hypothetical answers to continuous queries over data streams

luís cruz-filipe[1]
(joint work with graça gaspar[2] & isabel nunes[2])

[1]department of mathematics and computer science
university of southern denmark

[2]department of informatics
faculty of sciences, university of lisbon

days in logic
january 30th, 2020

Outline

1. introduction
2. denotational semantics
3. operational semantics
4. negation
5. conclusions

the context

continuous queries over data streams

- modern-day distributed systems
- information pouring in from e.g. sensors
- queries need to be answered in real-time
- answers are output as information arrives

the context

continuous queries over data streams

- modern-day distributed systems
- information pouring in from e.g. sensors
- queries need to be answered in real-time
- answers are output as information arrives

several models

common approach: rule-based reasoning

- usually based on variants of datalog
- set of facts dynamically obtained from a data stream D
- common problems: blocking queries, unbound wait

current contribution

online algorithm with offline pre-processing outputting partial information

- information that an answer may be output in the future
- fundamentation for such hypothetical answers

current contribution

online algorithm with offline pre-processing outputting partial information

- information that an answer may be output in the future
- fundamentation for such hypothetical answers

practical relevance

partial information allows for preventive measures to be taken

- an action might be required $\rightsquigarrow$ maybe prepare for it
- a failure might occur $\rightsquigarrow$ steps may be taken to prevent it

the justification for *why* the hypothetical answer is output can be used to evaluate its likelihood

detecting malfunctions in wind turbines

$$\mathsf{Temp}(X, \mathsf{high}, T) \rightarrow \mathsf{Flag}(X, T)$$
$$\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Cool}(X, T+1)$$
$$\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Shdn}(X, T+1)$$
$$\mathsf{Shdn}(X, T) \rightarrow \mathsf{Malf}(X, T-2)$$

- a data center managing a set of wind turbines receives temperature readings $\mathsf{Temp}(\mathit{Device}, \mathit{Level}, \mathit{Time})$ from sensors in each turbine
- the data centre tracks activation of cooling measures in each turbine, recording malfunctions and shutdowns by means of a program in temporal datalog

detecting malfunctions in wind turbines

$$\mathsf{Temp}(X, \mathsf{high}, T) \to \mathsf{Flag}(X, T)$$
$$\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) \to \mathsf{Cool}(X, T+1)$$
$$\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) \to \mathsf{Shdn}(X, T+1)$$
$$\mathsf{Shdn}(X, T) \to \mathsf{Malf}(X, T-2)$$

query: $Q = \mathsf{Malf}(X, T)$

if:

$$\mathsf{Temp}(\mathsf{wt25}, \mathsf{high}, i) \qquad i = 0, 1, 2$$

all arrive at the data stream, then $\{X := \mathsf{wt25}, T := 0\}$ is an answer to Q

detecting malfunctions in wind turbines

$$\mathsf{Temp}(X, \mathsf{high}, T) \rightarrow \mathsf{Flag}(X, T)$$
$$\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Cool}(X, T+1)$$
$$\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Shdn}(X, T+1)$$
$$\mathsf{Shdn}(X, T) \rightarrow \mathsf{Malf}(X, T-2)$$

query: $Q = \mathsf{Malf}(X, T)$

but: once

$$\mathsf{Temp}(\mathsf{wt25}, \mathsf{high}, 0)$$

arrives, we already know that $\{X := \mathsf{wt25}, T := 0\}$ *might* become an answer to Q

detecting malfunctions in wind turbines

$$\mathsf{Temp}(X, \mathsf{high}, T) \rightarrow \mathsf{Flag}(X, T)$$
$$\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Cool}(X, T+1)$$
$$\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Shdn}(X, T+1)$$
$$\mathsf{Shdn}(X, T) \rightarrow \mathsf{Malf}(X, T-2)$$

query: $Q = \mathsf{Malf}(X, T)$

and since

$$\mathsf{Temp}(\mathsf{wt42}, \mathsf{high}, 0)$$

does *not* arrive, we know that $\{X := \mathsf{wt42}, T := 0\}$ *cannot* become an answer to Q

detecting malfunctions in wind turbines

$$\mathsf{Temp}(X, \mathsf{high}, T) \rightarrow \mathsf{Flag}(X, T)$$
$$\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Cool}(X, T+1)$$
$$\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Shdn}(X, T+1)$$
$$\mathsf{Shdn}(X, T) \rightarrow \mathsf{Malf}(X, T-2)$$

assumption

we assume that the data stream D is complete at each time point, i.e. at time t it contains all facts with timestamps $\leq t$

we call this set of facts the τ-*history* D_τ

Outline

extensional predicates

we assume that the predicate symbols occurring in D do not appear in heads of rules in Π – these are *extensional* predicates

hypothetical answers

a *hypothetical answer* to a query Q over a program Π and a history D_τ is a pair $\langle \theta, H \rangle$, where θ is a substitution and H is a finite set of ground extensional atoms (the hypotheses) such that:

