COMPUTATIONAL INFERENCE FOR EVIDENTIAL REASONING IN SUPPORT OF JUDICIAL PROOF

Tod S. Levitt
Information Extraction & Transport, Inc.
Arlington, Virginia
tlevitt@iet.com

Kathryn Blackmond Laskey
George Mason University
Fairfax, Virginia
klaskey@gmu.edu

Abstract

Judicial proof accrues evidence to confirm or deny hypotheses about world events relevant to a legal case. Software applications that seek to support the process of judicial proof must provide the user with sophisticated capabilities to represent and manipulate evidential reasoning for legal cases. This requires computational techniques to represent the actors, entities, events and context of world situations, to structure alternative hypotheses interpreting evidence, and to execute processes that draw inferences about the truth of hypotheses by assessing the relevance and weight of evidence to confirm or deny the hypotheses. Bayesian inference networks are combined with knowledge representations from artificial intelligence to structure and analyze evidential argumentation. The infamous 1994 Raddad murder trial in Nice, France provides a backdrop against which we illustrate the application of these techniques to evidential reasoning in support of judicial proof.

1.0 Overview

Judicial proof involves evidential accrual to confirm or deny hypotheses about current or past states of the world. Examples of legal hypotheses include “John Doe committed the robbery,” and “The contract was breached because the shipment was late.” In order to manage information in a way that is useful to legal practitioners, software applications must employ sophisticated techniques to represent, explore, infer and analyze alternative interpretations of the hypotheses and evidence that arise during prosecution of a case.

Even for simple cases, to explicitly enumerate all potentially relevant states of the world would require resources far exceeding current and foreseeable computing capabilities. Experts are able to identify and restrict attention to key reasoning chains that comprise a tiny fraction of the search space of possible world states. A posteriori analysis by experts reveals contexts, certainties, anomalies and foci that served to organize and guide their inferential reasoning process. We argue that computer support systems must be designed to exploit such attention focusing organization schemes.

In this paper, we present computational techniques for representing and performing inductive evidential reasoning for judicial proof. Figure 1 summarizes characteristics our evidential reasoning framework is designed to capture. We use ***Bayesian networks*** (***BNs***) to capture the structure of arguments and to provide a numerical representation of their strength[1].

1 Charniak, E. (1991) “Bayesian Networks Without Tears”, AI Magazine 12(4), 50-63.

Repeating, parameterized argument structures can be represented as ***BN Fragment (BNFrag)*** objects. A BNFrag is a (fully or partially specified) BN that represents a fairly small, separable, and conceptually meaningful piece of the total argument structure supporting or denying a given hypothesis. BNFrags can be used to represent alternative hypothetical world states, evidence that bears upon which hypotheses are true, and chains of argument relating evidence to hypotheses. BNFrags are components that can be combined with other BNFrags to build complex argument structures, and can be re-used in multiple different case analyses.

Represent hypotheses and supporting or denying evidence.

Compare beliefs between alternative hypotheses.

Update belief based on incrementally accrued evidence.

Examine variations of the same hypothetical-evidential scenario.

Figure 1 Desiderata for Computational Evidential Reasoning for Judicial Proof

Our approach is to dynamically generate and merge BNFrags into larger BNs as evidence accrues relating to a given hypothesis. This approach enables efficient representation and analysis of many complex and inter-related hypotheses and evidential events. Explicit enumeration of the entire space of hypotheses represented by BN models would entail prohibitive computation costs. Dynamic construction of BN models promises to be tractable for problems of realistic complexity.

We illustrate the approach by applying it to elements of the case of Omar Raddad, accused and convicted of murdering and robbing his employer. We indicate approaches to computational representation of world state and evidential reasoning in the context of exploring hypotheses and supporting arguments for the Raddad murder trial. The next section presents a case summary from the viewpoint of computational representation of the actors and events of the case.

Sections 3 and 4 below deal with computational representation and inference respectively. We conclude with a discussion assessing our progress toward computational realization of the desiderata of Figure 1 in support of judicial proof.

2.0 Raddad Case Presentation

We consider the legal case of Omar Raddad, a native Moroccan who was convicted in 1994 of the murder and robbery of his employer, Mrs. Ghislaine Marchal (Mrs. M), at her estate in southern France[2]. We examine elements of this case in order to provide a concrete basis from which to assess the requirements for computational representation and reasoning.

D'Ambrosio, B. (1999) "Inference in Bayesian Networks", AI Magazine, 20:2, pp. 21-36.

Jensen, F. (1996) Introduction to Bayesian Networks, Springer-Verlag, New York.

Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Mateo, CA., p. 116-131.

[2] The use of this legal case to illustrate our computational framework was inspired by discussions with P. Snow and M. Belis, preparatory to the Symposium at the Cardozo Law School organized by Professor Peter Tillers.

Omar Raddad was employed as a gardener on the Marchal estate. On Sunday he worked at a neighboring estate, but he took a long lunch break. On Monday morning Raddad traveled away to visit relatives. On either Sunday or Monday morning, Mrs. M. was killed, presumably murdered with a knife-like object, in the basement of her house on the estate. The murder weapon was not found. There was no sign of forced entry. An open, empty purse was found near the body. Also near the body was the sentence "Omar m'a tuer" ("Omar murdered me") scrawled in Mrs. M.'s blood. It is interesting, and as Mrs. M. was highly educated, perhaps evidential, that the grammar is incorrect. Omar was perennially in debt, indicating a possible motive for robbery. He had a key to the house. Omar was arrested and subsequently convicted of Mrs. M's murder. The case achieved some international notoriety because of the racial element due to Omar Raddad's Moroccan nationality.

The examiner recorded the time of death as an interval on late Monday morning. Subsequently, the examiner's office said that a typographical error had occurred, and that the murder was actually committed during the same interval of the day, but on Sunday. Other relevant evidence includes the fact that the maid, Lilianne, had access to the house, but claimed that she had not been at the estate on Sunday. However, a visitor who rang the doorbell on Sunday morning said that a woman who was not Mrs. M. spoke from inside and told her that "Madame was not at home." In addition, Lilianne had a boyfriend, Pierre, who was an ex-convict with a history of engaging in violent robbery.

Figure 2, Figure 4, and Figure 5 are contiguously numbered sections of a combined table of contextual, evidential and hypothetical prose statements about the murder case. The letters "H" for "hypothesis", "E" for "evidence" and "C" for "context" are listed under the number to indicate the evidential reasoning function of the numbered concept. Terms necessary to describe the case are called out in the third column. These terms require computational representation.

The key representations are a few of the numerous choices for language and labels that must be constructed in order to build up a sufficiently expressive, but still unambiguously interpretable, vocabulary that can express hypotheses and evidence about legal reasoning tasks in a form amenable to computation. Figure 2, Figure 4 and Figure 5 include only a few of the representations needed for a complete evidential reasoning software environment. As we go down the list, we do not usually re-state the representations that have already been assigned, although a few are repeated for clarity. From first principles, any software application or environment that claims to support inductive legal reasoning must be able to represent the concepts of hypotheses, evidence and context. In a criminal investigation, for example, the crime that occurred is part of the context of the investigation. Alternative possible scenarios and perpetrators constitute hypotheses to be considered by investigators; and relevant fact, testimony, and other forms of information constitute evidence regarding the truth of the different hypotheses.

We define a ***hypothesis-alternative*** to be a statement about selected aspects of the state of the world expressed with sufficient detail and clarity that the statement must be either a true or false description of those aspects of the world at any point in time. A ***hypothesis*** is a set of hypothesis alternatives that together have the property that exactly one, and only one, of them is true at any point in time.

#	PROSE STATEMENT	KEY REPRESENTATIONS
1 C	Mrs. M was murdered in the basement of her home in the late morning of either Sunday or Monday, by a single individual who had access to the house and who used a sharp metal object to kill her.	Actor: Mrs. M Role: Perpetrator Activity: dying, murder Space-Location: estate basement Tools: sharp metal object Resources: estate access
2 H	Omar committed the murder on Sunday morning.	Actor: Omar Activity: Murder Space-Time Location: estate, Sunday AM
3 E	Omar had access to the murder location.	Activity: unforced entry Space-Location: estate Tools: key
4 H	Omar had a motive.	Resources: negative funds
5 H	Omar was at the murder location at the time of the murder.	Space-Time Location: estate basement, Sunday AM
6 E	Omar was in debt at the time of the murder.	Resources: negative funds
7 E	Omar was not at the murder location on late Monday morning.	Activity: travel Space-Time Location: relatives house, Monday AM Resources: transport mechanisms
8 H	Omar stole money from Mrs. M's purse on Sunday morning.	Activity: robbery Resources: purse with money Space-Time Location: estate, Sunday AM
9 E	Mrs. M's purse was open and empty subsequent to the murder.	Space-Time Location: hours post-murder in estate basement Resources: empty, open purse
10 H	Mrs. M's purse had contained money prior to the murder.	Space-Time Location: hours pre-murder in estate basement Resources: closed purse with money
11 H	The same person who took the money from Mrs. M's purse after the murder, was the murderer.	Roles: murderer, robber Relation: same actor

Figure 2 Murder Case Statements and Key Abstract Representations

When the hypothesis is binary, that is, consists of a single statement, S, and the alternative is "NOT S", then we follow the common practice of abusing notation to speak of the hypothesis-alternative, S, as synonymous with the hypothesis "(S, NOT S)". That is, we follow convention in referring to a binary hypothesis-alternative, such as, "Omar committed the murder", as the "hypothesis" under consideration. In the Raddad case, suspects include Omar, the maid's boyfriend, and the maid. This leads to consideration of four exclusive and exhaustive single-perpetrator alternatives listed in Figure 3.

1. **Omar killed Mrs. M.**
2. **The maid's boyfriend, Pierre, killed Mrs. M.**
3. **The maid, Lilianne, killed Mrs. M.**
4. **Someone other than Omar, Pierre or Lilianne killed Mrs. M.**

Figure 3 A Hypothesis Consisting of Four Hypothesis-Alternatives

Each of these is a hypothesis-alternative by our definition. Note that we could form other hypothesis-alternatives as exclusive combinations of the alternatives in Figure 3, such as the hypothesis-alternative that "Pierre ***and*** Lilianne killed Mrs. M". The fourth alternative is logically equivalent to the statement: "Any person who is not one of the others explicitly mentioned", and is usually abbreviated as the generic hypothesis-alternative, "Other". Appending "Other" to any list of exclusive hypothesis-alternatives necessarily creates an exclusive and exhaustive list of hypothesis-alternatives, forming a hypothesis.

We have stated that hypotheses must be sufficiently clearly defined that, while their truth-value may be unknown, they represent unambiguously true or false statements about the world. Thus, "Omar killed Mrs. M," is a hypothesis-alternative; "Omar is an honest, upstanding individual," is not, unless there are clearly defined criteria by which a person may be regarded to be honest and upstanding. The latter statement is relevant to the question of whether Omar committed the murder. However, it is not a statement to which a definite truth-value may be ascertained, even if all the facts of the case were entirely known. By this definition, the statement, "Omar is considered by the majority of individuals who know him to be an honest, upstanding individual," is a hypothesis-alternative.

