Fast and Simple Relational Processing of Uncertain Data

Lyublena Antova (Cornell), Thomas Jansen (SAP), Christoph Koch (Cornell), Dan Olteanu (Oxford)
We want to enter the information from forms like these into a database.

- What is the marital status of the first resp. the second person?
- What are the social security numbers? 185? 186? 785?
Much of the available information cannot be represented and is lost, e.g. Smith’s SSN is either 185 or 785; Brown’s SSN is either 185 or 186.

- Data cleaning: No two distinct persons can have the same SSN.
Main goals of the MayBMS project

Create a scalable DBMS for uncertain/probabilistic data

1. Representation and storage mechanisms
2. Uncertainty-aware query and data manipulation language
3. Efficient processing techniques for queries and constraints

This talk will cover some aspects of (1) and (3).
Representation of uncertain data
Desiderata for a representation system

1. **Succinctness/Space-efficient storage**
   - Large number of independent *local* alternatives, which multiply up to a very large number of worlds.

2. **Efficient real-world query processing**
   - Tradeoff between succinctness and complexity of query evaluation. We want to do well in practice.

3. **Expressiveness/Representability**
   - Ability to represent all results of query and constraint processing.
   - Constraints/queries enforce dependencies across alternatives!
Quest for well-behaved representation system (1)

Properties (ICDE’07, ICDT’07)

- Relational representation of uncertainty at attribute-level
- Complete in the case of finite sets of alternatives (worlds)
- Data independence naturally supported by relational product
  Decompositions via efficient prime factorization of relations
Equivalent column-oriented encoding with one relation per each attribute of $R$. 

$$U_{R[SSN]}$$

<table>
<thead>
<tr>
<th>V $\mapsto$ D</th>
<th>TID</th>
<th>SSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x \mapsto 1$</td>
<td>$t_1$</td>
<td>185</td>
</tr>
<tr>
<td>$x \mapsto 2$</td>
<td>$t_1$</td>
<td>785</td>
</tr>
<tr>
<td>$y \mapsto 1$</td>
<td>$t_2$</td>
<td>185</td>
</tr>
<tr>
<td>$y \mapsto 2$</td>
<td>$t_2$</td>
<td>186</td>
</tr>
</tbody>
</table>

$$U_{R[N]}$$

<table>
<thead>
<tr>
<th>V $\mapsto$ D</th>
<th>TID</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v \mapsto 1$</td>
<td>$t_1$</td>
<td>Smith</td>
</tr>
<tr>
<td>$v \mapsto 2$</td>
<td>$t_1$</td>
<td>Brown</td>
</tr>
<tr>
<td>$w \mapsto 1$</td>
<td>$t_2$</td>
<td>1</td>
</tr>
<tr>
<td>$w \mapsto 2$</td>
<td>$t_2$</td>
<td>2</td>
</tr>
<tr>
<td>$w \mapsto 3$</td>
<td>$t_2$</td>
<td>3</td>
</tr>
<tr>
<td>$w \mapsto 4$</td>
<td>$t_2$</td>
<td>4</td>
</tr>
</tbody>
</table>
U-Relational Databases

