On the Existential Fragments of Local First-Order Logics with Data

Benedikt Bollig, Arnaud Sangnier, **Olivier Stietel**

GandALF 2022

September 21, 2022

Introduction

Context

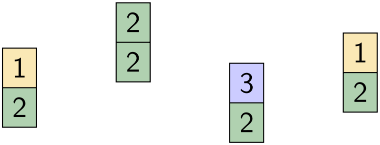
- Data-aware systems are omnipresent
  - Database
  - Sets of data for learning
  - Distributed/ Concurrent Systems
- Need for specification languages to describe systems with data

Motivation

- Logic to specify input-output behavior of distributed algorithms
- Structures with two data values
- The input values can be compared with the output values

Consensus problem

- Famous problem in distributed computing
- The goal is to design an algorithm such that:
  - All entities in a network have an input value
  - They should all agree on the same value
  - The chosen value should be one of the input values



First-order logic with data values

Structures

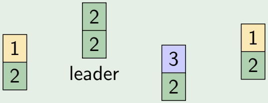
- Σ finite set of unary relation symbols
- $D \geq 0$ an integer

Definition

A D-structure is a tuple $\mathfrak{A} = (A, (P_\sigma)_{\sigma \in \Sigma}, f_1, \ldots, f_D)$ where:

- A is the nonempty finite universe
- $P_\sigma \subseteq A$ for all $\sigma \in \Sigma$
- $f_1, \ldots, f_D : A \to \mathbb{N}$ map each element to D data values

Example (2-structure)



Logic with data values

- Σ finite set of unary relation symbols
- $D \geq 0$ an integer

Definition

The logic dFO[Σ, D] is given as follows:

$$\varphi ::= \sigma(x) \mid x \; {}_i{\sim}_j \; y \mid x = y \mid \varphi \vee \varphi \mid \neg\varphi \mid \exists x.\varphi$$

where $\sigma \in \Sigma$ and $i, j \in \{1, \ldots, D\}$

Logic with data values

- Σ finite set of unary relation symbols
- $D \geq 0$ an integer

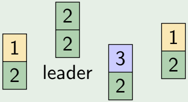
Definition

The logic dFO$[\Sigma, D]$ is given as follows:

$$\varphi ::= \sigma(x) \mid x \;{}_i\!\sim_j\; y \mid x = y \mid \varphi \vee \varphi \mid \neg\varphi \mid \exists x.\varphi$$

where $\sigma \in \Sigma$ and $i, j \in \{1, \ldots, D\}$

Example



$\models \exists^{=1} x.\text{leader}(x) \wedge \forall y.\exists x.(\text{leader}(x) \wedge x \;{}_1\!\sim_2\; y)$

Satisfiability problem

Definition

The problem $\mathrm{DataSat}(\mathcal{F}, D)$ is defined as follows:

Input: Finite set Σ; sentence $\varphi \in \mathcal{F}[\Sigma; D]$.

Question: Is there a data structure $\mathfrak{A}$ such that $\mathfrak{A} \models \varphi$?

Satisfiability problem

Definition

The problem DataSat($\mathcal{F}, D$) is defined as follows:

Input: Finite set Σ; sentence $\varphi \in \mathcal{F}[\Sigma; D]$.

Question: Is there a data structure $\mathfrak{A}$ such that $\mathfrak{A} \models \varphi$?

Theorem (Janiczak, 1953)

DataSat(dFO, 2) is undecidable, even when $\Sigma = \emptyset$ and without using ${}_1{\sim}_2$ and ${}_2{\sim}_1$.

Satisfiability problem

Definition

The problem DataSat($\mathcal{F}, D$) is defined as follows:

Input: Finite set Σ; sentence $\varphi \in \mathcal{F}[\Sigma; D]$.

Question: Is there a data structure $\mathfrak{A}$ such that $\mathfrak{A} \models \varphi$?

Theorem (Janiczak, 1953)

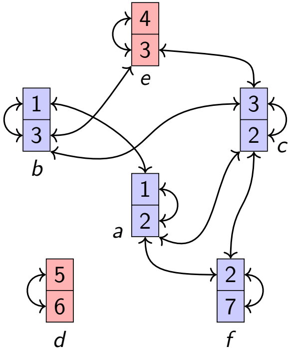
DataSat(dFO, 2) is undecidable, even when $\Sigma = \emptyset$ and without using ${}_1{\sim}_2$ and ${}_2{\sim}_1$.

