Knowledge Based Crime Scenario Modelling

Jeroen Keppens
Department of Computer Science
University of Wales,
Aberystwyth SY23 3DB, UK

Burkhard Schafer
School of Law
The University of Edinburgh
Old College, South Bridge
Edinburgh, EH8 9YL, UK

Abstract

A crucial concern in the evaluation of evidence related to a major crime is the formulation of sufficient alternative plausible scenarios that can explain the available evidence. However, software aimed at assisting human crime investigators by automatically constructing crime scenarios from evidence is difficult to develop because of the almost infinite variation of plausible crime scenarios. This paper introduces a novel knowledge driven methodology for crime scenario construction and it presents a decision support system based on it. The approach works by storing the component events of the scenarios instead of entire scenarios and by providing an algorithm that can instantiate and compose these component events into useful scenarios. The scenario composition approach is highly adaptable to unanticipated cases because it allows component events to match the case under investigation in many different ways. Given a description of the available evidence, it generates a network of plausible scenarios that can then be analysed to devise effective evidence collection strategies. The applicability of the ideas presented here are demonstrated by means of a realistic example and prototype decision support software.

1 Introduction

Methodologies for evaluating physical evidence in a major crime investigation and for determining effective strategies to proceed with the investigation rely on the formulation of hypothetical crime scenarios that can explain the available evidence. Ultimately, crime investigators and forensic scientists aim to discover what scenario has actually taken place. Therefore, their effectiveness is crucially dependent upon the investigator's ability to hypothesise plausible scenarios and to undertake investigative actions that are suitable for differentiating between them.

As humans are relatively poor at hypothetical reasoning, decision support systems (DSSs) may provide a useful means of assisting human investigators in constructing plausible scenarios and analysing them objectively. While existing work has tackled many of the issues involved in evidence collection and interpretation, research into generating the plausible underlying scenarios has focussed on appropriate argument structures [36], recalling similar instances of volume crime [45] and methodologies for human scenario generation [23]. At present, no knowledge driven approaches for automatically generating crime scenarios from evidence have been devised yet.

This paper addresses this lack by presenting a novel compositional modelling method and shows how it can aid human crime investigators. Compositional modellers [17, 26] aim at capturing a domain's first principles, i.e. fundamental theories describing the behaviours and mechanisms that occur in the domain of interest, by means of small, generic and reusable rules, called model fragments. The compositional modelling paradigm is adapted to the crime investigation domain by employing causal rules describing how combinations of assumed states and events lead to new states and events in plausible crime scenarios. A novel model composition algorithm that can abductively construct a space of plausible scenarios by means of such first principles is conceived. To demonstrate its usefulness, the paper also introduces some methods for analysing the resulting scenario space.

The remainder of this paper is organised as follows. First, Section 2 elaborates on the motivations underlying this work. Section 3 then presents an overview of the software system proposed. Sections 4 and 5 describe the knowledge presentation and inference mechanisms of the DSS respectively. The ideas expressed in the course of these theoretical discussions are illustrated by applying them to a small yet realistic example. The feasibility of the approach is shown by describing in Section 6 a prototype DSS available for free. Finally, Section 7 presents some important related work and Section 8 concludes this paper.

2 Motivation

As described in the introduction, this paper aims to present a decision support system for crime investigators to synthesise automatically plausible scenarios from available evidence and to analyse interactively the synthesised scenarios. This raises two questions: why would such a software system be useful and how does it work. This section aims to answer the former question while the remainder of this paper answers the latter.