<table>
<thead>
<tr>
<th>$U_{R[SSN]}$</th>
<th>$V \leftrightarrow D$</th>
<th>TID</th>
<th>SSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x \leftrightarrow 1$</td>
<td>$t_1$</td>
<td></td>
<td>185</td>
</tr>
<tr>
<td>$x \leftrightarrow 2$</td>
<td>$t_1$</td>
<td></td>
<td>785</td>
</tr>
<tr>
<td>$y \leftrightarrow 1$</td>
<td>$t_2$</td>
<td></td>
<td>185</td>
</tr>
<tr>
<td>$y \leftrightarrow 2$</td>
<td>$t_2$</td>
<td></td>
<td>186</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$U_{R[N]}$</th>
<th>TID</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td></td>
<td>Smith</td>
</tr>
<tr>
<td>$t_2$</td>
<td></td>
<td>Brown</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$W$</th>
<th>$V \leftrightarrow D$</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x \leftrightarrow 1$</td>
<td>.4</td>
<td></td>
</tr>
<tr>
<td>$x \leftrightarrow 2$</td>
<td>.6</td>
<td></td>
</tr>
<tr>
<td>$y \leftrightarrow 1$</td>
<td>.7</td>
<td></td>
</tr>
<tr>
<td>$y \leftrightarrow 2$</td>
<td>.3</td>
<td></td>
</tr>
<tr>
<td>$v \leftrightarrow 1$</td>
<td>.8</td>
<td></td>
</tr>
<tr>
<td>$v \leftrightarrow 2$</td>
<td>.2</td>
<td></td>
</tr>
<tr>
<td>$w \leftrightarrow 1$</td>
<td>.25</td>
<td></td>
</tr>
<tr>
<td>$w \leftrightarrow 2$</td>
<td>.25</td>
<td></td>
</tr>
<tr>
<td>$w \leftrightarrow 3$</td>
<td>.25</td>
<td></td>
</tr>
<tr>
<td>$w \leftrightarrow 4$</td>
<td>.25</td>
<td></td>
</tr>
</tbody>
</table>

- Discrete independent (random) variables ($x, y, v, w$).
- Representation: U-relations + table $W$ representing distributions.
- The schema of each U-relation consists of
  - a tuple id column,
  - a set of column pairs ($V_i, D_i$) representing variable assignments, and
  - a set of value columns.
Semantics of U-Relational Databases

- Each possible world is identified by a valuation $\theta$ that assigns one of the possible values to each variable.
- The probability of the possible world is the product of weights of the values of the variables.
- The value-component of a tuple of a U-relation is in a given possible world if its variable assignments are consistent with $\theta$.
- Attribute-level uncertainty through vertical decomposition.
We choose possible world \{x \mapsto 1, y \mapsto 2, v \mapsto 1, w \mapsto 1\}. 
Semantics of U-Relational Databases

- We choose possible world \(\{x \mapsto 1, y \mapsto 2, v \mapsto 1, w \mapsto 1\}\).
- Probability weight of this world: \(0.4 \times 0.3 \times 0.8 \times 0.25 = 0.024\).
- Now we have a vertically decomposed version of the chosen possible world.
Properties of U-Relational Databases

- Complete representation system for finite sets of possible worlds
  - MystiQ: independent tuples/block-independent disjoint tables
- Often exponentially more succinct than WSDs, ULDBs, prob. databases
- A special case of c-tables
  - like all other existing representation formalisms, BUT...
- Purely relational representation of uncertainty at attribute-level
  - in contrast to probabilistic databases of MystiQ and ULDBs of Trio
- Efficient relational evaluation of many query operators (next topic)
Efficient query evaluation
Positive relational algebra

Query evaluation under *possible world semantics*:

For any positive relational algebra query $q$ over any U-relational database $T$, there exists a positive relational algebra query $\overline{q}$ of polynomial size such that

$$\overline{q}(T) = rep^{-1}(\{q(A_i) \mid A_i \in rep(T)\}).$$

Properties

- relational evaluation using the query plan of your choice
- PTIME data complexity
- preserves the provenance of answer tuples
Query Evaluation: Example

Names of possibly married persons: \( \text{possible}(\pi_{\text{Name}}(\sigma_{\text{Status}=2}(S))) \)

\[
\begin{array}{|c|c|c|}
\hline
\text{U}_{S[\text{Name}]} & \text{V} \leftrightarrow \text{D} & \text{TID} & \text{Name} \\
\hline
x_3 & 1 & t_1 & \text{Smith} \\
x_5 & 1 & t_2 & \text{Brown} \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|}
\hline
\text{U}_{S[\text{Status}]} & \text{V} \leftrightarrow \text{D} & \text{TID} & \text{Status} \\
\hline
x_3 & 1 & t_1 & 1 \\
x_3 & 2 & t_1 & 2 \\
x_6 & 1 & t_2 & 1 \\
x_6 & 2 & t_2 & 2 \\
\hline
\end{array}
\]

Evaluation steps:

1. merge the U-relations storing the necessary columns:
   \[
   Q := \text{possible}(\pi_{\text{Name}}(\sigma_{\text{Status}=2}(\text{merge}(\pi_{\text{Name}}(S), \pi_{\text{Status}}(S)))))
   \]
Query Evaluation: Example

Names of possibly married persons: \( \text{possible}(\pi_{\text{Name}}(\sigma_{\text{Status}=2}(S))) \)

<table>
<thead>
<tr>
<th>( U_S[\text{Name}] )</th>
<th>V ↔ D</th>
<th>TID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_3 \leftrightarrow 1 )</td>
<td>( t_1 )</td>
<td>Smith</td>
<td></td>
</tr>
<tr>
<td>( x_5 \leftrightarrow 1 )</td>
<td>( t_2 )</td>
<td>Brown</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( U_S[\text{Status}] )</th>
<th>V ↔ D</th>
<th>TID</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_3 \leftrightarrow 1 )</td>
<td>( t_1 )</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>( x_3 \leftrightarrow 2 )</td>
<td>( t_1 )</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>( x_6 \leftrightarrow 1 )</td>
<td>( t_2 )</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>( x_6 \leftrightarrow 2 )</td>
<td>( t_2 )</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Evaluation steps:

1. merge the U-relations storing the necessary columns:
   \[
   Q := \text{possible}(\pi_{\text{Name}}(\sigma_{\text{Status}=2}(\text{merge}(\pi_{\text{Name}}(S), \pi_{\text{Status}}(S)))))
   \]

2. rewrite \( Q \) on column-store:
   \[
   P := \pi_{\text{Name}}(\sigma_{\text{Status}=2}(U_S[\text{Name}] \bowtie_{\psi \land \phi} U_S[\text{Status}])), \text{ where}
   \]
   \[
   \psi \text{ ensures that we only generate tuples that occur in some worlds:}
   \]
   \[
   \psi := (U_S[\text{Name}] \cdot V = U_S[\text{Status}] \cdot V \Rightarrow U_S[\text{Name}] \cdot D = U_S[\text{Status}] \cdot D),
   \]
   \[
   \phi \text{ ensures that we only merge valid tuples:}
   \]
   \[
   \phi := (U_S[\text{Name}] \cdot TID = U_S[\text{Status}] \cdot TID)
   \]
Query Evaluation: Example

Names of possibly married persons:  \( \text{possible}(\pi_{\text{Name}}(\sigma_{\text{Status}=2}(S))) \)

Evaluation steps:

1. merge the \( U \)-relations storing the necessary columns:
   \[
   Q := \text{possible}\left(\pi_{\text{Name}}(\sigma_{\text{Status}=2}\left(\text{merge}\left(\pi_{\text{Name}}(S), \pi_{\text{Status}}(S)\right)\right))\right)
   \]

2. rewrite \( Q \) on column-store:
   \[
   P := \pi_{\text{Name}}(\sigma_{\text{Status}=2}(U_{S[\text{Name}] \bowtie_{\psi \land \phi} U_{S[\text{Status}]}))), \text{ where}
   \]
   \[
   \psi \text{ ensures that we only generate tuples that occur in some worlds:} \\
   \psi := (U_{S[\text{Name}]}.V = U_{S[\text{Status}]}.V \Rightarrow U_{S[\text{Name}]}.D = U_{S[\text{Status}]}.D),
   \]
   \[
   \phi \text{ ensures that we only merge valid tuples:} \\
   \phi := (U_{S[\text{Name}]}.TID = U_{S[\text{Status}]}.TID)
   \]

3. feed \( P \) to any relational query optimizer

<table>
<thead>
<tr>
<th>( U_{S[\text{Name}]} )</th>
<th>( V \leftrightarrow D )</th>
<th>TID</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_3 \leftrightarrow 1 )</td>
<td>( t_1 )</td>
<td>Smith</td>
<td></td>
</tr>
<tr>
<td>( x_5 \leftrightarrow 1 )</td>
<td>( t_2 )</td>
<td>Brown</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( U_{S[\text{Status}]} )</th>
<th>( V \leftrightarrow D )</th>
<th>TID</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_3 \leftrightarrow 1 )</td>
<td>( t_1 )</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>( x_3 \leftrightarrow 2 )</td>
<td>( t_1 )</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>( x_6 \leftrightarrow 1 )</td>
<td>( t_2 )</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>( x_6 \leftrightarrow 2 )</td>
<td>( t_2 )</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
Query Evaluation: Example