#	PROSE STATEMENT	KEY REPRESENTATIONS
12 E	There was a sentence written in Mrs. M's blood at the time of the murder.	Role: writer Activity: writing sentence in blood Tools: writing instrument Resources: blood Product: sentence in blood
13 H	The sentence accuses Omar of committing the murder.	Role: interpreter Activity: interpreting sentence meaning Resources: multi-knowledge expertise
14 E	The sentence was written with poor grammar.	Actor: interpreter Activity: generating interpretation Product: interpretation
15 H	Mrs. M wrote the sentence as she was dying.	Actor: Mrs. M Activity: writing Tools: finger Resources: blood Product: sentence in blood
16 H	Someone else wrote the sentence.	Role: blood-writer Activity: writing Resources: blood Product: sentence in blood
17 H	The murderer wrote the sentence.	Roles: perpetrator, blood-writer Relation: same actor

Figure 4 Computational Representations for Uncertain Evidence

We define ***evidence*** to be a statement about selected aspects of the state of the world for which the truth-value of the statement is related to the truth-value of one or more hypotheses. Evidence is intended to help confirm or deny the truth of the hypotheses for which it is relevant. Usually, but not always, the truth-value of evidential statements can be assessed via direct observation. In such cases the statement is often known (or assumed) to be unambiguously true or false, but this is not an inherent property of evidence. For example, "time of death" evidence is usually best represented as a probability distribution over a time interval. Especially when evidence is uncertain, evidential statements can usefully be viewed as hierarchical hypotheses.

Context is a set of (possibly uncertain) statements assumed to apply throughout reasoning and analysis of a set of hypotheses. Usually context is a consequence or set of preconditions of the task that motivates the generation and resolution of the truth of hypotheses. For example, item #1 in Figure 2 contains the assumption that Mrs. M was murdered. This is made for the sake of brevity in the example modeling. That statement could be treated as a hypothesis, and ought to be represented at another level of reasoning about the case. This assumption is open to question, and presumably a suicide would be considered as an explicit hypothesis-alternative by an investigative team.

#	PROSE STATEMENT	KEY REPRESENTATIONS
18 C	There are other suspects besides Omar	Role: perpetrator Activity: murder
19 E	The maid's boyfriend, Pierre, is an ex-con with house access via the maid.	Actor: Pierre Role: maid's boyfriend Event: previously convicted of robbery Tools: maid's house key
20 H	The maid's boyfriend, Pierre, is the perpetrator.	Actor: Pierre Roles: maid's boyfriend, perpetrator Activity: murder & robbery
21 E	A witness heard a female's voice in the house on Monday AM.	Role: witness Activity: visiting at estate Event: voice heard saying sentence Interpretation: female voice not Mrs. M.
22 H	The maid was the person the witness heard speak.	Actor: Lilianne Roles: maid Activity: answering the doorbell Space-Time Location: estate front door on late morning Monday
23 H	The maid is the perpetrator.	Actor: Lilianne Roles: maid, perpetrator Activity: murder
24 H	Someone else besides Omar, Pierre or Lilianne is the perpetrator.	Actor: NOT {Omar OR Pierre OR Lilianne} Role: perpetrator Activity: murder
25 H	Mrs. M. committed suicide and also attempted to frame Omar for her apparent murder by using Lilianne to plant evidence.	Actors: Mrs. M., Lilianne Roles: Mrs M: victim, perpetrator, Lilianne: collaborator Activity: Mrs. M: suicide, Lilianne: cover-up and framing
26 E	Handwriting analysis indicates that Mrs. M. was not the blood writer	Actors: Handwriting Analyst Interpretation: "Omar m'a tuer" not written by Mrs. M.

Figure 5 Multi-Suspect Case Statements and Computational Representations

3.0 Computational Representation

The challenge of constructing effective computational representations is a central issue in developing software to support inductive reasoning relevant to judicial proof. The representation of hypotheses, evidence and context requires a vocabulary of terms from which hypotheses can be composed. At a minimum we need to be able to computationally express and manipulate the knowledge listed in Figure 2, Figure 4 and Figure 5.

When organized by knowledge categories and associated relationships, such a single domain-of-interest vocabulary is called an "ontology".[3] Ontologies are organized into hierarchies of categories. For example, the top-level category of "legal cases" includes the category of "criminal cases", which in turn includes the category of "murder cases" which in turn includes the category of "murder cases motivated by robbery".

Figure 2 demonstrates the need to represent categories of entities sharing a common set of attributes. For example, representing the category of ***murder-hypothesis-alternative*** requires representing the attributes of perpetrator, victim, location, weapon and motive. If we unambiguously specify a set of attributes and possible values for the attributes, in software interpretable form, and label them with a name, that name is called a ***software type***. Thus, we can define a ***murder-hypothesis-alternative*** software type by unambiguously specifying the set of attributes we need to represent in order to reason about murder hypothesis-alternatives, together with a computer-interpretable form for the possible values for the attributes. Software types can be hierarchical. The murder-hypothesis-alternative type could be represented as a sub-type of the hypothesis-alternative type. This is useful for defining new types, because sub-types can inherit attributes from their parent types, thus facilitating reuse of knowledge.

In Figure 4 we see first-principle needs for representation of logic-based combinations of more primitive representations in order to express combinations of events and to structure complex alternatives from simpler ones. The contexts, evidence and hypotheses in Figure 5 extend the representational issues to include multiple murder suspects.

3.1 Types, Variables, Predicates and Boolean Combinations

Software languages categorize everything by types because that is the theoretical and practical basis for automated interpretation and manipulation of a statement written in the language. All software languages provide the facility to declare the type of a representation. Abstract types can be defined independent of a particular software implementation.

---

[3] Guarino, N. (1993) "The Ontological Level", in, R. Casti, B. Smith, G. White [Eds.], *Philosophy of Cognitive Scienc*es, Holder, Pichler & Tempsky; Vienna.

Neches, R. and R. Fikes, T. Finin, T. Gruber, R. Patil, T. Senator, & W. R. Swartout. (1991) "Enabling technology for knowledge sharing". *AI Magazine*, 12(3): pp. 16-36.

Sowa, J. (2000) *Knowledge Representation: Logical, Philosophical and Computational Foundations.* Brooks-Cole Publishers.

A "data abstraction" or "abstract class" is a representation whose type is defined independent of any particular software language.[4] In order to implement data abstractions or abstract classes in an application software program, we must first map the abstract types to actual types in the software language that we use to implement the application.

Software languages have built-in types for representing abstract types such as numbers, text, graphical characters, and other standard types of entity. For sophisticated applications, such as evidential reasoning for judicial proof, many of the relevant types are not available in the implementation language, but must be created within the application program. Because we can write software code to create custom types, we can require advanced functionality for abstract representations and still hope to successfully implement them in languages that only provide much more primitive type representations.

We need to be able to represent situations, events, hypotheses, evidence and context, in terms of attributes of individual entities and relationships between entities.[5] For example, "Omar is perennially debt" describes an attribute, indebtedness, of the person, Omar. "Omar had access to the estate" describes a relationship between the person Omar and the crime scene, Mrs. M's estate. The concept of a "predicate" can be used to capture such statements. For example, we can create a predicate "InDebt(Omar)" that represents the statement "Omar is perennially in debt," in a fashion that can be easily interpreted by a software program.

In order to generalize predicates to cover more than a single case, the language needs to be able to represent predicates in which variables represent classes or sets of possible individuals that can take the place of the variable. InDebt(x) or HadAccess(x,y) are predicates that generalize the statements about Omar, the individual. Variables in predicates may have type restrictions. In the above expressions x represents a person and y represents a place. We could replace x, for example, by any of the suspects. This replacement is called "instantiating" the predicate. The predicate, InDebt(x) could be instantiated by any individual that has the same type as the variable x. For a legal evidential reasoning application we would need to create a type such as "person". Then for the InDebt(x) predicate, x would be declared to be of type "person". The variable x can then be instantiated by any individual person: e.g., InDebt(Omar), InDebt(Pierre) or InDebt(Lilianne).

It is possible to instantiate a predicate with individuals such that the predicate does not represent the true state of the world. For example, it may be the case the Lilianne is not in debt and therefore that InDebt(Lilianne) has truth value false. For this reason, predicates are commonly defined to be of type "Boolean", meaning that a predicate, once instantiated, takes on one of two values: TRUE or FALSE.[6] Predicates now become a method to represent hypothesis-alternatives either as specific conjectures, CommittedMurder(Omar), or as general questions, CommittedMurder(x).

---

[4] Abstract Class is also used in object-oriented programming to mean a class that has no proper instances, only subclasses. For example, "Activity" could be viewed this way. Any instance of an activity occurring is probably better indicated by the actual type of that activity, such as "gardening" or "murdering".

[5] Davis, R., Schrobe, H. and Szlovitzs, P. (1993) What is a Knowledge Representation? *AI Magazine* 14:1, 17-33.

[6] More generally, predicates can be defined to take on any of a finite set of target values, although {TRUE, FALSE} is the most common use. A predicate might have a value set of {ON, OFF}, {PAST, PRESENT, FUTURE} or {WIN, LOSE, DRAW}, for example.

Real-world cases require multiple statements to capture sufficient information about the world state to be useful to support exploration and analysis for judicial proof. We need to be able to build up more complex representations from simpler ones. For example, the state of the world that Omar murdered Mrs. M. entails in addition that Omar be present at the estate at the time of death, and that Omar have access to the house. Building more and more complex predicates, such as "CommittedMurderAtHouseWithAccess(x)", is clearly not a scalable approach to computational representation.

Instead we need a method to compose predicates into more complex expressions. A logical technique developed in the nineteenth century is to compose compound statements using logical connectives including conjunction ("AND"), disjunction ("OR"), and negation ("NOT").[7] In the twentieth century, major advances have been made in representing and computing with logic-based combinations of predicates.[8]

We can apply logical composition to predicate representation to make sophisticated statements in machine interpretable form. For example, we can capture the statement "Whoever murdered Mrs. M. must have been present at the time of the murder and had access to the house." with the following Boolean expression that uses "*x*" as a parameter denoting the unknown perpetrator:

"[AtHouse(*x*) AND HadAccess(*x*) AND MurderedMrsM (*x*)] OR [NOT [MurderedMrsM(*x*)]]".

Any statement, i.e. any logical combination of predicates, can represent a hypothesis (alternative). Evidence is any statement that is claimed to bear upon the truth or falsity of a hypothesis expression. Note that one hypothesis can be evidential, or be of type "evidence", with respect to another hypothesis statement. There is no inherent problem with a statement being both of type "evidence" and of type "hypothesis".

Context refers to a set of statements that necessarily condition all other statements relating to an evidential reasoning task. Mathematically, context statements are treated in the same way as evidence statements. Computationally, however, context statements are usually not represented explicitly, but instead are used to select which parts of the model to represent explicitly and what parameters to use. Note that a reasoning system can move back and forth between treating a

---

[7] Babbage, C. (1823) "On the Theoretical Principles of the Machinery for Calculating Tables", *Edin. Phtl. Jrl.* 8, pp. 122-128.