2.1 Miscarriages of Justice

In the late 80s, a string of high profile miscarriages of justice shook the foundations of the British legal system [49]. In 1991, the Runciman Commission was established with the following term of reference:

> "To examine the effectiveness of the criminal justice system in England and Wales in securing the convictions of those guilty of criminal offences and the acquittal of those who are innocent having regard to the efficient use of resources, and in particular to consider whether changes are needed in:
>
> 1. The conduct of police investigations,
> 2. The role of the prosecutor,
> 3. The role of experts,
> 4. The arrangements for the defence,
> 5. The opportunities for an accused person to state his position,
> 6. The power of the courts in directing proceedings,
> 7. The role of the court of appeal,

8. The arrangements for considering and investigating miscarriages of justice."

The system proposed here intends to deal in particular with points 1 and 3. In the wake of the Runciman commission, a significant body of knowledge has been produced analysing the potential for errors in criminal investigations and prosecutions. Later on, the establishment of the Criminal Cases Review Commissions in England and Scotland provided extensive case studies in addition to the investigations into the high profile cases of wrongful convictions such as the Birmingham Six or the Guildford Four.

One recurrent theme in these studies is the problem of premature case theories. Instead of establishing in a neutral fashion what has happened, police officers tend to decide at a very early stage of an investigation on the most likely suspects, and from then on investigate *against* them [43]. Or in the words of David Dixon [14]:

> "If any factor in investigative practice had to be nominated as most responsible for leading to miscarriages of justice, it would have to be the tendency for investigators to commit themselves to belief in a suspect's guilt in a way that blinds them to other possibilities"

The use of this sort of "case theories" is probably inevitable [32]. The problem is therefore not the fact that case theories are used at all, but rather the *premature convergence* to a single theory without proper consideration of alternatives. As Greer argues [19]:

> "[...] no criminal justice system could work without them. The dangers stem instead from the highly charged atmosphere surrounding an investigation, the haste with which the theory has been formed and the tenacity with which the police have clung to their original view in spite of strong countervailing evidence."

2.2 Limitations of human investigators

The miscarriages of justice described above are not due to any limitations that are specific to crime investigators. They are due to limitations of all human decision makers facing complex problems.

The crucial problem of premature convergence to a particular theory to explain the available evidence in an investigation can be attributed to the phenomenon of *cognitive dissonance*. Cognitive dissonance [18] is a bias of human decision makers in favour of learning information that confirms their preconceptions over information that contradicts them.

These problems are reinforced by the professional culture of the police service. Work is done properly, and a case solved, if a suspect gets convicted. This orientation towards positive results favours an "inductivist" ethos, where those pieces of evidence that point towards the guilt of the main suspect are seen as more valuable than those that would "falsify" the leading hypothesis. While the police service might pay lip service to a falsificationist model of rationality ("asking witnesses to come forward to eliminate them from the inquiry") existing reward structures make it difficult to implement this in practice. Our proposed system accounts for this by combining a "backchaining" abductivist model of reasoning with a "forward chaining" model that is based on the idea of indirect proof, sidestepping the issue of falsification and induction in a universe with only finitely many alternatives.

2.3 Towards methodologies for crime investigation

Irving and Dunningham address possible solutions to the problem identified above [22]. They argue for the need to improve officer's reasoning and decision-making by challenging the "common sense" about criminals and crimes and the detective's crafts "working rules about causation, about suspicion and guilt, about patterns of behaviour and behavioural signatures."

Although there is no consensus within the wider community of crime investigators as to what constitutes an appropriate crime investigation methodology to accomplish these improvements, researchers within the forensic science community increasingly argue for the adoption of a hypothetico-deductive method, which is commonly employed in scientific research [23].

