Taylor-Type Techniques for Handling Uncertainty in Expert Systems, with Potential Applications to Geoinformatics

Martine Ceberio, Vladik Kreinovich, Sanjeev Chopra  
NASA Pan-American Center for Earth and Environmental Studies (PACES)  
University of Texas at El Paso  
mceberio@cs.utep.edu, vladik@utep.edu

Bertram Ludaescher  
Department of Computer Science, University of California, Davis
1. Formulation of the Problem

- **Expert knowledge** consists of statements $S_j$: facts and rules.

- **Objective:** given a query $Q$, check whether $Q$ follows from the expert knowledge.

- **Example of a knowledge base:**

  $$S_1 : a \leftarrow b.$$  
  
  $$S_2 : b \leftarrow .$$  
  
  $$S_3 : a \leftarrow c.$$  
  
  $$S_4 : c \leftarrow .$$  

- In this example, $S_1$ and $S_3$ are rules, $S_2$ and $S_4$ are facts.

- **Example of a query $Q$:** $a$?.

- **Answer:** yes, e.g., $Q$ follows from $S_1$ and $S_2$.

- **Tools:** Prolog-type inference engines.
2. Enter Uncertainty

- **Fact:** experts are not 100% confident.

- **How:** the expert’s degree of confidence in each statement $S_j$ can be described as a (subjective) probability $p(S_j)$.

- **Example:** if we are interested in oil, we should look for certain geological structures (confidence 80%).

- **Question:** if a query $Q$ is deducible from facts and rules, what is our confidence $p(Q)$ in $Q$?

- **Example:**
  - to find oil, look for subterranean structures (80%);
  - to find these structures, analyze gravity data (90%);
  - what is our confidence that to find oil, we must look for gravity data?
3. Representation

- **Idea**: we can usually describe $Q$ as a propositional formula $F$ in terms of $S_j$.
- **Example**:
  
  $S_1: a \leftarrow b$.  
  $S_2: b \leftarrow \cdot$.  
  $S_3: a \leftarrow c$.  
  $S_4: c \leftarrow \cdot$.  

  Here, $F = (S_1 & S_2) \lor (S_3 & S_4)$.
- **Resulting problem**:
  
  - we have a propositional combination $F$ of known statements $S_j$;
  - we know the probabilities $p(S_j)$ of different statements;
  - we must determine the probability $p(F)$;
  - to be more precise, we need the interval $p(F)$ of possible values of $p(F)$. 

4. Traditional Approach

- **Fact:** the problem of finding the exact bounds for $p(F)$ is NP-hard.

- **Traditionally:** expert systems use technique similar to straightforward interval computations:
  
  - we parse $F$ and
  
  - replace each computation step with corresponding probability operation.

- **Operations:** if we know the bounds $[a, \bar{a}]$ for $p(A)$ and $[b, \bar{b}]$ for $p(B)$, then:
  
  - $p(A \& B)$ is in the interval
    
    $$[\max(a + b - 1, 0), \min(\bar{a}, \bar{b})];$$

  - $p(A \lor B)$ is in the interval
    
    $$[\max(a, b), \min(\bar{a} + \bar{b}, 1)].$$
5. Traditional Approach: Too Wide

- Example: $F = (A \& B) \lor (A \& \neg B)$, $p(A) = p(B) = 0.6$.

- Parsing:
  - we first find the bounds for $p(\neg B)$,
  - then for $p(A \& B)$ and $p(A \& \neg B)$, and
  - finally, the bounds for $p(F)$.

- Result: $p(\neg B) = 1 - 0.6 = 0.4$;
- $p(A \& B) = [\max(0.6 + 0.6 - 1, 0), \min(0.6, 0.6)] = [0.2, 0.6]$;
- $p(A \& \neg B) = [\max(0.6 + 0.4 - 1, 0), \min(0.6, 0.4)] = [0, 0.4]$;
- $p(F) = [\max(0, 0.2), \min(0.4 + 0.6, 1)] = [0.2, 1.0]$.