Babbage, C. (1826) "On a Method of Expressing by Signs the Action of Machinery", *Phil. Trans.* 116, pp. 250-265.

Babbage, C. (1837) "On the Mathematical Powers of the Calculating Engine", Ms. in Museum of History of Science, Oxford; re-published (1973), Brian Randell (Ed), *The Origins of Digital Computers*, pp, 17-52, Springer-Verlag.

[8] Bledsoe, W.W. & D. W. Loveland (1984) *Contemporary Mathematics: Automated Theorem Proving - After 25 Years*, American Mathematical Society, Providence, RI.

Boole, G. (1947) *The Mathematical Analysis of Logic*, Blackwell Press, Oxford.

Church, A. (1941) *The Calculi of Lambda Conversion*. Princeton University Press.

McCarthy, John (1960) "Recursive Functions of Symbolic Expressions and their Computation by Machine, part I", Comm. ACM, Vol. 3, No. 4, pp. 184-195.

McCarthy, John (1963) "A Basis for a Mathematical Theory of Computation", in P. Braffort and D. Hirschberg (Eds.), *Computer Programming and Formal Systems*, pp. 33-70. North-Holland Publishing Company, Amsterdam.

statement (e.g, “Mrs. M. was murdered”) as context or as a hypothesis-alternative about which one is uncertain. Judicious selection of appropriate contexts is key to effective evidential reasoning.

Two common situations in which context is useful are temporal stationarity and asymmetric reasoning. When relevant conditions remain approximately constant, i.e. are temporally stationary, during the application of evidential reasoning to a specific task, then it is computationally efficient to “roll-up” the corresponding parameters and evidential values into the rest of the reasoning model. For example, parameters like “season”, “weather”, “day-night” and “time of day” are often relevant, known and fixed for the length of an application computing session.

Asymmetric reasoning occurs when restricting attention to a given context allows us to ignore distinctions and details that are relevant only in other contexts. Our model is then said to be ***asymmetric*** with respect to the original, fully detailed, underlying model. These asymmetries and the simplifications they engender makes contextual reasoning extremely useful for achieving parsimony and tractability.

A common example is “what if” reasoning, as often occurs in the discovery and hypothesis generation phases of legal investigation. In the case referred to above, we assume as context that Mrs. M was, in fact, murdered. In the most complete legal inquiry, this is a hypothesis that could be investigated for its truth or falsity. However, given the superficially observed evidence, it seems practical to assume that she was in fact murdered, and to proceed with the investigation under that “context”. This allows us to ignore evidential chains of reasoning that would be relevant only under the hypothesis that her death was by suicide or natural causes.

3.2 Abstract Type Representations

The above discussion of the Raddad case demonstrated that a number of different abstract types are needed to computationally represent even the most elementary facts of the case. The abstract types in Figure 2, Figure 4 and Figure 5 are collected and defined in Figure 6. The following selects several of these to illustrate the subtle, but critical, first-principle issues that must be considered in constructing computational representations.

ROLES

Omar is an actor. Omar has a role as a gardener, he has a role as a relative in his family, and he has a role as a suspect in a murder case. The fact that his niece might call him the nickname “Uncle Omie” is relevant when he is engaged in his role as a family member, but is probably not germane in his role as a gardener or as a murder suspect. A generic lesson about building legal-support applications that partially motivates the concept of roles is the necessity of limiting the size of search spaces when building sophisticated reasoning applications. It turns out to be surprisingly easy to slow program execution times unacceptably as one accumulates the bodies of relevant information that surround even the most mundane of human endeavors, and consequently must sift through the entire volume to obtain any particular datum.

Having as context a set of task-relevant roles for each actor allows us to pre-select a very small portion of the knowledge of an individual as relevant to a given instance of judicial proof.

TOOLS

Tools are physical, electronic and/or biological entities that are used by actors to perform activities. By this definition, tools are tied to other representations at the "activity" level. If the activity is "hammering", then a rock can be a tool. The notion of a tool is fundamentally dependent on the activity for which the actor wields the tool. We cannot inherently know that an entity is not a tool, as that type classification depends on how an actor uses the entity, not just on the inherent nature of the entity.

Indirectly, a tool is related to the task for which the activity is performed, but at the task level we do not care, or know, by what means the activity will be implemented. Writing in blood versus ink is an activity-, writing-level distinction; the message-written / task-result are the same.

RESOURCES

Resources are quantities that do not function as physical objects in the usual sense, but that are needed to perform tasks. Evidential reasoning typically requires representing and reasoning about whether actors were in possession of the resources required to perform given activities, whether actors committed crimes in order to obtain desired resources, and other hypotheses involving resources and their use. Resources are most often aggregations, for example, of molecules, as in water or gasoline, or aggregations of units as with currency. In the sense of aggregations of many (smaller) objects, resources are usually physical. In the examples of fuel or currency, resources are consumed as they are used. If this were universally true, it would make a convenient distinction. But "expertise" is not "consumed" as used, nor is "expertise" an object like a tool. The choice to classify expertise as a "resource" contrasts with typing it as an "attribute" of a human actor.

SPACE-TIME LOCATION

Representing and reasoning with space and time is critical in evidential reasoning for legal applications. For example, a violent murder cannot be committed unless the murderer was in the physical proximity of the victim at the time of the murderer. Legal cases frequently involve complex chains of reasoning aimed at establishing when events occurred or where particular actors or objects were at the time the events in question occurred.

How space and time are represented is critical to the ability to do the required kinds of reasoning, and limits how efficiently it can be performed. There are many flavors of **"**space-time location**"** beyond the usual concept of a point on the surface of the earth at a specified instant in time. For example, space can be represented as a region or a volume, and time is often represented as an interval, or an exclusive choice between alternative intervals. Space and time specifications can be probabilistic and they can be symbolic. We can represent an estimate of the current location of an ocean-liner based on last communications and heading as a probability distribution over a region of the ocean. We can represent relative times with predicates like "before", "after" and "between".

To appreciate the practical complexity of making these, apparently straightforward, choices of space and time representations consider the computational requirements to automatically determine if Omar or any other suspect could possibly have been present at the murder location at the uncertain time of death of the victim. We must be able to reason with expressions that constrain an actor's location in space-time, and compare those against two disjoint, exclusive uncertain time intervals when the death may have occurred.

Although the point is suppressed in our examples for brevity, note that forensic times of death are probabilistic, based on empirical probability densities for chemical alterations in the human body after death. The programming requirements mount as we choose between the numerous different symbolic temporal calculi[9], and then layer on the continuous estimation representations, such as probability, belief functions or fuzzy sets that are needed to fully represent the available evidence.[10] In practice, that's exactly the sort of information that we would like to have an automated legal-support application be able to handle in order to obtain the most refined conclusions that the evidence supports.

Because space is three dimensional whereas time is only one-dimensional, spatial reasoning calculi are several orders of magnitude harder to develop and program than temporal ones. In addition, because space and time are inextricably related, we need, in effect to "multiply" the representations together to arrive at the needed computational machinery. Fortunately, spatial reasoning does not play a big role in our example case.

---

[9] Allen, J. (1984) Towards a general theory of action and time, *Intl J. Artificial Intelligence*, 23, 123-154.

[10] Kuipers, B.J. and T.S. Levitt (1989) "Navigation and Mapping in Large Scale Space," *AI Magazine*, July, 1988. Also reprinted in *Advances in Spatial Reasoning*, Vol. 2, S. Chen, Ed. Ablex Publishing.

Abstract Representation	Abstract Type Examples	Definition
Action	Step; Increment; Manipulate; Attack; Ascend	A parameterized behavior executed by an actor for a single interval of time.
Activity	Standing; Moving; Communicating; Planning; Working	Purposeful sets of actions engaged in by Actors.
Actor	Human, System, SoftwareProgram	An instance of an individual
Entity	Actor; PhysicalObject	A representation of a temporally persistent physical world objects that has an Identity. An electronic record or file is an entity.
Event	Meeting; Earthquake; Murder	A set of related instances of (actor, activity) pairs that occur in overlapping space-time intervals.
Explanation	Argument; Presentation	An Interpretation together with a finite, sequentially ordered list of prose statements, each statement associated to a hypothesis or evidence within the Interpretation Inference.
Identity	SocialSecurityNumber; UniqueIdentifier;	A representation and associated process that allows Actors to recognize the same Actor or Entity at different instances and times.
Inference	BayesianNetwork; ExpertSystem KnowledgeBase	A set of hypotheses and a process that updates the truth values of the hypotheses based on the instantiation of new or updated evidence.
Interpretation	ExpertJudgement; InferenceResult	A parameterized, observed, assumed or hypothetical WorldState and an associated Inference that purports to verify that an instance of the WorldState actually occurred.
Predicate	Boolean; Relation; Switch; Bit	A function applied to a parameterized statement that evaluates the statement as true or false at any point in time.
Product	Document; Building; Corpse	An entity that results from an actor performing an activity or task.
Relation	Distance; Angle; Bloodline; Parentage;	A predicate consisting of a set of type, space-time, activity and/or resource constraints between entities or actors.
Resource	Fuel, Currency, Ink, Water, Blood, Food, Expertise	Resources are observable quantities that are used in performing activities. It is a representational choice to include tools as resources or not. Power sources such as gasoline or money are resources.
Role	Employer; Employee; Friend; Perpetrator	A set of relevant-information-to and constraints-on an actor's behavior that are adopted or assumed to be in force when the actor is engaged in the role.
SpaceTime Location	{x,y,z,t}; {region, interval}; {place, week}; {p(space), p(time)}	A time-indexed estimate of a location in a one, two, three or four-dimensional space. Space and time locations may be symbolic or any subset of 3D space or 1D time, and uncertainty representations may layer on top of those.
Task	MaintainGarden; DetermineBestOption; SolveMurderCase	A triple consisting of a prior world state, a posterior world state and a set of constraints relating methods to transform the world from the prior to posterior state.
Tool	Hand-tool, Machine, Equipment, Manipulator	An entity that is used by an actor in performing some activity. (NB: tools can include s program, a robot, a hand or finger.)
WorldState	Today-at-location-with-actors-doing-tasks; Yesterday-actor-murdered	A set of constraints on world entities and situations.

Figure 6 Key Representation Definitions

4.0 Computational Inference

A vocabulary of types, predicates and Boolean logical connectives allow us to express hypotheses, evidence and context about a legal case. The next computational requirement is to be able to reason from information about the (partial) truth of some evidence or hypotheses to the (partial) truth of other hypotheses. This requires rules of inference, capability to computationally assess whether a hypothesis is true or false, and also to assess partial truths, i.e. degrees of belief intermediate between strict truth and falsity.

4.1 Accruing Evidence and Calculating Beliefs

A ***belief calculus*** is a mathematically consistent methodology of assigning beliefs to logical expressions of variables and predicates.[11] Bayesian probability theory[12], fuzzy set theory[13], Dempster-Shafer theory[14] and Spohn's inductive logic[15] are some of the different theories that can be used to construct belief calculi, each of which can be layered over first order logical representations. For reasons discussed in section 5.2 we choose to use Bayesian probability as our belief calculus.