Other related works:

- Works where the number of variables is bounded [Kieronski and Tendera, 2009]
- Two-variable logic on data words [Bojanczyk, David, Muscholl, Schwentick, and Segoufin, 2011]

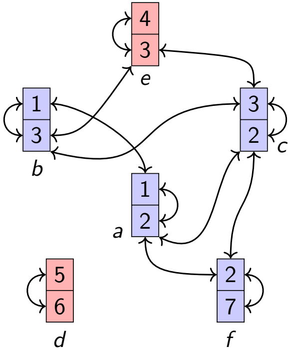
Local first-order logic

Local logics reason about neighborhoods



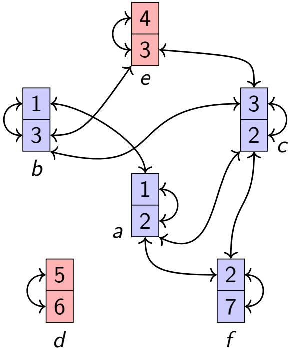
- $\mathfrak{A}$ — 2-data structure

Local logics reason about neighborhoods



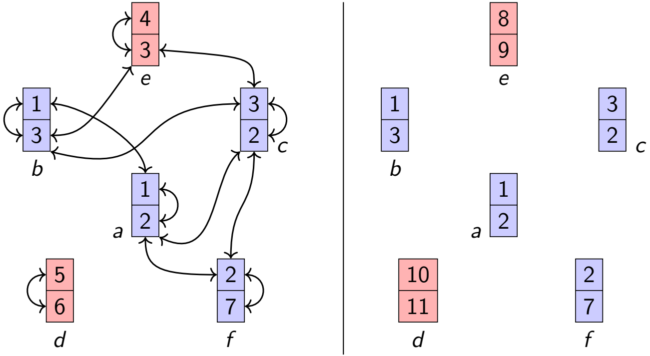
- $\mathfrak{A}$ — 2-data structure
- $\mathcal{G}(\mathfrak{A})$ — data graph

Local logics reason about neighborhoods



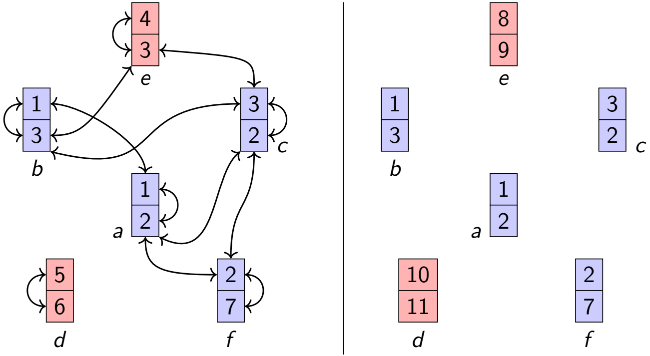
- $\mathfrak{A}$ — 2-data structure
- $\mathcal{G}(\mathfrak{A})$ — data graph
- $B_2^{\mathfrak{A}}(a)$ — 2-ball (blue nodes)

Local logics reason about neighborhoods



- $\mathfrak{A}$ — 2-data structure
- $\mathcal{G}(\mathfrak{A})$ — data graph
- $B_2^{\mathfrak{A}}(a)$ — 2-ball (blue nodes)
- $\mathfrak{A}|_a^2$ — 2-view of a

Local logics reason about neighborhoods



- $\mathfrak{A}$ — 2-data structure
- $\mathcal{G}(\mathfrak{A})$ — data graph
- $B_2^{\mathfrak{A}}(a)$ — 2-ball (blue nodes)
- $\mathfrak{A}|_a^2$ — 2-view of a
- local modality $\langle\!\langle \psi \rangle\!\rangle_x^2$ with $\psi \in \mathrm{FO}[\Sigma; \Gamma]$ can reason about $\mathfrak{A}|_a^2$

Local logic with data values

- Σ finite set of unary relation symbols
- $D > 0$ an integer

Definition

For a radius $r \in \mathbb{N}$, the logic r-Loc-dFO$[\Sigma, D]$ is given as follows:

$$\varphi \;::=\; \langle\!\langle \psi \rangle\!\rangle_x^r \mid x = y \mid \exists x.\varphi \mid \varphi \vee \varphi \mid \neg\varphi$$

where $\psi \in$ dFO$[\Sigma, D]$.