- θ only instantiates variables free in Q
- H only contains atoms with time stamp $\tau' > \tau$
- $\Pi \cup D_\tau \cup H \models Q\theta$
- H is minimal with respect to set inclusion

our example program

$$
\begin{aligned}
\mathsf{Temp}(X, \mathsf{high}, T) &\rightarrow \mathsf{Flag}(X, T) \\
\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) &\rightarrow \mathsf{Cool}(X, T+1) \\
\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) &\rightarrow \mathsf{Shdn}(X, T+1) \\
\mathsf{Shdn}(X, T) &\rightarrow \mathsf{Malf}(X, T-2)
\end{aligned}
$$

query

$Q = \mathsf{Malf}(X, T)$

our example program

$$\mathsf{Temp}(X, \mathsf{high}, T) \rightarrow \mathsf{Flag}(X, T)$$
$$\mathsf{Flag}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Cool}(X, T+1)$$
$$\mathsf{Cool}(X, T) \wedge \mathsf{Flag}(X, T+1) \rightarrow \mathsf{Shdn}(X, T+1)$$
$$\mathsf{Shdn}(X, T) \rightarrow \mathsf{Malf}(X, T-2)$$

query

$Q = \mathsf{Malf}(X, T)$

$\mathsf{Temp}(\mathsf{wt25}, \mathsf{high}, 0) \in D_0$

$\langle\{X := \mathsf{wt25}, T := 0\}, H\rangle$ is a hypothetical answer to Q for
$H = \{\mathsf{Temp}(\mathsf{wt25}, \mathsf{high}, i) \mid i = 1, 2\}$

our example program

$$\begin{aligned} \mathsf{Temp}(X, \mathsf{high}, T) &\rightarrow \mathsf{Flag}(X, T) \\ \mathsf{Flag}(X, T) \land \mathsf{Flag}(X, T+1) &\rightarrow \mathsf{Cool}(X, T+1) \\ \mathsf{Cool}(X, T) \land \mathsf{Flag}(X, T+1) &\rightarrow \mathsf{Shdn}(X, T+1) \\ \mathsf{Shdn}(X, T) &\rightarrow \mathsf{Malf}(X, T-2) \end{aligned}$$

query

$Q = \mathsf{Malf}(X, T)$

$\mathsf{Temp}(\mathsf{wt42}, \mathsf{high}, 0) \notin D_0$

$\langle\{X := \mathsf{wt42}, T := 0\}, H\rangle$ is not a hypothetical answer to Q for any H

supported answers

- a non-empty set of facts $E \subseteq D_\tau$ is *evidence* supporting a hypothetical answer $\langle\theta, H\rangle$ if E is a minimal set s.t. $\Pi \cup E \cup H \models P\theta$
- a *supported answer* to Q over D_τ is a triple $\langle\theta, H, E\rangle$ where E is evidence supporting $\langle\theta, H\rangle$

supported answers

- a non-empty set of facts $E \subseteq D_\tau$ is *evidence* supporting a hypothetical answer $\langle \theta, H \rangle$ if E is a minimal set s.t. $\Pi \cup E \cup H \models P\theta$
- a *supported answer* to Q over D_τ is a triple $\langle \theta, H, E \rangle$ where E is evidence supporting $\langle \theta, H \rangle$

in our example program

the fact

$$\mathsf{Temp}(\mathsf{wt25}, \mathsf{high}, 0) \in D_0$$

is evidence that $\langle \{X := \mathsf{wt25}, T := 0\}, H \rangle$ is a hypothetical answer to Q for

$$H = \{\mathsf{Temp}(\mathsf{wt25}, \mathsf{high}, i) \mid i = 1, 2\}$$

Outline

future atom

an atom $P(t_1, \ldots, t_n)$ is a *future atom wrt* τ if P is a temporal predicate and the time term t_n either contains a temporal variable or is a time instant $t_n > \tau$

future atom

an atom $P(t_1, \ldots, t_n)$ is a *future atom wrt* τ if P is a temporal predicate and the time term t_n either contains a temporal variable or is a time instant $t_n > \tau$

sld-refutation, revisited

an *sld-refutation* *with future premises* of Π and Q over D_τ is a finite sld-derivation of $P \cup D_\tau \cup \{\neg Q\}$ whose last goal only contains extensional future atoms wrt τ