BNs are one knowledge representation for implementing a Bayesian probabilistic belief calculus. A BN represents the joint probability distribution for a set of random (or uncertain) variables (RV). The directed, acyclic graph of a BN consists of nodes and directed arcs. Each node corresponds to an RV. For our purposes, we can identify RVs with hypotheses, or with Boolean expressions over logical predicates. They represent exclusive and exhaustive possibilities for an uncertain state of the world. Formally, an RV is a function that maps sets of possible world events to possible states that the events can adopt. A predicate is a type of RV that usually maps its parameters to one of the states {TRUE, FALSE}. For example, In_Debt(x) maps the person x to the state TRUE if the person is in debt, or to FALSE if the person is not in debt. Expressions that are logical combinations of RVs are also RVs. In this paper we will tend to use the terms RV, logical combination of predicates" and "node" (in a BN) interchangeably, although technically there are some minor distinctions.

---

[11] Heckerman, D.E. (1988) "An Axiomatic Framework for Belief Updates", *Uncertainty in Artificial Intelligence 2*, J. Lemmer & L. Kanal [Eds.] North-Holland.

Horvitz, E.J., Heckerman, D.E. & Langlotz, C.P. (1988). "A framework for comparing alternate formalisms for plausible reasoning", *Uncertainty in Artificial Intelligence 2,* J. Lemmer & L. Kanal [Eds.] North-Holland.

Kyburg, H. (1988) "Knowledge", *Uncertainty in Artificial Intelligence 2,* North-Holland.

Shafer, G.& P. Shenoy (1990) "Probability Propagation", *Annals of Mathematics and Artificial Intelligence*, 2, pp. 327-352.

[12] Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA., p. 116-131.

[13] Zadeh, L. (1975), Fuzzy Logic and Approximate Reasoning, *Synthese* 30, 407-428.

[14] Dempster, A.P. (1968), "A generalization of Bayesian inference", Journal of the Royal Statistical Society, Series B, 30, 205-247.

Shafer, G. (1976) *A Mathematical Theory of Evidence*, Princeton University Press.

[15] Spohn, W. (1990) "A general non-probabilistic theory of inductive reasoning" *Readings in Uncertain Reasoning*, J. Pearl and G. Shafer, [Eds.], Morgan-Kaufmann, San Mateo, CA.

Figure 7 shows a basic argument structure for exploratory analysis of the culpability of actor X, where X is used as a label to represent a non-specific individual hypothesized to have committed the murder. The figure shows two copies of the BNFrag, differing only in the probabilities displayed in the belief bars.

Each of the rectangular boxes in the figure represents a random variable (RV), or set of exclusive and exhaustive hypothesis alternatives. The name in the title bar is the name given to the hypothesis.[16] The hypothesis alternatives are listed below the RV name, along with the beliefs assigned to the hypothesis alternatives. Note that because the hypothesis alternatives for a given hypothesis are exclusive and exhaustive, the corresponding beliefs always sum to 100%. The fragment of Figure 7 represents four lines of argument: If X committed the murder, (1) the victim must have been found murdered; (2) X must have entered the house (either a forced or unforced entry); (3) X should have a motive (money is explicitly considered as a motive and all other motives are collapsed into the single category "other"); and (4) X must have been at the estate at the time of the murder. The left-hand part of the figure represents the BNFrag prior to discovery of the victim's body. At this time, the degree of belief is negligible that X committed the murder, reflecting the knowledge that murders are rare events. When the evidence is entered that the hypothesis Victim_Murdered has value TRUE, the belief propagation algorithm updates beliefs to indicate that someone must have committed the murder, most likely had a motive, must have been at the estate at the time of murder, and must have entered the house. In our model, X is the label given to this someone.

There would be no reason to construct this BNFrag if no body had been discovered. After the body has been discovered, the fragment would be retrieved from the system's knowledge base, with the beliefs on the left representing an interim state prior to entering the evidence that the victim had been murdered. Once the evidence has been declared, beliefs would be as shown in the BNFrag to the right of The next step in the argument construction process would be to consider the discovery of the inscription, which would implicate a specific individual, Omar Raddad. Before doing this, we pause to consider the model of Figure 7 in more detail.

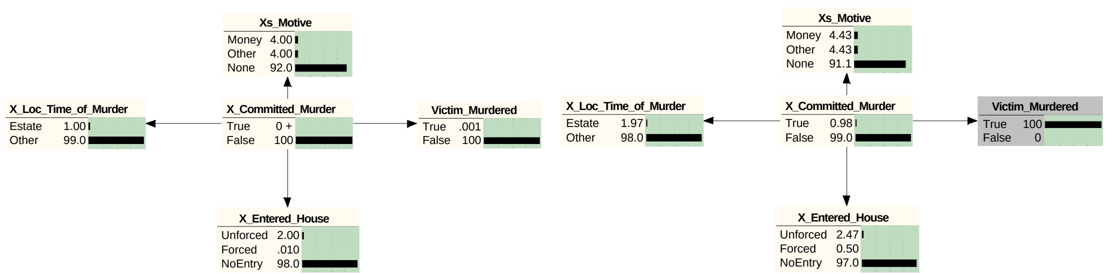


Figure 7 BNFrag Argument Structure for Judicial Proof of Murder

A BN model consists of two parts, structure and probabilities. The structure of a BNFrag model consists of the RVs, or hypotheses, and the links connecting them. In Figure 7 there are five hypotheses, with hypothesis alternatives as shown in the figure. The hypothesis at the tail of an

---

[16] The notation differs from the notation of previous sections because the Netica® Bayesian network package, in which these models were developed, does not allow parentheses in RV names. Thus, for example, the predicate referred to above as CommittedMurder(x) is represented in Netica® as X_Committed_Murder.

arrow is called a parent of the hypothesis at the head of the arrow; conversely, the hypothesis at the head is called the child of the hypothesis at the tail. For example, X_Committed_Murder is a parent of all the other nodes in Figure 7. To specify probability information in a BN, one needs to specify a set of distributions for each hypothesis, one for each possible configuration of hypothesis alternatives for its parent hypotheses.

RV Name	Parent State(s)	Distribution		
X_Committed_Murder	--	**True**	**False**	
		0.00001%	**99.99999%**	
Victim_Murdered		**True**	**False**	
	X_Committed_Murder=True	**100**	**0**	
	X_Committed_Murder=False	**0.001%**	**99.999**	
Xs_Motive		**Money**	**Other**	**None**
	X_Committed_Murder=True	**48%**	**48%**	**4%**
	X_Committed_Murder=False	**4%**	**4%**	**92%**
X_Entered_House		**Unforced**	**Forced**	**NoEntry**
	X_Committed_Murder=True	**50%**	**50%**	**0%**
	X_Committed_Murder=False	**2%**	**0.01%**	**97.99%**
X_Loc_Time_of_Murder		**Estate**	**Other**	
	X_Committed_Murder=True	**100%**	**0%**	
	X_Committed_Murder=False	**1%**	**99%**	

Figure 8 Probabilities for BNFrag Argument Structure

Figure 8 shows the probability distributions for our example. We assume a very small *a priori* probability that the victim would be murdered. The link from X_Committed_Murder to Victim_Murdered indicates logical necessity that the victim was murdered if X murdered her, and a small chance she would be murdered by someone else if not. Note that the model considers any specific individual, absent other evidence, to be 100 times less likely to have murdered the victim than that she would be murdered by *someone*. We assume that murderers most often have a motive for committing the crime. Money is the only motive called out explicitly in this example, but in more complex cases, other motives may be considered. We assume a 100% probability that the culprit was at the estate at the time of the murder,[17] and a small chance that the perpetrator was at the estate at the time of the murder if not committing the murder. As a matter of logical necessity, the purported culprit must have entered the house if he/she committed the murder. We assign equal likelihood to unforced and forced entry. Entry by any means is considered unlikely if the purported culprit did not commit the murder.

[17] The context of our model ignores possibilities such as murder by time delay bomb. One of the benefits of explicit representation of context is that if such a possibility should become relevant, we could revise the context and bring additional possibilities into the model.

The graph of Figure 7 and the set of local probability distributions of Figure 8 together implicitly encode a joint probability distribution over the five random variables shown in the graph. The graph encodes a set of independence constraints that the probability distribution satisfies. For example, the graph says that if we know that individual X committed the murder, then his/her location at the time of the murder is independent of whether or not he/she entered the house. We will say more about these independence constraints later. For now, we note that independence assumptions allow us to structure an argument in terms of modular components that can be treated in isolation from other parts of the argument, and then recombined in a principled way.

The model of Figure 7 and Figure 8 implicitly assigns a probability to each combination of values of all the variables. The total number of probability values implicitly specified by this BNFrag is equal to the product of the number of values of each of the variables, or 2×2×3×3×2 = 48 probabilities.[18] These 48 probabilities are obtained implicitly from the 22 numbers of Figure 8[19]. This may not seem like great savings, but the savings multiply as the model becomes more complex. In models involving hundreds of variables, a BNFrag representation can reduce a task of specifying millions of probabilities to that of specifying a graph structure and a few dozen numbers. The joint probability of any assignment of values to random variables is obtained by multiplying the appropriate factors from Figure 8. For example, the probability that individual X committed the murder for the motive of money, was at the estate at the time of the murder, and entered the house via non-forced entry, is equal to:

**P(X_Committed_Murder=True, Victim_Murdered=True, X_Motive=Money,
X_Entered_House=Unforced, X_Loc_Time_of_Murder=Estate)**

**= P(X_Committed_Murder=True)
× P(Victim_Murdered=True | X_Committed_Murder=True)
× P(X_Motive=Money | X_Committed_Murder=True)
× P(X_Loc_Time_of_Murder=Estate | X_Committed_Murder=True)
× P(X_Entered_House=Unforced | X_Committed_Murder=True)**

= 0.00001 × 1.00 × 0.48 × 0.50 × 1.00 = 0.0000024.

It is rare that we are directly interested in the joint probability of all RVs over which a BN or BNFrag is defined. More likely, we are interested in the probability distribution of a single target or focus RV (or perhaps a small number of RVs) conditional on evidence regarding other RVs. That a BN model implicitly specifies a joint distribution over all variables means that these probabilities of interest can be readily computed. The independence assumptions encoded in a BN can be used to make the inference tractable.

The BNFrag at the left of Figure 7 represents relationships between hypotheses at each node, but does not show any evidence. In a process known as ***entering evidence*****,** we can declare that one of the hypothesis alternatives for one of the nodes is known to be the actual state of the world.[20]

---

[18] Only 47 numbers need to be specified. The 48th value is determined from the other values and the constraint that all the probability values must add up to 1.0.

[19] There are only 13 independent numbers: the other 9 values are determined from the constraint that each of the local probability distributions (each of the rows of Figure 8) must add up to 1.0.

[20] One can also enter *soft evidence*, or evidence regarding the relative likelihoods of different hypothesis alternatives. The examples of this paper deal only with *hard evidence*, or evidence that a particular hypothesis alternative is true.

In our example, we declare that the victim was murdered. As can be seen in the right-hand side of Figure 7 belief in the victim's murder becomes 100% and this node is darkened to indicate its status as an evidence node. Beliefs are automatically updated to incorporate this new knowledge via a process known as ***belief propagation***. The impact of this evidence follows the links in the graph, propagating upward to increase belief in X_Committed_Murder=TRUE, and then downward to the children of X_Committed_Murder. The resulting beliefs are shown in the right-hand side of Figure 7.