Local logic with data values

- Σ finite set of unary relation symbols
- $D > 0$ an integer

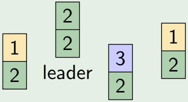
Definition

For a radius $r \in \mathbb{N}$, the logic r-Loc-dFO$[\Sigma, D]$ is given as follows:

$$\varphi \ ::= \ \langle\!\langle \psi \rangle\!\rangle_x^r \ \mid \ x = y \ \mid \ \exists x.\varphi \ \mid \ \varphi \vee \varphi \ \mid \ \neg\varphi$$

where $\psi \in$ dFO$[\Sigma, D]$.

Example



$$\models \quad \begin{array}{l} \exists^{=1} x.\langle\!\langle \mathrm{leader}(x) \rangle\!\rangle_x^1 \ \wedge \\ \forall y.\langle\!\langle \exists x.\mathrm{leader}(x) \wedge y \ {}_2{\sim}_1 \ x \rangle\!\rangle_y^1 \end{array}$$

Local logic with data values

- Σ finite set of unary relation symbols
- $D > 0$ an integer

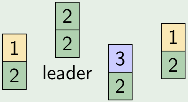
Definition

For a radius $r \in \mathbb{N}$, the logic r-Loc-dFO$[\Sigma, D]$ is given as follows:

$$\varphi \ ::= \ \langle\!\langle \psi \rangle\!\rangle_x^r \ \mid \ x = y \ \mid \ \exists x.\varphi \ \mid \ \varphi \vee \varphi \ \mid \ \neg\varphi$$

where $\psi \in$ dFO$[\Sigma, D]$.

Example



$$\models \quad \exists^{=1} x.\langle\!\langle \text{leader}(x) \rangle\!\rangle_x^1 \ \wedge$$
$$\forall y.\langle\!\langle \exists x.\text{leader}(x) \wedge y \ {}_2{\sim}_1 \ x \rangle\!\rangle_y^1$$
$$\in 1\text{-Loc-dFO}[\{\text{leader}\}, 2]$$

Previous results [FSTTCS'21]

Theorem

DataSat(2-Loc-dFO, 2) is undecidable

Theorem

DataSat(1-Loc-dFO, 2) is decidable
(with relations in $\{_1{\sim}_1, {}_2{\sim}_2, {}_1{\sim}_2\}$)

- Decidability of DataSat(1-Loc-dFO, 2) is an open problem

Previous results [FSTTCS'21]

Theorem

DataSat(2-Loc-dFO, 2) is undecidable

Theorem

DataSat(1-Loc-dFO, 2) is decidable
(with relations in $\{{}_1{\sim}_1, {}_2{\sim}_2, {}_1{\sim}_2\}$)

- Decidability of DataSat(1-Loc-dFO, 2) is an open problem

THIS TALK: Find other decidable fragments

The existential fragment

- Σ finite set of unary relation symbols
- $D > 0$ an integer

Definition

For a radius $r \in \mathbb{N}$, the logic $\exists$-r-Loc-dFO$[\Sigma, D]$ is given as follows:

$$\varphi \ ::= \ \langle\!\langle \psi \rangle\!\rangle_x^r \ | \ x = y \ | \ x \neq y \ | \ \exists x.\varphi \ | \ \varphi \vee \varphi \ | \ \varphi \wedge \varphi$$

where $\psi \in$ dFO$[\Sigma, D]$.

Remark:

- No restriction on the quantifiers in ψ located in $\langle\!\langle \psi \rangle\!\rangle_x^r$

Decidability results

Preliminary results

- With 0 **data value** (classical decision problem for first order logic) [Böger, Grädel and Gurevich, 1997]

Theorem

DataSat(dFO, 0) is NEXP-complete.

- With 1 **data value** (equivalent satisfiability problem for Hybrid Logic over Kripke structues with equivalent relation) [Mundhenk and Schneider, 2009]

Theorem

DataSat(dFO, 1) is N2EXP-complete

First result

Theorem

DataSat($\exists$-2-Loc-dFO$, 2)$ is $\mathrm{N2EXP}$-complete.

First result

Theorem

DataSat($\exists$-2-Loc-dFO, 2) is $\mathrm{N2EXP}$-complete.

Proof.