Names of possibly married persons: \( \textit{possible}(\pi_{\text{Name}}(\sigma_{\text{Status}=2}(S))) \)

\[
\begin{array}{c|c|c|c}
U_{S[Name]} & V \leftrightarrow D & TID & \text{Name} \\
\hline
x_3 \leftrightarrow 1 & t_1 & & Smith \\
x_5 \leftrightarrow 1 & t_2 & & Brown \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
U_{S[Status]} & V \leftrightarrow D & TID & \text{Status} \\
\hline
x_3 \leftrightarrow 1 & t_1 & & 1 \\
x_3 \leftrightarrow 2 & t_1 & & 2 \\
x_6 \leftrightarrow 1 & t_2 & & 1 \\
x_6 \leftrightarrow 2 & t_2 & & 2 \\
\end{array}
\]

\[
\begin{array}{c|c|c|c|c|c}
& V_1 \leftrightarrow D_1 & V_2 \leftrightarrow D_2 & \text{TID} & \text{Name} & \text{Status} \\
\hline
\text{wrong Status} & x_3 \leftrightarrow 1 & x_3 \leftrightarrow 1 & t_1 \equiv t_1 & \text{Smith} & 1 \\
\text{inconsistent} & x_3 \leftrightarrow 1 & x_3 \leftrightarrow 2 & t_1 \equiv t_1 & \text{Smith} & 2 \\
\text{wrong TIDs} & x_3 \leftrightarrow 1 & x_6 \leftrightarrow 1 & t_1 \equiv t_2 & \text{Smith} & 1 \\
\text{wrong TIDs} & x_3 \leftrightarrow 1 & x_6 \leftrightarrow 2 & t_1 \equiv t_2 & \text{Smith} & 2 \\
\text{wrong TIDs} & x_5 \leftrightarrow 1 & x_3 \leftrightarrow 1 & t_1 \equiv t_2 & \text{Brown} & 1 \\
\text{wrong TIDs} & x_5 \leftrightarrow 1 & x_3 \leftrightarrow 2 & t_1 \equiv t_2 & \text{Brown} & 2 \\
\text{wrong TIDs} & x_5 \leftrightarrow 1 & x_6 \leftrightarrow 1 & t_2 \equiv t_2 & \text{Brown} & 1 \\
\text{wrong Status} & x_5 \leftrightarrow 1 & x_6 \leftrightarrow 2 & t_2 \equiv t_2 & \text{Brown} & 2 \\
\end{array}
\]
Query Evaluation: Example

Names of possibly married persons: \( \text{possible}(\pi_{Name}(\sigma_{Status=2}(S))) \)

\[
\begin{array}{c|c|c|c}
U_{S}\{Name\} & V \mapsto D & TID & Name \\
\hline
x_3 \mapsto 1 & t_1 & Smith \\
x_5 \mapsto 1 & t_2 & Brown \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
U_{S}\{Status\} & V \mapsto D & TID & Status \\
\hline
x_3 \mapsto 1 & t_1 & 1 \\
x_3 \mapsto 2 & t_1 & 2 \\
x_6 \mapsto 1 & t_2 & 1 \\
x_6 \mapsto 2 & t_2 & 2 \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
V_1 \mapsto D_1 & V_2 \mapsto D_2 & TID & Name \mapsto \text{Status} \\
\hline
x_5 \mapsto 1 & x_6 \mapsto 2 & t_2 & Brown \mapsto 2 \\
\end{array}
\]
Beyond positive relational algebra

Difference
Tuple q-possibility is NP-hard even for normalized tuple-level U-relations and queries with difference. BUT this is already true for Codd tables.

World-set Algebra [SIGMOD’07,VLDB’07]

- **Possible** \((R)\)
  Implemented using projection

- **Certain** \((R)\)
  Implemented using division for *normalized* tuple-level U-relations (normalization = at most one variable assignment per tuple)

- **repair-key\(\vec{A}@P\)(R)**
  Turns a possible world into the set of worlds consisting of all possible maximal repairs of key \(\vec{A}\) in \(R\).