This automated solution algorithm for probability propagation and evidential updating in a BN provides the "calculus" part of a belief calculus for automated evidential accrual. A number of published algorithms exist, all of which exploit independence constraints to make inference highly efficient.[21]

4.2 Constructing a Single Suspect Model

The existence of a language for formulating hypotheses and representing evidence, together with a belief calculus for deriving the impact of evidence on beliefs provides the underlying mathematics for a computational system that represents and reasons with evidential arguments. When first invented, these algorithms were initially applied to problems in which a fixed model could be defined and used again and again. The prototypical example is a diagnosis problem in which a list of failure modes, symptoms, and other relevant variables could be identified ahead of time, and encoded into a BN model. Each time the model is applied, a list of observed symptoms is entered as evidence and the BN marginalization produces a probability distribution over the possible faults or diseases or whatever the object of diagnosis might be. Clearly, it is infeasible to construct a priori a single BN model for all possible legal cases, even in a severely restricted domain. Therefore, it is necessary to extend the BN methodology to allow complex models to be automatically constructed from modular components.

Knowledge based model construction (KBMC) systems construct problem-specific models from a probabilistic knowledge base and use the model to answer queries about the domain.[22] A query takes the form of a request for the probability of one or more hypotheses, given a list of evidence

---

[21] Shachter, R. (1986) "Evaluating Influence Diagrams", *Operation Research*, 34, pp. 871-882.

Howard, R. and J. Matheson (1981) Influence Diagrams. In Howard, R. and J. Matheson, editors, *Readings on the Principles and Applications of Decision Analysis*, vol. II, pp. 721-762. Strategic Decisions Group, Menlo Park, CA.

Lauritzen and D.J. Spiegelhalter (1988) Local computation and probabilities on graphical structures and their applications to expert systems. *Journal of Royal Statistical Society B*, 50(2):157-224.

D'Ambrosio, B. (1993) Incremental Probabilistic Inference. In *Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence*, pp. 301-308, July 1993. Morgan Kaufmann, Publishers.

D'Ambrosio, B. (1995) "Local expression languages for probabilistic dependence". *International Journal of Approximate Reasoning* 13:61-81.

[22] Breese, John S. (1987) Knowledge Representation and Inference in Intelligent Decision Systems. Ph.D. dissertation, Department of Engineering-Economic Systems, Stanford University.

Goldman R. P. and J. S. Breese (1992) Integrating Model Construction and Evaluation. In *Uncertainty in Artificial Intelligence 8*, Morgan Kaufmann Publishers, San Mateo, CA. pp. 104-111.

Wellman, M.P., J.S. Breese, and R.P. Goldman (1992) "From knowledge bases to decision models". *The Knowledge Engineering Review*, 7(1):35-53. November.

items and a set of assumed contextual statements. The KBMC system must search and retrieve those BNFrags relevant to the given query and merge them to form a new BNFrag. Then the variables are replaced by specific domain entities, evidence and context variables are declared, and the belief propagation algorithm derives probabilities for the hypotheses of interest.[23]

We have already seen most of the basic elements of our computational framework: BNFrag retrieval triggered by a question posed by an investigator ("Who murdered Mrs. Marchal?"), declaration of evidence, and belief propagation. Two additional elements: specialization of generic BNFrags to specific cases and the combination of BNFrags into a single BN, are illustrated as we consider the evidence of the inscription.

It is with the discovery of the inscription that Omar Raddad becomes singled out for suspicion. Figure 3 shows how the model is extended by incorporating this new evidence and singling Omar out as a suspect. The figure shows two copies of the fragment shown in Figure 1, one specialized to Omar and the other specialized to Other. A third is added to treat the evidence of the inscription.

Depending on the situation, specialization might involve no change, or could require major or minor changes to both the structure and the probabilities. In our case, we leave the entire model intact except for the probability distribution for the RV Other_Committed_Murder. We assign the True state of this RV a prior probability of 0.001%, in contrast to the 0.00001% assigned to X_Committed_Murder=True. The reason for this, as noted above, is that the *a priori* likelihood that any specific individual will commit a murder is greater than the likelihood that some unknown person will commit the murder.[24]

While the BNFrag of Figure 7 is a generic structure that could be applied across a variety of problems, an inscription in the victim's blood accusing someone of the murder is hardly the kind of typical situation for which one would find a ready-made BNFrag in a system's knowledge base. A special purpose BNFrag would have to be constructed to handle this evidence. The fragment has an RV indicating whether or not the victim wrote an inscription implicating Omar (representing the possibility that the inscription could have been forged), and another RV for the evidence that the inscription was found at the murder scene.[25] This fragment would probably be

---

[23] Koller, D. and A. Pfeffer (1997) Object-Oriented Bayesian Networks In Geiger, D. and Shenoy, P. (Eds) *Uncertainty in Artificial Intelligence13*, San Francisco, CA: Morgan Kaufmann.

Laskey, K.B. (1995) Sensitivity Analysis for Probability Assessments in Bayesian Networks, *IEEE Transactions in Systems, Man and Cybernetics 25(6)*, pp. 901-909.

Laskey, K.B. and S. M. Mahoney (1997) Network Fragments: Representing Knowledge for Constructing Probabilistic Models. In Geiger, D. and Shenoy, P. (Eds.) *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference*, San Francisco, CA: Morgan Kaufmann.

Mahoney, S.M. and Laskey, K.B. (1998) Constructing Situation Specific Networks. In Cooper, G. and Moral, S. (Eds) *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference*, San Francisco, CA: Morgan Kaufmann.

[24] To understand why this is so, consider that the proposition "I will draw the Jack of diamonds" is much less likely than the proposition "I will draw a red card," because there are many more ways to draw a red card than to draw a specific card.

[25] Probability distributions for all RVs can be found by downloading the Netica models at http://www.ite.gmu.edu/~klaskey/papers/raddad_model.dne.

constructed as a minor modification of a generic fragment representing evidence incriminating a specific individual.

Note that the RV titled Victim_Murdered is common to both the Omar BNFrag and the Other Suspect BNFrag. This creates a difficulty, because the distribution of this node needs to be defined for all combinations of values of its parent RVs, which means its distribution cannot be defined locally for a single fragment. For this node, the distribution in the combined model is straightforward; it takes value True with probability 100% if any of its parents is true and False with probability 100% if none of its parents is true. We will encounter other cases in which a node has parents in several different fragments. For such situations, an RV object has a method associated with it called ***influence combination***, in which the influences from parents in different fragments are combined to define a distribution for the RV.[26]

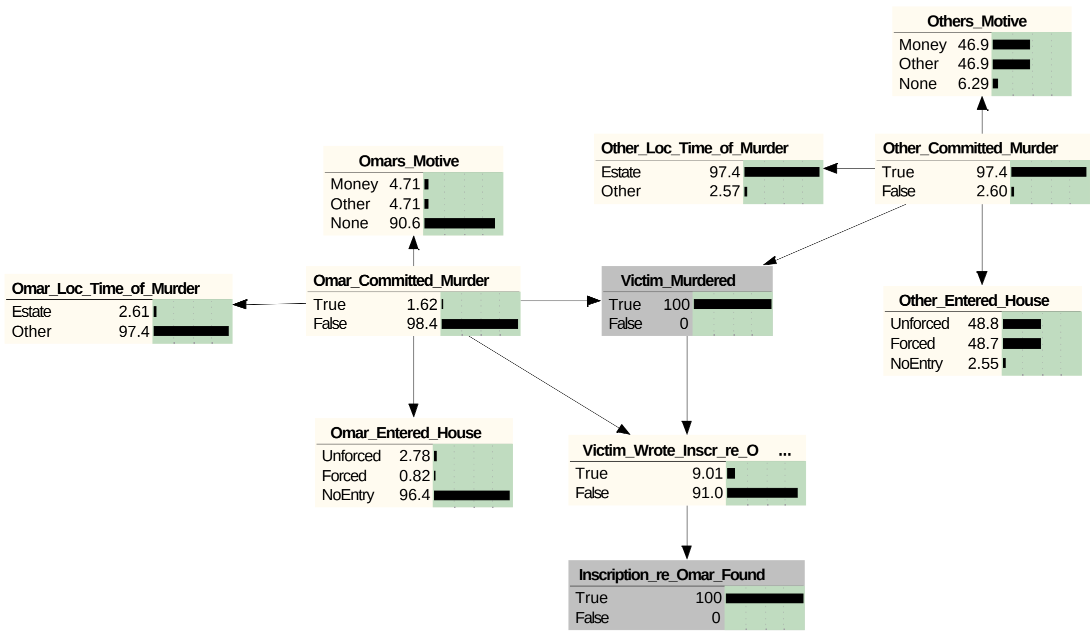


Figure 9 Revised BNFrag for Omar Raddad after Inscription is Found

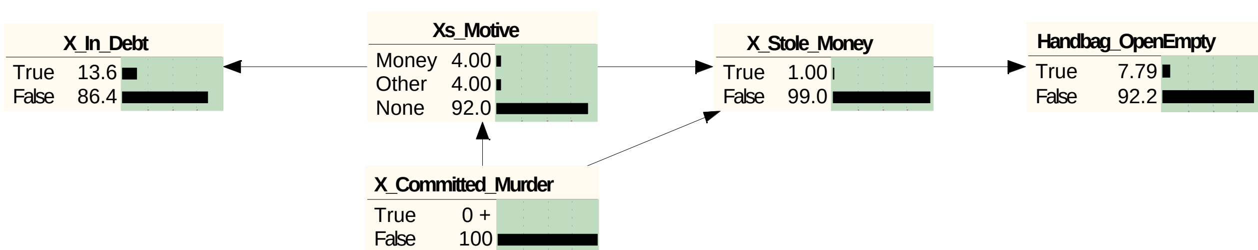
Note that just the evidence of a body being found with an inscription accusing Omar of the murder is sufficient, according to the model of Figure 9, to arouse suspicion. However, the probability the model assigns to his guilt is small – only 1.6%. Further evidence is needed to make a persuasive case that Omar committed the murder. Figure 10 shows two additional generic BNFrags that can be incorporated into the model to process additional items of evidence. Figure 10*a* deals with the motive, and is used to incorporate the evidence that Omar was perennially in debt and that the handbag was found open and empty at the scene of the crime. Figure 10*b* deals with entry to the estate, and is used to incorporate the evidence that Omar had a key and there were no signs of forced entry. The evidence regarding the coroner's time of death

[26] Laskey, K.B. and S. M. Mahoney (1997) Network Fragments: Representing Knowledge for Constructing Probabilistic Models. In Geiger, D. and Shenoy, P. (Eds.) *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference*, San Francisco, CA: Morgan Kaufmann

report and Omar's whereabouts on the two days at issue is represented by another special case BNFrag, shown in Figure 11.