- Problem: $F$ is equivalent to $A$, so $p(F) = 0.6$. 
6. Main Idea

- **Similar problem**: excess width in straightforward interval computations.
- **Solution to the similar problem**: Taylor methods narrow down the resulting intervals.
- **Idea behind this solution**: if we use linear Taylor models, then, for each intermediate result $y_j$:
  - we not only keep the interval of its possible values,
  - we also keep the relation between this value and the original inputs –
  - in the form of a linear dependence
    \[
    y_j = a_{0j} + a_{1j} \cdot x_1 + \ldots + a_{nj} \cdot x_n.
    \]
- For quadratic Taylor models, we also keep the relation between $y_j$ and pairs of inputs (as terms $a_{jkl} \cdot x_k \cdot x_l$),
- etc.
7. **Taylor Model-Type Techniques**

- **Main idea:** similarly to Taylor arithmetic, for each intermediate result $F_j$:
  - besides an interval of possible values for $p(F_j)$,
  - we also compute intervals of possible values for pairs $p(F_j \& F_i)$
  - (or even all Boolean functions of pairs);
  - on each step, use all such probabilities to get new estimates.

- **If this is not enough:** we use an analog of $k$-th order Taylor methods – estimate intervals for
  $$p(F_{j_1} \& \ldots \& F_{j_{k+1}}).$$

- The higher the order $k$:
  - the more accurate the results, but
  - the longer the computations.
8. Technical Details

- **Minor problem:** even if we know the probability of triples, then, in general, the problem is NP-hard.

- **Proof:** reduction to satisfiability of 3-CNF formulas.

- **Solution:** when estimating interval for $p(F_i \& \ldots)$, we take into account only $\leq l$ known probabilities.

- **How:**
  - we describe both known and estimated probabilities as sums of probabilities of atomic statements $S_{i_1}^{\varepsilon_1} \& \ldots \& S_{i_m}^{\varepsilon_m}$, where $m \leq k \cdot l$, and
  - use linear programming (LP) to get desired bounds on the unknown probability.

  + When $k \to \infty$ and $l \to \infty$, we get exact results.
  
  - However, computation time grows exponentially with $k$ and $l$. 

9. Example of Using LP

- We know: \( p(A) = a = 0.6 \) and \( p(B) = b = 0.6 \).
- We want to estimate: \( p(A \lor B) \).
- Atomic statements: \( p_{++} = p(A \land B) \), \( p_{+-} = p(A \land \neg B) \), \( p_{-+} = p(\neg A \land B) \), \( p_{--} = p(\neg A \land \neg B) \).
- LP: \( p_{++} + p_{+-} + p_{-+} \rightarrow \min(\max) \) under the conditions:
  \[
  p_{++} + p_{+-} = a; \quad p_{++} + p_{-+} = b; \\
  p_{++} + p_{+-} + p_{-+} + p_{--} = 1; \\
  p_{++} \geq 0; \quad p_{+-} \geq 0; \quad p_{-+} \geq 0; \quad p_{--} \geq 0.
  \]
- General solution: on one of the vertices, i.e., when the largest possible # of inequalities is equalities.
- Specifics: \( p(A \lor B) \) is the smallest when \( p_{-+} = 0 \); \( p(A \lor B) \) is the largest when \( p_{--} = 0 \).
10. Example: Intervals Are Narrower

- **Problem**: estimate \( p(A \lor \neg A) \) for \( p(A) = 0.6 \).

- **Desired answer**: \( p(A \lor \neg A) = 1 \).

- **Parsing**:
  - \( F_1 = A \),
  - \( F_2 = \neg A \),
  - \( F = F_1 \lor F_2 \).

- **Traditional approach**:
  - \( p(F_1) = 0.6 \);
  - \( p(F_2) = 1 - p(F_1) = 1 - 0.6 = 0.4 \);
  - \( p(F_1 \lor F_2) = [\max(0.4, 0.6), \min(0.4 + 0.6, 1)] = [0.4, 1] \).
11. New Approach

• Details:
  
  • $p(F_1) = 0.6$;
  
  • in addition to $p(F_2) = 1 - p(F_1) = 1 - 0.6 = 0.4$, we also use the relation $F_2 = \neg F_1$ to estimate probabilities of other binary combinations:

    \[
    p(F_1 \& F_2) = 0; \quad p(F_1 \& \neg F_2) = 0.6;
    \]
    \[
    p(\neg F_1 \& F_2) = 0.4; \quad p(F_1 \lor F_2) = 1;
    \]
    \[
    p(F_1 \lor \neg F_2) = 0.6; \quad p(\neg F_1 \lor F_2) = 0.4;
    \]
    \[
    p(\neg F_1 \lor \neg F_2) = 1;
    \]

  • based on these estimates, we get $p(F_1 \lor F_2) = 1.0$.