- **conf** \((R)\)
  Computes the exact confidence of (distinct) tuples

...
repair-key example

Tossing a biased coin twice.

\[
\begin{array}{c|ccc}
R & \text{Toss} & \text{Face} & \text{FProb} \\
1 & H & .4 \\
1 & T & .6 \\
2 & H & .4 \\
2 & T & .6 \\
\end{array}
\]

\(Pr = 1\)

\[S := \text{repair-key}_{\text{Toss@FProb}}(R)\] results in four worlds:

\[
\begin{array}{c|ccc}
S^1 & \text{Toss} & \text{Face} & \text{FProb} \\
1 & H & .4 \\
2 & H & .4 \\
\end{array}
\]

\[
\begin{array}{c|ccc}
S^2 & \text{Toss} & \text{Face} & \text{FProb} \\
1 & H & .4 \\
2 & T & .6 \\
\end{array}
\]

\[
\begin{array}{c|ccc}
S^3 & \text{Toss} & \text{Face} & \text{FProb} \\
1 & T & .6 \\
2 & H & .4 \\
\end{array}
\]

\[
\begin{array}{c|ccc}
S^4 & \text{Toss} & \text{Face} & \text{FProb} \\
1 & T & .6 \\
2 & T & .6 \\
\end{array}
\]

\[
Pr(S^1) = 1 \cdot \frac{.4}{.4 + .6} \cdot \frac{.4}{.4 + .6} = .16, \quad Pr(S^2) = Pr(S^3) = .24, \quad Pr(S^4) = .36
\]
repair-key example

Tossing a biased coin twice.

\[
\begin{array}{c|ccc}
R & \text{Toss} & \text{Face} & \text{FProb} \\
\hline
1 & H & .4 \\
1 & T & .6 \\
2 & H & .4 \\
2 & T & .6 \\
\end{array}
\]

\[\text{Pr} = 1\]

\[S := \text{repair-key}_{\text{Toss@FProb}}(R) \text{ is just a projection/copying of columns (even though we may create an exponential number of possible worlds)!}\]
What about probabilities?

Given a tuple $t$ with a set of valuations $S$, compute $\text{conf}(t)$ by partitioning $S$

(a) into independent subsets (exploit contextual independence)

(b) by removing variables (modified Davis-Putnam)

(c) by removing valuations (compute equiv. set of pairwise mutex valuations)

Our current approach is a cost-based interplay of (a)-(c).

More in *Conditioning Probabilistic Databases* by Koch&Olteanu.
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{ \{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\} \} \]
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{\{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\}\} \]
Confidence computation example

\[ S = \{ \{x \mapsto 1\}, \{x \mapsto 2, y \mapsto 1\}, \{x \mapsto 2, z \mapsto 1\}, \{u \mapsto 1, v \mapsto 1\}, \{u \mapsto 2\} \} \]

\[ P(S) = 0.7578. \]
Experiments
Uncertain data generator

- extend TPC-H population generator 2.6 to generate U-relational databases

  any generated world has the sizes of relations and join selectivities of the original TPC-H one-world case

- parameters: scale \((s)\), uncertainty ratio \((x)\), correlation ratio \((z)\), max alternatives per field \((8)\), drop after correlation \((0.25)\)

- correlations follow a pattern obtained by chasing egds on uncertain data [ICDE’07]
## Uncertainty and storage

Total number of worlds, max. number of domain values for a variable (Rng), and size in MB of the U-relational database for each of our settings.