These BNFrags are combined into a single model for reasoning about Omar's guilt, shown in Figure 12. Figure 13 is the corresponding situation specific network, in which RVs irrelevant to the computation have been removed. Constructing and computing only with the situation specific network can be far more efficient than considering the entire BNFrag.[27] There are circumstances, however, in which it is useful to consider the entire BNFrag. For example, although we have no evidence regarding the RV Other_Had_Key, it is useful to know that this variable is relevant, because the investigation can be focused on identifying people who may have had keys to the house. Of course, when such individuals are identified, they will become enumerated explicitly as suspects, as we shall see below in the case of the maid's boyfriend Pierre.

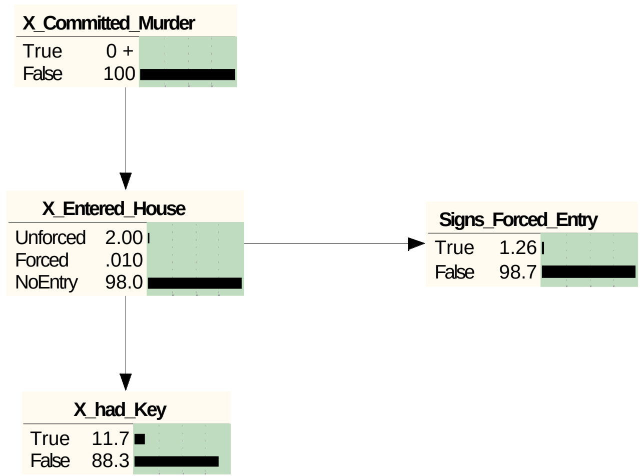
Figure 14 shows how the probability of guilt changes as evidence is entered sequentially into the model. The figure shows three columns, each with a different prior probability of Omar_Commited_Murder=TRUE. These probabilities are 0.00001, 0.0001, and 0.00025, respectively. The prior probability of Other_Committed_Murder remains at 0.001 for all three cases. All other probabilities in the model also remain unchanged. This is a simple example of ***sensitivity analysis***, or examining how sensitive conclusions are to assumptions of the model. In this case, we seek to determine how sensitive conclusions are to the assumptions about the prior probability that Omar is the perpetrator. The key concern is not the absolute prior probability that Omar committed the murder (very small in all three cases), but rather the ratio between this probability and the probability that someone else committed the murder. We consider three different priors, 1:100, 1:10, and 1:4, respectively. It is instructive to note that although the probabilities in all three columns are quite different, the qualitative pattern of how each item of evidence changes beliefs is the same regardless of which prior probability we assume.



a. Motive BNFrag

---

[27] Mahoney, S.M. and Laskey, K.B. (1998) Constructing Situation Specific Networks. In Cooper, G. and Moral, S. (Eds) *Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference*, San Francisco, CA: Morgan Kaufmann.



b. House Entry BNFrag

Figure 10 Additional Generic BNFrags

The model of Figure 12 also demonstrates the dynamic and automatic change of relevance of an item of evidence as other evidence is observed. For example, consider the way belief in Omar's guilt or innocence is affected by the examiner's report. When there is no evidence regarding Omar's whereabouts at the estimated of time Mrs. M's death, the examiner's report on the time of death is not relevant to Omar's guilt. Thus, we see in Figure 14 that belief does not change when this evidence is introduced.

Now consider how the model behaves when we declare, i.e. instantiate the evidence, that Omar has visited his relatives on Monday. If the time of death is on Monday then Omar's visit to his relatives constitutes an alibi. In accordance with the laws of physics, the event that he was at the estate at the time of death given that he was elsewhere at the same time is assigned probability zero. A (generous) 20% probability is assigned to the event that he visits his relatives on any given day. Other than the constraint that Omar cannot be two places at once, this probability is otherwise independent of the time of death and of Omar's location at the victim's time of death.

The probability that Omar was the murderer plummets when evidence of the visit to relatives is introduced, as would be expected when a suspect produces an alibi. But the new report by the examiner, that the original time of death report contained a typographical error and the death actually occurred on Sunday, calls into question the evidential value of Omar's alibi, and dramatically increases the probability that Omar is the perpetrator.

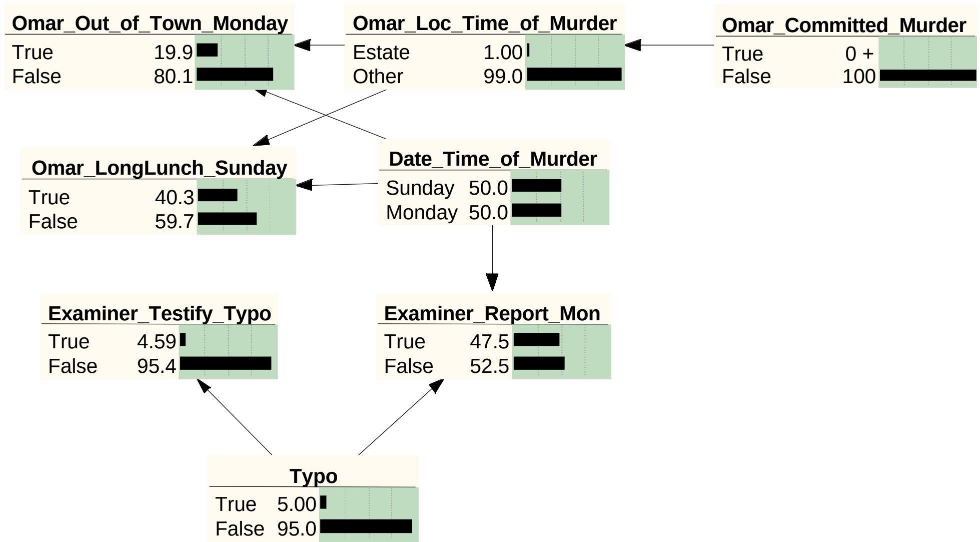


Figure 11 Time of Death BNFrag

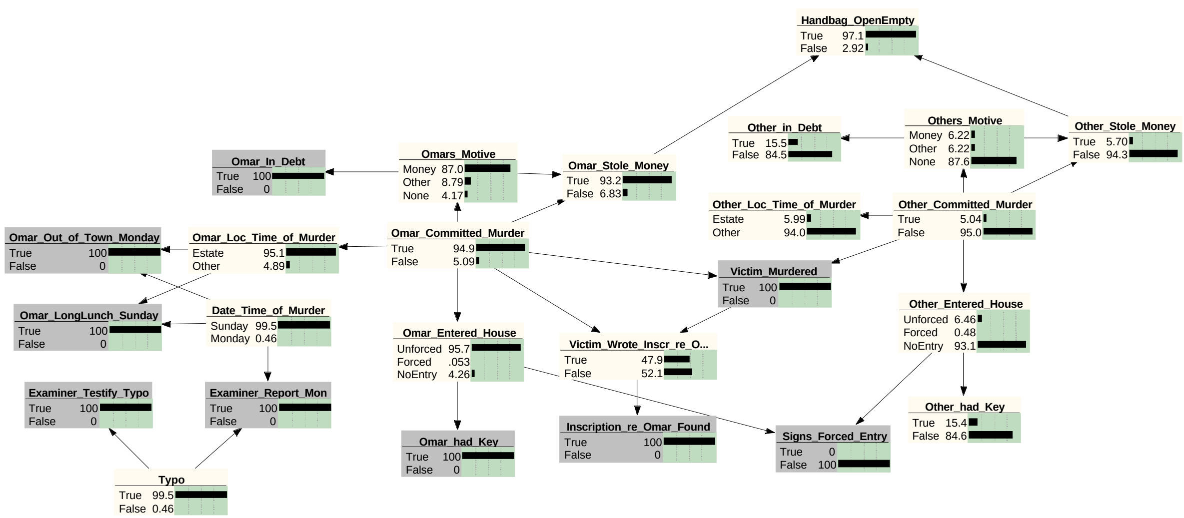


Figure 12 Omar as Suspect BNFrag

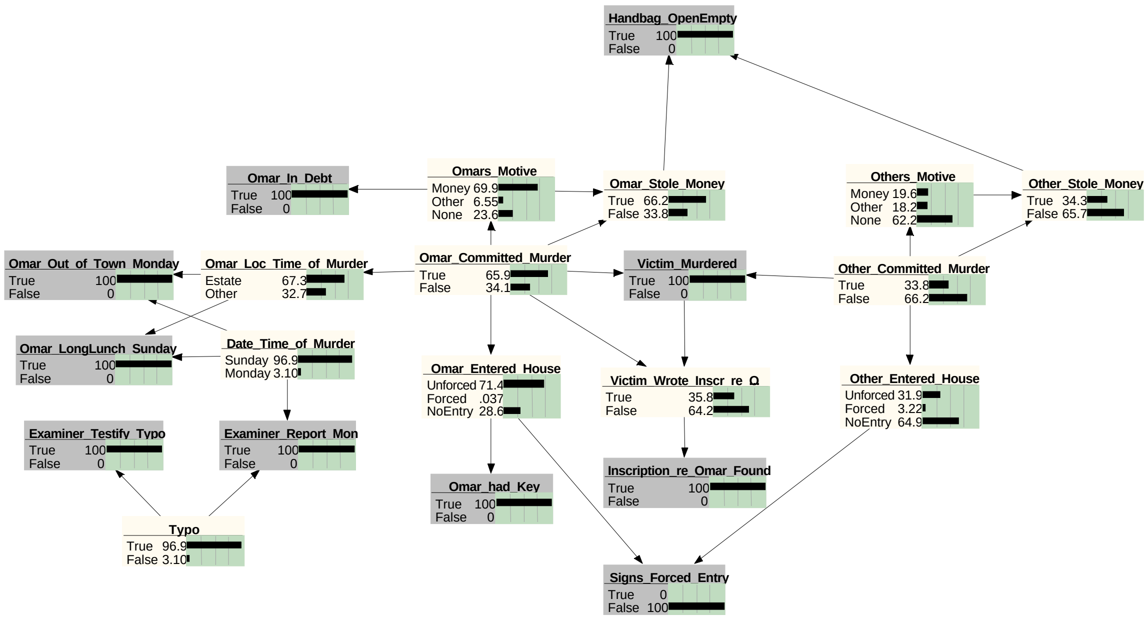


Figure 13 Omar as Suspect Situation Specific Network

Sequentially Observed Evidence Encoded in BNFrags	Cumulative Probability (prior .00001)	Cumulative Probability (prior .0001)	Cumulative Probability (prior .00025)
Victim_Murdered	1.0%	9.0%	19.8%
Inscription_re_Omar BNFrag	1.6%	14.2%	83.3%
Motive BNFrag	6.2%	40.0%	88.8%
House Entry BNFrag	32.8%	92.4%	96.8%
Time Of Death BNFrag			
Examiner_Report_ Monday (evidence)	32.8%	92.4%	96.8%
Omar_Out_Of_Town_Mon (evidence)	0.5%	11.5%	23.0%
Examiner_Testify_Typo (evidence)	30.7%	91.7%	96.2%
Omar_LongLunch _Sunday (evidence)	65.9%	98.0%	99.2%

Figure 14 Evidence Accrual in Omar as Suspect BNFrag

4.3 BNFrags for Incremental Evidence Acquisition & Hypothesis Exploration

To explore the possibility that someone other than Omar committed the crime, we need to expand the set of hypothesized perpetrators. Suppose Pierre is also considered as a suspect. Then a BNFrag for Pierre, analogous to that for Omar in Figure 12, can be constructed.