• Result: we get the exact desired probability, with no excess width.
12. Other Examples

- **Example 1:**
  - for \((A \land B) \lor (A \land \neg B)\), the traditional method leads to excess width in comparison with \(A\);
  - if we use triples (analogue of quadratic Taylor approximations), then we can estimate the probability of \((A \land B) \lor (A \land \neg B)\) as \(p(A)\).

- **Example 2:**
  - for \((A \land B) \lor (A \land C)\), the traditional method leads to excess width in comparison with \(A \lor (B \land C)\);
  - if we use higher-order methods, we get the exact interval for
    \[p((A \land B) \lor (A \land C))\]
    i.e., we get **distributivity**.
13. General Comment about Expert Systems and Fuzzy Logic

• A general argument against expert systems and fuzzy logic is that:
  
  • \( p(A \lor \neg A) \) is estimated as \( f(p(A), p(\neg A)) \) – e.g., as \( \max(p(A), p(\neg A)) \), while
  
  • the correct value of \( p(A \lor \neg A) \) is 1.

• Solution:
  
  • in addition to probabilities of individual intermediate statements,
  
  • keep probabilities of pairs, triples, etc.
14. Traditional Trust & Its Limitations

- **Traditional approach**: we either trust an agent or not.

- **Corollary**: if we trust an agent, we allow this agent full access to a particular task.

- **Example**: I trust my bank to handle my account.

- **Sub-agents**: the agent allows trusted sub-agents the same access, etc.

- **Example**: bank outsources money operations to another company.

- **Problem**: trust is not complete: I may have 99.9% trust in bank, bank in contractor, etc.

- **Result**: for long chains, the probability of a security leak increases beyond 0.1%.

- **Problem**: keep track of trust probabilities.
15. **Probabilistic Approach: Main Idea**

- We have a finite set $A$; its elements are called *agents*.
- For some pairs $(a, b)$ of agents, we know the probability $p_0(a, b)$ with which $a$ directly trusts $b$.
- **Objective:** to describe, for given two agents $f$ and $s$, the probability $p_t(f, s)$ with which the agent $s$ trusts the agent $s$.
- **In graph terms:** we have edge $(a, b)$ w/prob. $p_0(a, b)$.
- We must find the probability $p_t(f, s)$ that there is a path from $f$ to $s$.
- **Problem:** we have no information on the dependence between different direct trust links.
16. Possibly Dependent Case: Formulation of the Problem

- *It is usually assumed:* all the trusts are statistically independent.
- *In reality:* trusts may come from the assurances of the same third party.
- *Corollary:* trust may be correlated.
- *Problem:* depending on the degree of correlation, we may get different values of the resulting trust $p_t(f, s)$.
- *In critical systems:* it is reasonable to guarantee the trust only if all possible values of $p_t(f, s)$ are $\geq \tilde{p}$.
- *Equivalent formulation:* the smallest possible value $p_t(f, s)$ of $p_t(f, s)$ exceeds the threshold: $p_t(f, s) \geq \tilde{p}$.
- *Resulting problem:* we must be able to compute this “worst-case” trust probability $p_t(f, s)$. 
17. Precise Formulation of the Problem

- **Given:** graph $(A, E)$.
- **Given:** values $p_0(a, b)$ for all $(a, b) \in E$.
- **We consider:** all possible probability distributions $p(E')$ on the set of all subgraphs $E' \subseteq E$ for which, for every $(a, b) \in E$, we have
  \[
  \sum_{E' \colon (a, b) \in E'} p(E') = p_0(a, b).
  \]
- For every two edges $f$ and $s$,
  \[
  p_t(f, s) \stackrel{\text{def}}{=} \sum_{E' \colon f \rightarrow s} p(E').
  \]
- We define $p_t(f, s)$ as the exact lower bound of all such values $p_t(f, s)$:
  \[
  p_t(f, s) \stackrel{\text{def}}{=} \inf\{p_t(f, s) \mid p \text{ is consistent with the given information}\}.
  \]
- **Objective:** compute $p_t(f, s)$. 
18. This Problem Is Difficult to Solve

- *In the independent case:* we knew the exact distribution $p(E')$.