Reduction to DataSat(dFO, 1):

First result

Theorem

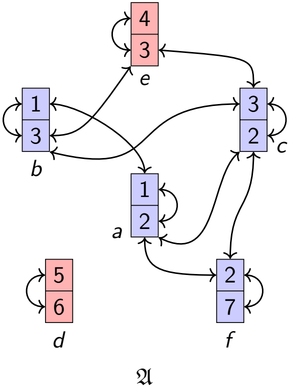
DataSat($\exists$-2-Loc-dFO, 2) is $\mathrm{N2EXP}$-complete.

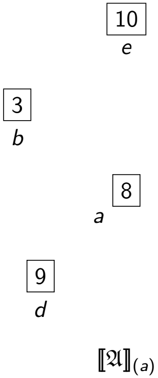
Proof.

Reduction to DataSat(dFO, 1):

- Take a formula in prenex normal form
$$\exists x_1 \ldots \exists x_n . \varphi_{qf}(x_1, \ldots, x_n)$$
- Take n elements $a_1, \ldots, a_n$ of the structures
- Encode the relation with the data of a_i's by unary predicates
- Keep only the data in the views of a_i's not in relation with a_i's (at most 1 per element because we have 2-views)
- Ensure 1-data structures are well-formed □

From 2-data structure to 1-data structure





$P_{a[1,1]} = \{a, b\}$
$P_{a[2,2]} = \{a, c\}$
$P_{a[1,2]} = \emptyset$
$P_{a[2,1]} = \{f\}$

- Each element in $B_2^{\mathfrak{A}}(a)$ shares at least one data with a
- $P_{a[i,j]}$: the element has its j-th data equals to the i-th data of a

Translating the formula $\varphi_{qf}(x_1, \ldots, x_n)$

- We want to translate $\varphi_{qf}(x_1, \ldots, x_n)$ into $[\![\varphi_{qf}]\!](x_1, \ldots, x_n)$ of $\mathrm{dFO}[\Sigma', 1]$
- Main issue: formulas of the form $y \;{}_j{\sim}_k\; z$

Trick to solve this case when located in the subformula $\langle\!\langle \psi \rangle\!\rangle^2_x$:

Translating the formula $\varphi_{qf}(x_1, \ldots, x_n)$

- We want to translate $\varphi_{qf}(x_1, \ldots, x_n)$ into $[\![\varphi_{qf}]\!](x_1, \ldots, x_n)$ of $\mathrm{dFO}[\Sigma', 1]$
- Main issue: formulas of the form $y \;_j\!\sim_k\; z$

Trick to solve this case when located in the subformula $\langle\!\langle \psi \rangle\!\rangle_x^2$:

1. the j-th data of y is **in the radius 1 ball** around x:
   - y have to be labeled with $P_{a[i,j]}$ and z with $P_{a[i,k]}$ for $i \in \{1, 2\}$
2. the j-th data of y is **in the radius 2** ball but **not in the radius 1 ball** around x:
   - y and z have the same data in the translated data structure
3. the j-th data of y is **not in the radius 2 ball** around x:
   - $y \;_j\!\sim_k\; z$ cannot hold in $\langle\!\langle \psi \rangle\!\rangle_x^2$

Translating the formula $\varphi_{qf}(x_1, \ldots, x_n)$

- We want to translate $\varphi_{qf}(x_1, \ldots, x_n)$ into $[\![\varphi_{qf}]\!](x_1, \ldots, x_n)$ of $\mathrm{dFO}[\Sigma', 1]$
- Main issue: formulas of the form $y \;{}_j\!\sim_k\; z$

Trick to solve this case when located in the subformula $\langle\!\langle\psi\rangle\!\rangle^2_x$:

1. the j-th data of y is **in the radius 1 ball** around x:
   - y have to be labeled with $P_{a[i,j]}$ and z with $P_{a[i,k]}$ for $i \in \{1, 2\}$
2. the j-th data of y is **in the radius 2** ball but **not in the radius 1 ball** around x:
   - y and z have the same data in the translated data structure
3. the j-th data of y is **not in the radius 2 ball** around x:
   - $y \;{}_j\!\sim_k\; z$ cannot hold in $\langle\!\langle\psi\rangle\!\rangle^2_x$