future atom

an atom $P(t_1, \ldots, t_n)$ is a *future atom wrt* τ if P is a temporal predicate and the time term t_n either contains a temporal variable or is a time instant $t_n > \tau$

sld-refutation, revisited

an *sld-refutation with future premises* of Π and Q over D_τ is a finite sld-derivation of $P \cup D_\tau \cup \{\neg Q\}$ whose last goal only contains extensional future atoms wrt τ

computed answer with premises

if $\mathcal{D}$ is an sld-refutation with future premises of Q over D_τ with last goal $G = \neg \bigwedge_i \alpha_i$ and θ is the restriction of the composition of the substitutions in $\mathcal{D}$ to var(Q), then $\langle \theta, \wedge_i \alpha_i \rangle$ is a *computed answer with premises* to Q over D_τ

independence of the computation rule

from classical results about sld-resolution, we can reorder the steps of any sld-refutation with future premises to use the facts from D_τ in temporal order

independence of the computation rule

from classical results about sld-resolution, we can reorder the steps of any sld-refutation with future premises to use the facts from D_τ in temporal order

key idea

this simple observation gives us an incremental algorithm

- at each step, update any "ongoing" derivations with the new facts
- any derivations expecting facts that did not arrive are forgotten
- some pre-processing allows us to identify relevant facts

a two-stage algorithm

pre-processing step

we compute answers with premises to Q over D_{-1}

- we store the minimal answers wrt set inclusion in a set $\mathcal{P}_Q$
- we initialize the set $\mathcal{S}_{-1}$ of *schematic supported answers* to $\emptyset$

a two-stage algorithm

pre-processing step

we compute answers with premises to Q over D_{-1}

- we store the minimal answers wrt set inclusion in a set $\mathcal{P}_Q$
- we initialize the set $\mathcal{S}_{-1}$ of *schematic supported answers* to $\emptyset$

online step

to compute $\mathcal{S}_{\tau+1}$ from $\mathcal{S}_\tau$ and $D_{\tau+1} \setminus D_\tau$:

- for each answer in $\mathcal{P}_Q$, we perform sld-resolution between its set of elements with minimal timestamps and $D_{\tau+1} \setminus D_\tau$
- for each element of $\mathcal{S}_\tau$, we perform sld-resolution between its set of elements with timestamp $\tau + 1$ and $D_{\tau+1} \setminus D_\tau$

each refutation yields an element in $\mathcal{S}_{\tau+1}$

termination (i)

under suitable assumptions, the pre-processing step terminates

termination (i)

under suitable assumptions, the pre-processing step terminates

termination (ii)

the online step terminates in polynomial time in the size of $\mathcal{S}_\tau$, $\mathcal{P}_Q$ and $D_{\tau+1} \setminus D_\tau$

termination (i)

under suitable assumptions, the pre-processing step terminates

termination (ii)

the online step terminates in polynomial time in the size of $\mathcal{S}_\tau$, $\mathcal{P}_Q$ and $D_{\tau+1} \setminus D_\tau$

soundness

every instantiation of an element of $\mathcal{S}_\tau$ is a supported answer to Q over Π and D_τ

termination (i)

under suitable assumptions, the pre-processing step terminates

termination (ii)

the online step terminates in polynomial time in the size of $\mathcal{S}_\tau$, $\mathcal{P}_Q$ and $D_{\tau+1} \setminus D_\tau$

soundness

every instantiation of an element of $\mathcal{S}_\tau$ is a supported answer to Q over Π and D_τ

completeness

every supported answer to Q over Π and D_τ is an instantiation of an element of $\mathcal{S}_\tau$

Outline

safe negation

we can add (safe) negation in the usual way to our framework

- most results go through (but complexity increases)

safe negation

we can add (safe) negation in the usual way to our framework

- most results go through (but complexity increases)

stratification

stronger results in presence of stratified negation

- more complex notion
- possibly infinitely many strata
- not necessarily temporally ordered

our contribution

- new notion of stratification
- decision procedure + algorithm returning a finite representation of the strata

our contribution

- new notion of stratification
- decision procedure + algorithm returning a finite representation of the strata

results

- fixed-parameter tractability for the online step
- soundness and completeness wrt well-founded semantics (wip)

Outline

main achievements

our contribution

- denotational semantics for hypothetical answers
- notion of evidence for hypothetical answers
- operational semantics based on sld-resolution
- online algorithm with offline pre-processing outputting partial information
- parallel computation of answers (bypasses some usual problems)
- more expressive negation in the language
- new (decidable) notion of stratification with computable strata

future work

- an implementation. . .

thank you!