<table>
<thead>
<tr>
<th>s</th>
<th>z</th>
<th>TPC-H dbsize</th>
<th>#worlds</th>
<th>Rng</th>
<th>dbsize</th>
<th>#worlds</th>
<th>Rng</th>
<th>dbsize</th>
<th>#worlds</th>
<th>Rng</th>
<th>dbsize</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>0.1</td>
<td>17</td>
<td>$10^{857.076}$</td>
<td>21</td>
<td>82</td>
<td>$10^{7955.30}$</td>
<td>57</td>
<td>85</td>
<td>$10^{79354.1}$</td>
<td>57</td>
<td>114</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17</td>
<td>$10^{523.031}$</td>
<td>71</td>
<td>82</td>
<td>$10^{4724.56}$</td>
<td>901</td>
<td>88</td>
<td>$10^{46675.6}$</td>
<td>662</td>
<td>139</td>
</tr>
<tr>
<td>0.05</td>
<td>0.1</td>
<td>85</td>
<td>$10^{4287.23}$</td>
<td>22</td>
<td>389</td>
<td>$10^{39913.8}$</td>
<td>33</td>
<td>403</td>
<td>$10^{396137}$</td>
<td>65</td>
<td>547</td>
</tr>
<tr>
<td></td>
<td></td>
<td>85</td>
<td>$10^{2549.14}$</td>
<td>178</td>
<td>390</td>
<td>$10^{23515.5}$</td>
<td>449</td>
<td>416</td>
<td>$10^{232650}$</td>
<td>1155</td>
<td>672</td>
</tr>
<tr>
<td>0.10</td>
<td>0.1</td>
<td>170</td>
<td>$10^{8606.77}$</td>
<td>27</td>
<td>773</td>
<td>$10^{79889.9}$</td>
<td>49</td>
<td>802</td>
<td>$10^{793611}$</td>
<td>53</td>
<td>1090</td>
</tr>
<tr>
<td></td>
<td></td>
<td>170</td>
<td>$10^{5044.65}$</td>
<td>181</td>
<td>776</td>
<td>$10^{46901.8}$</td>
<td>773</td>
<td>826</td>
<td>$10^{466038}$</td>
<td>924</td>
<td>1339</td>
</tr>
<tr>
<td>0.50</td>
<td>0.1</td>
<td>853</td>
<td>$10^{43368.0}$</td>
<td>49</td>
<td>3843</td>
<td>$10^{400185}$</td>
<td>71</td>
<td>3987</td>
<td>$10^{3.97e+06}$</td>
<td>85</td>
<td>5427</td>
</tr>
<tr>
<td></td>
<td></td>
<td>853</td>
<td>$10^{25528.9}$</td>
<td>214</td>
<td>3866</td>
<td>$10^{234840}$</td>
<td>1832</td>
<td>4012</td>
<td>$10^{2.33e+06}$</td>
<td>2586</td>
<td>6682</td>
</tr>
<tr>
<td>1.00</td>
<td>0.1</td>
<td>1706</td>
<td>$10^{87203.0}$</td>
<td>57</td>
<td>7683</td>
<td>$10^{800997}$</td>
<td>99</td>
<td>7971</td>
<td>$10^{7.94e+06}$</td>
<td>113</td>
<td>11264</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1706</td>
<td>$10^{51290.9}$</td>
<td>993</td>
<td>7712</td>
<td>$10^{470401}$</td>
<td>1675</td>
<td>8228</td>
<td>$10^{4.66e+06}$</td>
<td>3392</td>
<td>13312</td>
</tr>
</tbody>
</table>

- exponentially more succinct than representing worlds individually
- $10^{8\cdot10^6}$ worlds need 13 GBs $\approx$ 8 times the size of one world (1.4 GBs)
- case $x = 0$ is the DB generated by the original TPC-H (without uncertainty)
Evaluation of positive relational algebra queries

$Q_1$: possible (select o.orderkey, o.orderdate, o.shippriority from customer c, orders o, lineitem l where c.mktsegment = 'BUILDING' and c.custkey = o.custkey and o.orderkey = l.orderkey and o.orderdate > '1995-03-15' and l.shipdate < '1995-03-17')

- Uncertainty varies from 0.001 to 0.1 → evaluation time up to 6 times slower
- Correlation varies from 0.1 to 0.5 → evaluation time up to 3 times slower
- Scale varies from 0.01 to 1 → evaluation time up to 400 times slower

scale = 1: the answer size ranges from tens of thousands to tens of millions.
Attribute-level vs. tuple-level