To build the expanded model shown in Figure 15, we specialize the generic BNFrags of Figure 7 and Figure 10 to Pierre's case, and add additional components specific to Pierre: that he may have had access to the estate through Liliane and that he has a history of criminal activity. Entering this new evidence and propagating beliefs results in a probability of 17.0% that Pierre committed the murder. The probability that Omar committed the murder has decreased from

65.9% to 54.8%. Thus, according to this model, the evidence that someone with a criminal record has a key to the house casts doubt on Omar's guilt. In addition, a potentially important factor has been left entirely out of the model: the possibility that the perpetrator wrote the inscription in an attempt to frame Omar. To extend the model in this way, we would make the nodes Other_Committed_Murder and Pierre_Committted_Murder parents of the node Victim_Wrote_Inscr_re_Omar; break the "False" state into two possibilities in which no inscription was written and in which the perpetrator framed Omar; modify the probability distributions for Inscription_re_Omar_Found accordingly; and incorporate a generic evidence node X_Knew_Omar which would be specialized to Pierre and Other in the revised model. Note that the ungrammatical phrasing of the inscription would become a relevant evidence item, also to be added to the model. An operational evidential reasoning system might include a generic "Suspect was framed" BNFrag that would be modified and tailored for this purpose.

In the interest of brevity, we do not explicitly modify our to incorporate either the hypothesis that the inscription was written by the murder to frame Omar or the hypothesis that Liliane committed the murder. Our purpose in this paper is not to create a definitive model for the Omar Raddad case, but to demonstrate the ability of BNFrags to capture both stereotypic and case-unique patterns of evidential reasoning. We make no claims as to the accuracy of the probabilities generated by our model or the completeness of the model with regard to arguments relevant to the case. A far more detailed and careful analysis, including explicit models for alternative hypotheses such as collusion between Pierre and Liliane, would be required if the purpose were to serve justice in the case of Omar Raddad.

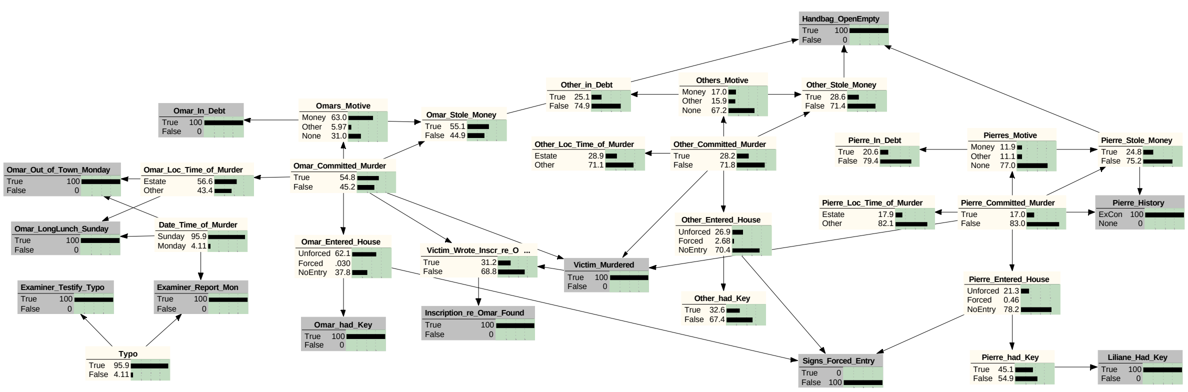


Figure 15 Bayesian Network with Omar and Pierre as Suspects

4.4 Sensitivity Analysis to Assess Evidential Relevance

The BN knowledge representation can capture useful qualitative behavior regarding alternative explanations for the same items of evidence. For example, the relevance of items of evidence regarding Omar depends on their relationship in the evidential argument implied by the BN, and as we have seen, they can change as evidence accrues. In particular, the evidence of Omar's location at the time of the murder is co-dependent with the evidence from the examiner's testimony about the time of death. The relevance of one depends dynamically on the other and they co-vary as evidence is accrued to the global evidential argument about Omar's guilt or innocence that is modeled by the BN of Figure 12. The introduction of the examiner's testimony in Figure 12 does not change the probability of Omar's guilt. That is, the evidence is not relevant to Omar's guilt given the evidence accrued up to that point. The examiner's report becomes relevant when we accrue the evidence that Omar was with his relatives on Monday. In the presence of the examiner's report, the evidence provides an alibi and greatly reduces the probability of guilt. Subsequently, the evidence regarding a possible typographical error of the recording of the day of death changes the relevance of Omar's alibi for his whereabouts on Monday from very strongly relevant to very weakly relevant.

The process of exploring complex models to identify subtleties such as this can be facilitated by computational tools, which are in turn enabled by the sophisticated representational and inferential capabilities of the modular BNFrag architecture described in this paper. For example, sensitivity analysis can be used to examine the impact of changes in modeling assumptions or of the strength of relevance of evidence to hypothesized conclusions.[28] The term sensitivity analysis has multiple, related, but different definitions in the literature on statistics and scientific experimentation. In this section we illustrate the use of a particular sensitivity analysis, sometimes called an "importance measure" specifically to compute a measure of the weight or relevance of evidence items to a BN query.[29] An item of evidence is called "relevant" to a hypothesis if observing the evidence changes the probability that the hypothesis is true. Relevance can be determined purely from the BN graph without specifying the probabilities, but the graph alone cannot provide information about the strength of a relationship that has been determined to exist. This is the role played by the numerical sensitivity analysis method we apply.

[28] Laskey, K.B. (1995) Sensitivity Analysis for Probability Assessments in Bayesian Networks, *IEEE Transactions in Systems, Man and Cybernetics 25(6),* pp. 901-909.

Morgan, M. G, M. Henrion and M. Small (1990) *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis,* Cambridge University Press, New York.

[29] Weiss, Jonathan J. (1995) *Importance Analysis Measure:* IET Technical Report, August 1995.

a. Evidence: Victim Murdered		*b.* Evidence: Victim Murdered, Inscription Found, Examiner Reports TOD Monday	
Importance	**Variable Name**	**Importance**	**Variable Name**
********** 77	Omar_Entered_House	********** 95	Other_Committed_Murder
********* 70	Other_Committed_Murder	********** 95	Omar_Loc_Time_of_Murder
********* 70	Omar_Loc_Time_of_Murder	********** 94	Other_Loc_Time_of_Murder
********* 70	Other_Loc_Time_of_Murder	********** 94	Other_Entered_House
********* 70	Other_Entered_House	********** 92	Omar_Entered_House
********* 67	Omar_Stole_Money	********** 92	Omar_Stole_Money
***** 37	Other_Motive	******** 75	Other_Motive
***** 35	Other_Stole_Money	******** 74	Other_Stole_Money
**** 31	Omar_Motive	******* 69	Omar_Motive
* 11	Omar_in_Debt	*** 30	Omar_in_Debt
* 11	Omar_had_Key	*** 29	Omar_had_Key
** 9*	*Victim_Wrote_Inscr_re_O.mar*	** 22	Other_in_Debt
** 8*	*Omar_LongLunch_Sunday*	** 21	Other_had_Key
* 8	Other_in_Debt	*** 15*	*Omar_Out_Of_Town_Monday*
* 7	Other_had_Key	3	Pierre_Committed_Murder
2	*Omar_Out_Of_Town_Monday*	2	Pierre_Entered_House
2	*Inscription_re_Omar_Found*	2	Pierre_Loc_Time_of_Murder
1	Pierre_Committed_Murder	2	Pierre_Stole_Money
1	Pierre_Entered_House	1	Pierres_Motive
1	Pierre_Loc_Time_of_Murder	1	Pierre_History
1	Pierre_Stole_Money	*1*	*Omar_LongLunch_Sunday*
0+	Pierres_Motive	0+	Pierre_in_Debt
0+	Pierre_History	0+	Pierre_had_Key
0+	Pierre_in_Debt	0+	Lilane_had_Key
0+	Pierre_had_Key		
0+	Lilane_had_Key		

Figure 16 Importance Analysis for Non-Monotonic Evidential Relevance Assessment

We illustrate how to prioritize evidence gathering by examining which unobserved evidence items, if observed, would have the strongest influence on a focus variable. Consider the model of Figure 15 prior to the collection of any evidence except that the victim was murdered. Figure 16*a* shows an analysis of the importance of each evidence item, as measured by the impact of learning its truth-value on the probability distribution for the perpetrator node. According to this model, evidence items relating to estate access of Omar and Other are most important, followed by motive. After these, the next most important variables relate to the inscription and to Omar's whereabouts on Sunday. Figure 16*b* repeats the analysis after the inscription has been found and the examiner has reported that the time of death is Monday. Note first that the variable indicating whether the inscription was written by the victim has become negligible in importance. Note also that the importance of the long lunch on Sunday and the visit to relatives on Monday have reversed. This analysis shows how the impact of evidence can change conditional on what other evidence is present.

BNs are a natural way to capture such subtleties of evidential interactions, and tools such as the sensitivity analysis described here provide support to consumers of models in exploring the implications of the set of assumptions encapsulated in the model. Computational tools both to identify qualitatively which evidence items are relevant to a query and to measure quantitatively the weight of these evidence items if observed, can be of substantial use in constructing evidential arguments and prioritizing investigative resources.

The importance, or weight of relevance of evidence to a hypothesis can be mathematically formalized in the BN concept of "d-separation".[30] In particular, any two nodes connected one to the other via a directed path are relevant to each other (i.e., not d-separated from each other). Generally, the more intervening nodes, the weaker the relevance relationship.[31] As noted above, directed graphs are capable of encoding a type of relevance not represented in undirected graphical models: competing causes of a common effect that become relevant by virtue of evidence that the effect has occurred.[32] If two RVs are "d-separated" (i.e., conditionally independent given currently observed evidence) in the BN encoding the relationship between the variables, then the inferential weight one has on the truth-value of the other can be completely summarized by the truth-values of the d-separating evidence nodes in the BN.[33] In a strong sense, each RV has no direct inferential impact on the other given the evidence nodes that d-separate them. This property can be used to reduce the complexity of inferential argument by summarizing it into the smallest relevant set of RVs and relationships.

The concept of *d*-separation is different from ordinary graph separation, because it takes into account the direction of the arcs in the graph. Because of this, *d*-separation is able to capture important qualitative features of reasoning. To understand *d*-separation, consider the simplest

---

[30] Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA., p. 116-131.

[31] Schum, D. A., 1994, *Evidential Foundations of Probabilistic Reasoning*, Wiley, New York.

Wellman, M. P. (1990) Fundamental Concepts of Qualitative Probabilistic Networks, *Intl J. Artificial Intelligence*, 44, 2587-303.

[32] Pearl, J. (2000) *Causality: Models, Inference and Reasoning.* Cambridge University Press.

[33] Druzdzel, M. J. (1994) "Some Useful Properties of Probabilistic Knowledge Representations From the Point of View of Intelligent Systems*", Proc. Third International Workshop on Intelligent Information Systems (WIS-94),* pp. 278-292.

