbf{y}) \rightarrow [P(k):\text{eq}](\mathbf{x}, \mathbf{y})).$$

encodes the local agreement of permutations.

Recovering

The formulas

$$[L:el-i](\mathbf{x}, \mathbf{y}) = \bigwedge_{1 \leq k \leq e} ([P(k):eq-i](\mathbf{x}) \rightarrow I_k(\mathbf{x}, \mathbf{y}))$$

recover the original equivalences ($i \in [1, e]$).

Remark

Let \mathfrak{A} be an L -structure satisfying $[P:perm] \wedge [L:fixperm]$ such that the level symbols I_i are interpreted as a sequence of equivalence relations in refinement. Let $L_i = I_i^{\mathfrak{A}}$ and $E_i = [L:el-i]^{\mathfrak{A}}$. Then E_i is a sequence of equivalence relations in local agreement and $L_k[a] = E_{\alpha(k)}[a]$ for any $a \in A$ and $\alpha = [P:data]^{\mathfrak{A}}$.

Translation

Let $\Sigma' = \Sigma + L$ and $L' = \Sigma' - E$. The translation $\text{ltr } \varphi : \mathcal{L}[\Sigma] \rightarrow \mathcal{L}[L']$ is defined by

$$\text{ltr } \varphi = \varphi' \wedge [P:\text{perm}] \wedge [L:\text{fixperm}],$$

where φ' is obtained from φ by replacing all occurrences of $e_i(x, y)$ by $[L:\text{el-}i](x, y)$.

Remark

φ is (finitely) satisfiable over $\mathcal{L}eE_{\text{local}}$ iff $\text{ltr } \varphi$ is (finitely) satisfiable over $\mathcal{L}eE_{\text{refine}}$.

- ▶ if $\mathfrak{A} \models \varphi$, interpret I_i as the levels and encode $[\Sigma:\text{chperm}]^{\mathfrak{A}}$ in the permutation setup P .
- ▶ if $\mathfrak{A}' \models \text{ltr } \varphi$, interpret e_i as $[L:\text{el-}i]$.

Translation

The translation just uses polynomially many new unary predicate symbols (it can “reuse” the builtin equivalences).

Proposition

- ▶ *the logic $\mathcal{L}eE_{\text{local}}$ has the finite model property iff the logic $\mathcal{L}eE_{\text{refine}}$ has the finite model property*
- ▶ *the corresponding satisfiability problems are polynomial-time equivalent*
- ▶ *also works for $\mathcal{L}_1eE_{\text{local}}$ and $\mathcal{L}^2eE_{\text{local}}$*

Monadic Logics

Results

It is known that:

- ▶ \mathcal{L}_1 has the finite model property [Löwenheim 1915] and its satisfiability problem is NEXPTIME -complete

Results

It is known that:

- ▶ \mathcal{L}_1 has the finite model property [Löwenheim 1915] and its satisfiability problem is NEXPTIME -complete
- ▶ \mathcal{L}_01E has the finite model property and its satisfiability problem is PSPACE -complete

Results

It is known that:

- ▶ \mathcal{L}_1 has the finite model property [Löwenheim 1915] and its satisfiability problem is NEXPTIME -complete
- ▶ \mathcal{L}_01E has the finite model property and its satisfiability problem is PSPACE -complete
- ▶ \mathcal{L}_02E lacks the finite model property and both the satisfiability and finite satisfiability are undecidable [Janiczak 1953]

Results

It is known that:

- ▶ \mathcal{L}_1 has the finite model property [Löwenheim 1915] and its satisfiability problem is NEXPTIME -complete
- ▶ \mathcal{L}_01E has the finite model property and its satisfiability problem is PSPACE -complete
- ▶ \mathcal{L}_02E lacks the finite model property and both the satisfiability and finite satisfiability are undecidable [Janiczak 1953]

How about \mathcal{L}_11E ?

Results

It is known that:

- ▶ \mathcal{L}_1 has the finite model property [Löwenheim 1915] and its satisfiability problem is NEXPTIME -complete
- ▶ \mathcal{L}_01E has the finite model property and its satisfiability problem is PSPACE -complete
- ▶ \mathcal{L}_02E lacks the finite model property and both the satisfiability and finite satisfiability are undecidable [Janiczak 1953]

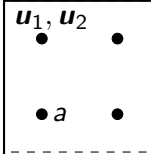
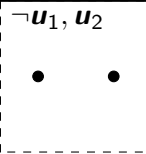
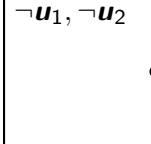

How about \mathcal{L}_11E ? **We show that:**

- ▶ \mathcal{L}_11E has the finite model property and its satisfiability problem is $\text{N}^2\text{EXPTIME}$ -complete
- ▶ in general, $\mathcal{L}_1eE_{\text{refine}}$ has the finite model property and its satisfiability problem is $\text{N}(e + 1)\text{EXPTIME}$ -complete

Complexity: Cells

Let $\Sigma = \langle \mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_u, \mathbf{e} \rangle$
and \mathfrak{A} be a Σ -structure.
A *cell* $C \subseteq A$ is a maximal
set of \mathbf{e} -equivalent elements
satisfying the same
 \mathbf{u} -predicates.