SPJ query on six relations represented by equivalent
- attribute-level U-relational databases
- tuple-level U-relational databases
- Trio’s ULDBs (are tuple-level only)
  Skipped the exponential time task of removing erroneous tuples

Experiment only possible for small scenarios:
1% uncertainty, lowest correlation factor 0.1, and scale up to 0.1.
An increase in any of our parameters would create prohibitively large
(exponential in the arity of relations) tuple-level representations.
Papers on MayBMS

Experiments: Confidence computation

Excellent behaviour (within seconds) for
- few variables (100), many ws-descriptors (5K - 50K)
- many variables (100K), few ws-descriptors (01.K - 5K)

Heuristics for variable elimination: good variable choices are extremely valuable even if they require polynomial time

Competitive even when compared with Monte Carlo simulation based on Karp-Luby FPRAS (fully polynomial randomized approx. scheme) for #DNF.

![KL versus INDVE (50 variables, r=2, s=4)](image)

Karp-Luby (KL): with at least 90% probability, the estimated error is within 1%, and 10% resp., from the exact value.
Query evaluation: Example 2

Violated SSN keys: \( \text{possible}(\pi_{r_1}.\text{SSN}((R \ r_1) \bowtie_{r_1}.\text{SSN}=r_2.\text{SSN} \land r_1.\text{N}<>r_2.\text{N} \ (R \ r_2))) \)

Rewritten query on column-store:
\[
S := U_{S[\text{SSN}]} \bowtie_{\psi \land \phi} U_{S[\text{Name}]}
\]
\[
P := \pi_{s_1.\text{SSN}} \text{ as } \text{SSN}((S \ s_1) \bowtie_{s_1.\text{SSN}=s_2.\text{SSN} \land s_1.\text{Name}<>s_2.\text{Name} \ (S \ s_2)))
\]
Uncertainty-aware query language
**Desiderata for a Query Language for Uncertain Data**

- **genericity** — declarative queries, independent from representation details
  - Trio’s TriQL is **not** generic

- **ability to transform data**
  - beyond the filtering of world-sets as in MystiQ

- **ability to introduce additional uncertainty (!!!)**
  - To make it a natural query language for the possible worlds model: compositionality
  - Decision support queries/hypothetical queries
  - Probabilistic databases: extending the hypothesis space to use evidence

- **right degree of expressive power** — not too strong and not too weak

- **efficient query evaluation**
World-set Algebra

- The operations of relational algebra.
  - Evaluated individually, in “parallel” in all possible worlds.

- An operation \( \text{conf}(R) \) for computing tuple confidence values.
  - Computes, for each tuple that occurs in \( R \) in at least one world, the sum of the probabilities of the worlds in which it occurs.

- An operation \( \text{assert}_\phi(R) \) that conditions the database using a constraint \( \phi \).
  - Removes those worlds that violate \( \phi \).

- An operation \( \text{repair-key}_{\vec{A}[@P]}(R) \) for introducing uncertainty.
  - Turns a possible world into the set of worlds consisting of all possible maximal repairs of key \( \vec{A} \) in \( R \).
  - We will also look at a special case of repair-key called choice-of.

- An operation for grouping worlds based on common properties
  - property = answer to a given query
  - (we will not discuss this one here)
Introducing uncertainty using the `choice-of` operation allows to extend the hypothesis space.

\[
\begin{array}{c|ccc}
R^1 & A & B & C \\
\hline
a & 1 & c \\
a & 1 & d \\
b & 3 & e \\
\end{array}
\]

\[
Pr = .5 \quad \ldots \text{ (further worlds)}
\]

\[
S := \text{choice-of}_{A \@ B}(R)
\]

\[
\begin{array}{c|ccc}
S^{1.1} & A & B & C \\
\hline
a & 1 & c \\
a & 1 & d \\
\end{array}
\]

\[
Pr = .5 \times 1/4 = 1/8
\]

\[
\begin{array}{c|ccc}
S^{1.2} & A & B & C \\
\hline
b & 3 & e \\
\end{array}
\]

\[
Pr = .5 \times 3/4 = 3/8
\]

There must be a functional dependency \( R : A \rightarrow B \).