- We obtain finally:
  $\mathfrak{A} \models \varphi_{qf}(a_1, \ldots, a_n)$ **iff** $[\![\mathfrak{A}]\!]_{(a_1,\ldots,a_n)} \models [\![\varphi_{qf}]\!](a_1, \ldots, a_n)$

Ensuring 1-data structure are well-formed

- To finish we need to find a 1-data structure $\mathfrak{B}$ such that
  1. $\mathfrak{B}$ has n elements $a_1, \ldots, a_n$
  2. $\mathfrak{B} \models [\![\varphi_{qf}]\!](a_1, \ldots, a_n)$
  3. $\mathfrak{B} = [\![\mathfrak{A}]\!]_{(a_1,\ldots,a_n)}$ for some 2-data structure $\mathfrak{A}$.
- For this last point, we ensure with formulas in dFO$[\Sigma', 1]$ that
  - The labeling by $P_{a_k[i,j]}$ is consistent
  - If a node is is not labelled by any $P_{a_k[i,j]}$ its value is unique
  - The same holds for elements labelled by $P_{a_k[i,1]}$ and $P_{a_\ell[j,2]}$
- This gives rise to a formula $\varphi_{wf}(x_1, \ldots, x_n)$
- Finally, $\exists x_1. \ldots .\exists x_n.\varphi_{qf}(x_1, \ldots, x_n)$ **is satisfiable iff** $\exists x_1. \ldots .\exists x_n.[\![\varphi_{qf}]\!](x_1, \ldots, x_n) \wedge \varphi_{wf}(x_1, \ldots, x_n)$ **is satisfiable**

Other result

Theorem

For all $D \geq 1$, DataSat($\exists$-1-Loc-dFO, D) is NEXP-complete.

Other result

Theorem

For all $D \geq 1$, DataSat($\exists$-1-Loc-dFO, D) is NEXP-complete.

Proof.

Reduction to DataSat(dFO, 0):

Other result

Theorem

For all $D \geq 1$, DataSat($\exists$-1-Loc-dFO, D) is NEXP-complete.

Proof.

Reduction to DataSat(dFO, 0):

- Take a formula in prenex normal form $\exists x_1. \ldots \exists x_n. \varphi_{qf}(x_1, \ldots, x_n)$
- Take n elements $a_1, \ldots, a_n$ of the structures
- Encode the relation with the data of a_i's by unary predicates
- ~~Keep only the data in the views of a_i's not in relation with a_i's (at most 1 per element because we have 2-neighborhoods)~~
- Ensure 0-data structures are well-formed □

Undecidability results

Radius 3 and two data values

Theorem

DataSat($\exists$-3-Loc-dFO$, 2$) is undecidable.

Radius 3 and two data values

Theorem

DataSat($\exists$-3-Loc-dFO, 2) is undecidable.

Proof.

- Reduction from DataSat(dFO, 2)
- We can translate any 2-data structure $\mathfrak{A}$ into another one $\mathfrak{B}$ such that:
  - In $\mathfrak{B}$, all the elements are located in the radius 3 balls of any elements
  - To obtain $\mathfrak{B}$ from $\mathfrak{A}$: add elements of the form (v, v') where v and v' are data present in $\mathfrak{A}$
  - Label these new elements with a specific predicate
  - Lastly use that in $\langle\langle\psi\rangle\rangle_x^3$, there is no restriction of ψ in the existential fragment

□

Radius 2 and three data values

Theorem

DataSat($\exists$-2-Loc-dFO, 3) is undecidable.

Radius 2 and three data values

Theorem

DataSat(∃-2-Loc-dFO, 3) is undecidable.

Proof.

- Reduction from DataSat(dFO, 2)
- We can translate any 2-data structure $\mathfrak{A}$ into a 3-data structure $\mathfrak{B}$ such that:
  - In $\mathfrak{B}$, all the elements are located in the radius 2 balls of any elements
  - To obtain $\mathfrak{B}$ from $\mathfrak{A}$, we add a third value equal to 0 to all elements

□

Conclusion and Future Works

Satisfiability of the existential fragments:

Radius \ Data	1	2	3	4	…
⋮	⋮	⋮	⋮	⋮	
4	✓	✗	✗	✗	…
3	✓	✗	✗	✗	…
2	✓	✓	✗	✗	…
1	✓	✓	✓	✓	…

Perspectives:

- Study other fragments with quantifiers restrictions
- Develop approximation methods

Thank you!