Example $\Sigma = \langle \mathbf{u}_1, \mathbf{u}_2, \mathbf{e} \rangle$

$E[a]$	$E[b]$
$\mathbf{u}_1, \mathbf{u}_2$ 	$\neg \mathbf{u}_1, \mathbf{u}_2$ 
$\neg \mathbf{u}_1, \neg \mathbf{u}_2$ 	$\neg \mathbf{u}_1, \neg \mathbf{u}_2$ 

Small Cells

Lemma

Let $r \geq 1$ and suppose that \mathfrak{A} is a Σ -structure. Then there is a substructure $\mathfrak{B} \subseteq \mathfrak{A}$ such that $\mathfrak{B} \equiv_r \mathfrak{A}$ and every \mathfrak{B} -cell has cardinality at most r .

Proof Idea.

For every \mathfrak{A} -cell, if it has less than r elements select them all, otherwise select any r elements. Consider \mathfrak{B} induced by the selected elements. Win the r -round *Ehrenfeucht-Fraïssé game* as Duplicator: if the challenge is new, choose a new selected element from the same cell. Since the game lasts r rounds, you'll never run out of selected elements. □

Few Isomorphic Classes

Lemma

Let $r \geq 1$ and suppose that \mathfrak{A} is a Σ -structure. Then there is a substructure $\mathfrak{B} \subseteq \mathfrak{A}$ such that $\mathfrak{B} \equiv_r \mathfrak{A}$ and \mathfrak{B} -class is isomorphic to at most $(r - 1)$ other \mathfrak{B} -classes.

Combining these and doing the math, we get:

Remark

Let \mathfrak{A} be a $\Sigma(u, 1)$ -structure and let $r \geq 1$. There is some $\mathfrak{B} \subseteq \mathfrak{A}$ such that $\mathfrak{B} \equiv_r \mathfrak{A}$ and $|B| \leq r^2 2^u ((r + 1)^{2^u} - 1)$.

This is doubly exponential with respect to the size of φ , hence (FIN-)SAT- \mathcal{L}_1 1E is in N2EXPTIME.

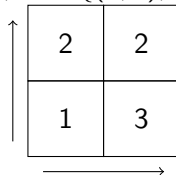
Hardness: Domino Problem

- ▶ Reduce the N2EXPTIME-complete Square Domino Tiling Problem to (FIN-)SAT- \mathcal{L}_1 1E.
- ▶ A *domino system* is a triple $D = (T, H, V)$, where $T = [1, k]$ is a set of *tiles* and $H, V \subseteq T \times T$ are the *horizontal* and *vertical matching relations*.
- ▶ A *tiling* of the $m \times m$ square for a domino system D with initial condition $c^0 = \langle t_1^0, t_2^0, \dots, t_n^0 \rangle$, where $n \leq m$, is a mapping $t : [1, m] \times [1, m] \rightarrow T$ such that:
 - ▶ $(t(i, j), t(i + 1, j)) \in H$ for all $i \in [1, m - 1], j \in [1, m]$
 - ▶ $(t(i, j), t(i, j + 1)) \in V$ for all $i \in [1, m], j \in [1, m - 1]$
 - ▶ $t(i, 1) = t_i^0$ for all $i \in [1, n]$

Domino Problem

Example $T = [1, 3]$,

$H = \{(1, 3), (2, 1), (2, 2)\}$, $V = \{(2, 2), (3, 2), (1, 2)\}$



Theorem

There is a domino system D_0 such that the problem of asking if there exists a tiling for D_0 with initial condition c_0 of length n for the $2^{2^n} \times 2^{2^n}$ -square is N2EXPTIME-complete.

Hardness

Main issue: given $\Sigma = \langle \mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_u, \mathbf{e} \rangle$, how can we define a doubly exponential grid?

Hardness

Main issue: given $\Sigma = \langle \mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_u, \mathbf{e} \rangle$, how can we define a doubly exponential grid?

- ▶ Each class can contain exponentially many cells
- ▶ If we encode bits in cells, the classes encode doubly exponential numbers

Encoding

- ▶ we can ensure that every class contains maximally many 2^u cells
- ▶ a cell containing a single element encodes bit 0:

$$[\Sigma:\text{bit-0}](x) = \forall y (e(y, x) \wedge [\Sigma:\text{pos-eq}](y, x) \rightarrow y = x)$$

- ▶ a cell containing more elements encodes bit 1:

$$[\Sigma:\text{bit-1}](x) = \exists y (e(y, x) \wedge [\Sigma:\text{pos-eq}](y, x) \wedge y \neq x)$$

Data

Let \mathfrak{A} be a Σ -structure and let $E = e^{\mathfrak{A}}$. The equivalence classes of E are $\mathcal{C}E$. The number encoded by the bitstring b is \underline{b} .