Necessary if we want to introduce evidence.
Operation repair-key

Example: Tossing a biased coin twice.

<table>
<thead>
<tr>
<th></th>
<th>Toss</th>
<th>Face</th>
<th>FProb</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>H</td>
<td>.4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>T</td>
<td>.6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>H</td>
<td>.4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>T</td>
<td>.6</td>
<td></td>
</tr>
</tbody>
</table>

Pr = 1

\[ S := \text{repair-key}_{\text{Toss@FProb}}(R) \] results in four worlds:

\[
\begin{align*}
S^1 & | Toss & Face & FProb \\
1   & H    & .4   \\
2   & H    & .4   \\
S^2 & Toss & Face & FProb \\
1   & H    & .4   \\
2   & T    & .6   \\
S^3 & Toss & Face & FProb \\
1   & T    & .6   \\
2   & H    & .4   \\
S^4 & Toss & Face & FProb \\
1   & T    & .6   \\
2   & T    & .6   \\
\end{align*}
\]

\[
\Pr(S^1) = 1 \cdot \frac{.4}{.4 + .6} \cdot \frac{.4}{.4 + .6} = .16, \quad \Pr(S^2) = \Pr(S^3) = .24, \quad \Pr(S^4) = .36
\]
Operation conf

\[
\begin{array}{|c|c|c|}
\hline
R^A & A & B \\
\hline
 & a & b \\
\hline
 & b & c \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|}
\hline
R^B & A & B \\
\hline
 & a & b \\
\hline
 & c & d \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|}
\hline
R^C & A & B \\
\hline
 & a & c \\
\hline
 & c & d \\
\hline
\end{array}
\]

conf(R) gives the probability of each tuple across all worlds:

<table>
<thead>
<tr>
<th>conf(R)</th>
<th>x</th>
<th>z</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>a b</td>
<td>.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a c</td>
<td>.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b c</td>
<td>.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c d</td>
<td>.7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For a Boolean query \( Q \) and a world-set \( W \), \( conf(Q) \) gives us one number, the probability of the event \( \{ I \in W \mid I \models Q \} \), which is the confidence of tuple \( \langle \rangle \).
Conditioning using assert

Example: enforcing a key constraint on SSN.

<table>
<thead>
<tr>
<th>$U_{R[SSN]}$</th>
<th>V</th>
<th>D</th>
<th>TID</th>
<th>SSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>1</td>
<td>$t_1$</td>
<td>185</td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>2</td>
<td>$t_1$</td>
<td>785</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>1</td>
<td>$t_2$</td>
<td>185</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>2</td>
<td>$t_2$</td>
<td>186</td>
<td></td>
</tr>
</tbody>
</table>

$T := \text{assert}_{fd:SSN \rightarrow TID}(R)$.

We drop the worlds where both tuples $t_1$ and $t_2$ occur with SSN = 185.

<table>
<thead>
<tr>
<th>$U_{T[SSN]}$</th>
<th>$V_1$</th>
<th>$D_1$</th>
<th>$V_2$</th>
<th>$D_2$</th>
<th>TID</th>
<th>SSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>1</td>
<td>y</td>
<td>2</td>
<td></td>
<td>$t_1$</td>
<td>185</td>
</tr>
<tr>
<td>x</td>
<td>1</td>
<td>y</td>
<td>2</td>
<td></td>
<td>$t_2$</td>
<td>186</td>
</tr>
<tr>
<td>x</td>
<td>2</td>
<td>y</td>
<td>1</td>
<td></td>
<td>$t_1$</td>
<td>785</td>
</tr>
<tr>
<td>x</td>
<td>2</td>
<td>y</td>
<td>1</td>
<td></td>
<td>$t_2$</td>
<td>185</td>
</tr>
<tr>
<td>x</td>
<td>2</td>
<td>y</td>
<td>2</td>
<td></td>
<td>$t_1$</td>
<td>785</td>
</tr>
<tr>
<td>x</td>
<td>2</td>
<td>y</td>
<td>2</td>
<td></td>
<td>$t_2$</td>
<td>186</td>
</tr>
</tbody>
</table>