Lin, Y. and M. J. Druzdzel (1997) Computational Advantages of Relevance Reasoning in Bayesian Belief Networks. In Geiger, D. and Shenoy, P. (eds) *Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference,* San Francisco, CA: Morgan Kaufmann. pp. 342-350.

relevant BN consisting of three nodes in which there is no arc between two of them, but both are connected to the third. The qualitative features of this connection depend on whether the connection is converging (arcs point from both unconnected nodes to the third) or non-converging. In Figure 12 the two nodes Omar_Loc_Time_Of_Death and ToD have a converging connection to Omar_Out_Of_Town_Mon. Examples of the two kinds of non-converging connections are the serial connection from Omar_Motive to Omar_Stole_Money to Handbag_Open_And_Empty, and the diverging connection from Typo to Examiner_Report_Mon and Examiner_Testify_Typo.

Converging connections exhibit a characteristic feature known as “explaining away”, which is useful in evidential reasoning for judicial proof. The discussion of Section 4.2 described an instance of this phenomenon: the examiner’s report of a typographical error explained away the evidence of the Monday visit to relatives, negating the evidential value of the visit for the purpose of ascertaining Omar’s location at the time of death.

Consider the two variables Omar_Loc_Time_Of_Death and ToD. It is straightforward to verify that the only connections between these nodes in Figure 12 are through the converging connections to Omar_Out_Of_Town_Mon and At_Lunch_Sunday. When two nodes are connected to each other only through serial connections to nodes for which there is no direct or indirect evidence, the two nodes are not directly relevant to each other.

The examiner’s report is directly relevant to the time of death, as evidenced by the direct link in Figure 12. However, it is not directly relevant to Omar_Loc_Time_Of_Death in the absence of any evidence regarding either Omar_Out_Of_Town_Mon or At_Lunch_Sunday. It becomes relevant only when the examiner’s report is received regarding the day of death. This changing relevance relationship is a direct consequence of the structure of the model, in particular of the pattern of converging and diverging links, and is not affected by the particular probability values in the prior conditional probability tables.

5.0 Progress Toward Desiderata Supporting Judicial Proof

In each subsection below, we assess our progress contributed by the results in this paper to one of the desiderata identified in Figure 1. We complete our exposition with a summary of accomplishments and their potential impact for legal applications in support of judicial proof.

5.1 Represent hypotheses and supporting or denying evidence.

As illustrated in the example presented in this paper, complex and subtle entities, relationships, situations, events and arguments can be represented for effective computation. Types, predicates, and their logical combinations into Boolean expressions of random variables can be utilized to construct computational representations. Causality, functionality, factual and uncertain information can be captured and manipulated in persistent electronic forms that are intuitive to legal practitioners.

Hypotheses can be represented as exclusive and exhaustive sets of world-states. Methods were described to recognize and combine evidence relevant to the truth or falsity of various hypothesis-alternatives. Context for argumentation about hypotheses can be explicitly included into representations, or held as “notes” for human investigators to take into account as they interactively structure evidential argumentation in support of judicial proof.

Effective use of representational concepts requires commercial software tools that are tailored to provide basic computational components from which specific modular and hierarchical arguments could be developed on a case by case basis. While we made use in our example of automated software for accruing evidence in the BN models we constructed, the construction and combination of BNFrags, and the instantiation of BNFrags with specific individuals, was done manually. Experimental prototype versions of software with limited capability to manipulate and combine BNFrags exist.

5.2 Compare beliefs between alternative hypotheses.

We have demonstrated that practical evidential reasoning in support of judicial proof can be accomplished using BN representations for interacting hypotheses and evidence. The automated solution algorithms for BNs provide a powerful tool to assess and compare quantitative beliefs based on the qualitative world states, hypotheses, and evidence represented by the legal practitioner.

Some form of ranking of hypotheses by degree of belief is required for evidential exploration. In a ranking, hypotheses certain to be true are ranked higher than uncertain hypotheses, which in turn are ranked higher than hypotheses certain to be false. In order to compare beliefs in alternative hypotheses, we use Bayesian probability theory for a belief calculus over our representations. There are multiple reasons for this choice.[34]

---

[34] Bernardo, J.M. & A.F.M. Smith (1994) *Bayesian Theory*, Wiley & Sons, Inc.

Heckerman, D.E. (1988) “An Axiomatic Framework for Belief Updates”, *Uncertainty in Artificial Intelligence 2*, J. Lemmer & L. Kanal [Eds.] North-Holland.

Howson, C. & P. Urbach (1993) *Scientific Reasoning: The Bayesian Approach*, Open Court, Chicago, IL.

Kyburg, H. (1988) “Knowledge”, *Uncertainty in Artificial Intelligence 2*, North-Holland.

1. Bayesian probability theory provides a coherent and consistent calculus with well understood semantics that is compatible with the scientific method from first principles
2. Probability theory supports a total ordering of hierarchical hypotheses, which is necessary for effective automated support for weight, relevance and evidential importance computations.
3. Computational tools for constructing probability models are commercially available.
4. The theory for constructing models from reusable BNFrags extends naturally from the theory underlying probability models and is reasonably well developed, although is not yet available in commercial software applications.

We make no claim that any of our choices of numerical values for probabilities are “correct” or “accurate” probabilities. The assignment of probabilities is an exercise that must be done systematically and with scientific rigor. If BN or similar models were to be admitted in formal arguments at the bar, then the process of vetting both the structures of the model and the probabilities would need to be justified according to reasonable legal standards.

5.3 Update belief based on incrementally accrued evidence.

We have presented the “Omar as suspect” model of Figure 12 as a single static structure and described how the model could be extended to incorporate an additional suspect. We discussed how such a model could be built up beginning with only the first few BNFrags, and then adding BNFrags to incorporate additional evidence items. In our example, all the evidence was available at the outset and incremental model construction was applied mainly as an expository device in explicating the model. If this technology were to be applied to support the construction of legal arguments and the prioritization of evidence gathering, the incremental construction process would reflect the incremental nature of how evidence is accrued in practice. For example, the two nodes Typographic_Error and Examiner_Testifies_Typo were not included in the original hypotheses developed by the Raddad case investigators. These nodes would be added to the existing BNFrag at the time the hypothesis of the typographical error was introduced.

Evidential reasoning in a typical legal case would not involve constructing a single monolithic static model encompassing all conceivable patterns of evidence that could be observed. Rather, we would begin with a focus variable of interest and a few key items of evidence, and add evidential reasoning chains as we encountered new evidence, introduced new hypotheses, or evaluated whether or not to seek a given item of evidence. Thus, we would have not a single model, but a nested sequence of models, where each model in the sequence contained earlier models as sub-structures.

Processing the same query on the succession of models gives a picture of how conclusions change as evidence is accrued. We can examine multiple scenarios of different subsets of the observed evidence, to determine which evidence items have the strongest impact on conclusions. We can examine whether evidence is corroborating, impacting the conclusion in the same direction and strengthen each other; or conflicting, i.e. impacting the conclusion in opposite directions.

As we have already seen with the example of the examiner’s testimony, the relevance and corresponding strength of evidence and whether it corroborates or conflicts with other evidence may change with the presence of additional evidence. Our analysis suggests that this can occur because the various evidence items are related through a common functional structure that exists in the world. The Bayesian belief calculi solve the otherwise quite difficult computational problem of updating these sophisticated structures as new evidence and BNFrags are appended.

5.4 Examine variations of the same hypothetical-evidential scenario.

A unique application of sensitivity analysis has been developed for BNs that provides a quantitatively powerful, scientifically meaningful and qualitatively intuitive measure of the relevance and importance of evidence to the truth or falsity of a target hypothesis or BN query.

Before evidence is observed, we can assess the impact of observing any possible state of an evidence variable, as well as the impact, if relevant, of failing to learn anything about the evidence variable. We can examine the degree to which the probabilities in which we are most interested would be affected by observing the evidence. We can rank the different evidence-gathering strategies by impact on the conclusion, and prioritize evidence gathering accordingly.

When there are monetary or other costs to evidence gathering, these techniques provide tools for balancing the information gain of evidence gathering against the cost. Sheer volume necessitates that we find reasoning and programming methodologies for dynamically constructing subsets of our knowledge base(s) that are relevant to any immediate reasoning task. In this way our total size of knowledge bases can grow without bound while the time to perform any specific search remains approximately constant.

5.5 Summary

Our results demonstrate that complex legal arguments in support of judicial proof can usefully be represented in a computational architecture. Computational representation and inference enables analysis of the impact of different items of evidence on a conclusion of interest with both qualitative and quantitative precision far exceeding that which can reasonably be achieved without software support. We have scoped our case analysis to be simple enough to fit on a page, but have demonstrated how the complexity of a model can increase rapidly as additional hypotheses and evidence are considered. Realistically complex cases can generate models with hundreds of variables.[35] The modularity of the models allows an investigator to focus on meaningful contexts and context-relevant sets of variables, thus enabling a complex argument to be broken into meaningful pieces and then recombined in a principled way.

BN models can be used to explore a large variety of "what if" arguments. We can consider different combinations of evidence in varying contexts and assess their impact on alternative hypotheses. Numerical values for probabilities would have to be carefully justified if such a model were to be used in court. However, using only rough order of magnitude assessments, this model's incremental evidential accrual produced strikingly intuitive qualitative behavior. This suggests that methodologies to standardize acceptable numerical estimates might be forthcoming, should the legal community take on the issue.

Regardless of organized pursuit by the legal community, exploration of the behavior of such a model could prove quite fruitful as an aid to constructing legal arguments even if the numeric results were not considered sufficiently accurate or reliable for application in court. But representational issues must be addressed prior to issues of numerical or symbolic assessments of

---

[35] Kadane, J.B. and Schum, D. (1996) *A Probabilistic Analysis of the Sacco and Vanzetti Evidence*. New York: Wiley.

truth or falsity of hypotheses. Issues of clarity, expressiveness and completeness drive automation of subtle reasoning capabilities to a far greater degree than does the choice of numerical estimation technique. Given the existence of easy to use and robust tools for performing computations on complex probability models, representational issues form the greatest technology barrier to robustly useful legal evidential reasoning applications.

We have addressed a number of fundamental issues that arise in software development in support of judicial proof, including the structuring of arguments in a computational framework, incremental assessment of the weight of evidence to a hypothesis alternative and the relevance of evidence to an argument. We have shown how to integrate and apply computational theories of knowledge representation to provide a capability for constructing legal arguments from modular knowledge fragments that represent commonly occurring and re-usable patterns of argument. We have presented computational techniques that semi-automate support for the construction, exploration, examination and evaluation of the structural aspects of legal arguments, as well as providing intuitive and effective numerical analyses. We conclude that the development of applications providing powerful, semi-automated support for judicial proof is within the reach of existing computational technology.

6.0 Acknowledgements

We gratefully acknowledge the students of George Mason University course INFT 819 and scientists at IET, Inc. for providing useful feedback on the models described in this paper. Special thanks go to Dr. Keung-Chi Ng for performing the sensitivity analyses reported in the paper.