- *Corollary:* we could use the Monte-Carlo simulation techniques and estimate the $p_t(f, s)$.

- *In the possibly dependent case:* there several different probability distributions $p(E')$ consistent with the given information.

- *Seemingly reasonable idea:*
  - use the Monte-Carlo simulation for each of these distributions, and
  - find the smallest of the resulting values $p_f(f, s)$.

- *Problem:* there are infinitely many such distributions.

- *Result:* we cannot find the smallest possible value $p_t(f, s)$ by simply simulating all such distributions.
19. Possibly Dependent Case: Algorithm

- **Auxiliary definitions:**
  - The length (distrust) of an edge is defined as
    \[ d_0(a, b) \overset{\text{def}}{=} 1 - p_0(a, b). \]
  - The length \( \ell(\gamma) \) of a path \( \gamma = (a_0, \ldots, a_n) \) is defined as usual:
    \[ \ell(\gamma) \overset{\text{def}}{=} \sum_{i=0}^{n-1} d_0(a_i, a_{i+1}). \]
  - The length of the shortest path from \( f \) to \( s \) is defined as:
    \[ d_t(f, s) \overset{\text{def}}{=} \min\{\ell(\gamma) \mid \text{\( \gamma \) is a path from} \ f \text{\ to} \ s\}. \]

- **Formula:** the desired value \( p_t(f, s) \) is equal to:
  \[ p_t(f, s) = \max(1 - d_t(f, s), 0). \]

- **Algorithm:** use Dijkstra’s algorithm to find the shortest path, then compute \( p_t(f, s) \).
20. Acknowledgments

This work was supported in part:

- by NASA under cooperative agreement NCC5-209;
- by NSF grants EAR-0112968, EAR-0225670, and EIA-0321328;
- by Army Research Laboratories grant DATM-05-02-C-0046;
- by NIH grant 3T34GM008048-20S1;
- by Applied Biomathematics.
21. **Algorithm: Justification**

- Let \( p(E') \) be consistent with the given information.
- We want to prove: \( d_t(f, s) \leq \underline{d}_t(f, s) \), where
  \[
  d_t(f, s) \overset{\text{def}}{=} 1 - p_t(f, s).
  \]
- Let \( \gamma_0 = (a_0, a_1, \ldots, a_n) \) be the shortest path from \( a_0 = f \) to \( a_n = s \); then,
  \[
  \underline{d}_t(f, s) = d_0(a_0, a_1) + \ldots + d_0(a_{n-1}, a_n).
  \]
- If there is no path from \( f \) to \( s \) (\( N_t(f, s) \)), then at least one of the connections \((a_i, a_{i+1})\) is not present in \( E' \) (\( N_0(a_i, a_{i+1}) \)):
  \[
  N_t(f, s) \supset (N_0(a_0, a_1) \lor \ldots \lor N_0(a_{n-1}, a_n)).
  \]
- Hence,
  \[
  d_t(f, s) \leq P(N_0(a_0, a_1) \lor \ldots \lor N_0(a_{n-1}, a_n)).
  \]
- So, \( d_t(f, s) \leq d_0(a_0, a_1) + \ldots + d_0(a_{n-1}, a_n) = \underline{d}_t(f, s). \)
22. Proof (cont-d)

- To complete the proof, we produce a distribution $p(E')$ for which
  \[ p_t(f, s) \leq \max(1 - d_t(f, s), 0). \]

- Let $\pi(x) \overset{\text{def}}{=} x - \lfloor x \rfloor$.

- We define $E(\omega)$ for $\omega = U([0, 1])$ as follows:

- For every $(a, b) \in E$, this edge is in $E(\omega)$ iff $\omega \not\in \pi(I(a, b))$, where
  \[ I(a, b) \overset{\text{def}}{=} [d_t(f, a), d_t(f, a) + d_0(a, b)]. \]

- Since $\pi(I(a, b))$ has width $p_0(a, b)$, the distribution $p(E')$ is consistent with $p_0(a, b)$.

- Induction proves that for every path starting at $a_0 = f$, if all its edges $(a_i, a_{i+1}) \in E(\omega)$, then $\omega \geq d_t(a_0, a_n)$.

- Hence, $p_t(f, s) \leq \max(1 - p_t(f, s), 0)$. 