With *a bit of work* we can define:

- ▶ $[\Sigma:\text{Data}]^{\mathfrak{A}} : \mathcal{C}E \rightarrow \mathbb{B}^{2^u}$ that assigns exponential bitstrings (hence doubly exponential numbers) to the classes of \mathfrak{A}
- ▶ $\mathfrak{A} \models [\Sigma:\text{Zero}](a)$ iff $\underline{[\Sigma:\text{Data}]^{\mathfrak{A}} E[a]} = 0$
- ▶ $\mathfrak{A} \models [\Sigma:\text{Succ}](a, b)$ iff $\underline{[\Sigma:\text{Data}]^{\mathfrak{A}} E[b]} = 1 + \underline{[\Sigma:\text{Data}]^{\mathfrak{A}} E[a]}$
- ▶ etc.

Reduction

Given $D_0 = (T, V, H)$, where $T = [1, k]$, and $c^0 = \langle t_1^0, t_1^0, \dots, t_n^0 \rangle$, consider:

$$\Sigma = \langle \mathbf{u}_1^H, \mathbf{u}_2^H, \dots, \mathbf{u}_n^H; \mathbf{u}_1^V, \mathbf{u}_2^V, \dots, \mathbf{u}_n^V; \mathbf{u}_1^T, \mathbf{u}_2^T, \dots, \mathbf{u}_k^T; \mathbf{e} \rangle$$

$\mathbf{u}_1^T, \mathbf{u}_2^T, \dots, \mathbf{u}_k^T$ for tiles

$$\Sigma^{HV} = \langle \mathbf{u}_1^H, \mathbf{u}_2^H, \dots, \mathbf{u}_n^H; \mathbf{u}_1^V, \mathbf{u}_2^V, \dots, \mathbf{u}_n^V; \mathbf{e} \rangle \text{ for the full grid}$$

$$\Sigma^H = \langle \mathbf{u}_1^H, \mathbf{u}_2^H, \dots, \mathbf{u}_n^H; \mathbf{e} \rangle \text{ for horizontal matching}$$

$$\Sigma^V = \langle \mathbf{u}_1^V, \mathbf{u}_2^V, \dots, \mathbf{u}_n^V; \mathbf{e} \rangle \text{ for vertical matching}$$

Reduction

small formulas

- ▶ $[\Sigma^{HV}:\text{pos-full}] \wedge [\Sigma^{HV}:\text{Full}] \wedge [\Sigma^{HV}:\text{Alldiff}]$ defines a full doubly exponential grid
- ▶ $\forall \mathbf{x} \left(\bigvee_{1 \leq i \leq k} \left(\mathbf{u}_i^T(\mathbf{x}) \wedge \bigwedge_{j \in [1,k] \setminus \{i\}} \neg \mathbf{u}_j^T(\mathbf{x}) \right) \right)$ asserts that every element has a unique type
- ▶ $\forall \mathbf{x} \forall \mathbf{y} \left(\mathbf{e}(\mathbf{x}, \mathbf{y}) \rightarrow \bigwedge_{1 \leq i \leq k} (\mathbf{u}_i^T(\mathbf{x}) \leftrightarrow \mathbf{u}_i^T(\mathbf{y})) \right)$ asserts that the type is the same in each class

Reduction

small formulas

- ▶ $\forall \mathbf{x} \left([D^H:\text{Eq}-(j-1)](\mathbf{x}) \wedge [D^V:\text{Zero}](\mathbf{x}) \rightarrow \mathbf{u}_{t_j^0}^T(\mathbf{x}) \right)$ encodes the initial condition
- ▶ $\forall \mathbf{x} \forall \mathbf{y} \left([D^H:\text{Succ}](\mathbf{x}, \mathbf{y}) \wedge [D^V:\text{Eq}](\mathbf{x}, \mathbf{y}) \rightarrow \bigvee_{(i,j) \in H} \mathbf{u}_i^T(\mathbf{x}) \wedge \mathbf{u}_j^T(\mathbf{y}) \right)$ encodes the horizontal tiling condition

Summary

- ▶ this shows that the satisfiability of $\mathcal{L}^2 1E$ is N^2EXP^{TIME} -hard
- ▶ the argument can be generalized and applied by induction to show that the satisfiability of $\mathcal{L}^2 eE_{\text{refine}}$ is $N(e + 1)EXP^{TIME}$ -hard