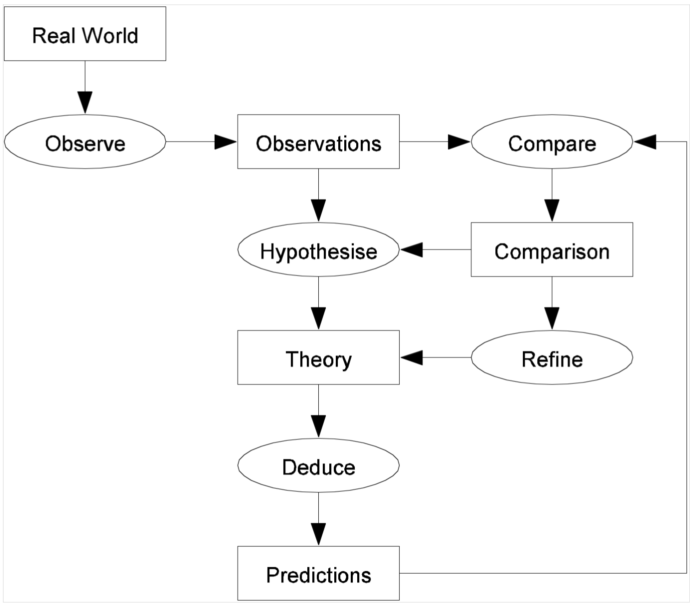


Figure 1: The hypothetico-deductive method

This method, depicted in Figure 1, is designed to formulate theories to explain complex behaviour in the real world. Based on observations of the behaviour of interest, theories explaining them are hypothesised. With such theories, predictions that should be observable if the hypothesised theory is true can be deduced. If the predictions based on a theory do not match the observations, that theory is disproved. If the predictions match the observations sufficiently closely, the plausibility of the theory is confirmed (and, it can then be further refined). As centuries of scientific practice have shown, the hypothetico-deductive method is a paradigm to help subjective humans study the world objectively. It achieves this by providing a self-correction mechanism whereby false hypothesis and viewpoints should eventually be exposed and only true ones remain.

The way in which this paradigm fits with crime investigations is obvious. The relevant part of the real world consists of the crime related locations, people, objects and events. The observations correspond to evidence, the theories are plausible scenarios and the predictions are related to possible additional evidence (or absence of evidence). However, the application of the hypothetico-deductive method is not straightforward.

The formulation of hypothetical scenarios and the deduction of new evidence to test these theories require both experience and careful formal analysis.

2.4 A case for computer aided crime investigation

Although the hypothetico-deductive and similar methodologies may certainly improve the accuracy of crime investigations, they fail to address a number of concerns raised in Section 2.2. First, any proper application of the hypothetico-deductive methodology requires significant discipline that can not be proceduralised or taught easily. Second, it implicitly relies on a large body of knowledge and experience that is only available to senior investigators. And finally, the methodology does not account for any human bias due to institutional reward systems or the emotional commitment of the “chase”.

Decision support systems, however, are very effective in analysing information systematically and objectively. They can employ substantial bodies of expert knowledge. And, they are unaffected by institutional or emotional bias. Therefore, a novel type of decision support system has been devised to support the activities of human crime investigators. This system is capable of synthesising case theories, or so-called scenarios, to explain the available evidence, and it provides a means of visualising and comparing these scenarios.

3 System Overview

3.1 Objectives and methodology

The main goal of this work is the construction of a DSS that aids in crime investigation efforts by constructing and analysing a space of alternative theories with only limited user intervention. Specifically, the decision support system is devised to find:

- The set of all scenarios that explain a given set of available evidence. *Scenarios* are descriptions of a combination of situations and events. An example of a scenario is shown in Figure 3, which describes how a suicidal person kills himself by hanging and what evidence the associated events generate.
- The hypotheses supported by scenarios that explain the available evidence. *Hypotheses* are important features of a presumed crime, such as type of death and characteristics of the perpetrator. The aforementioned scenario, for instance, entails the “suicide” hypothesis.
- Additional pieces of evidence that could be found if a certain scenario/hypothesis is true. For example, in the aforementioned scenario, a further examination into the state of mind of the victim before his death by a psychologist (e.g. by reading through the victim’s diary and talking with relatives) can help confirm a particular scenario.
- Additional investigative actions by means of which evidence can uncovered that may help differentiate between two or more hypotheses.

This work employs an *abductive diagnosis* approach to achieve the objectives set out above. Abductive diagnosers determine the conditions of a physical system or world under investigation by comparing observations predicted by models with observations extracted from the real-world [8].

The models generated by an abductive diagnoser are synthesised by means of a knowledge base of *first principles*. As opposed to the heuristic rules (of thumb) normally found in expert systems, first principles are generally applicable domain rules that are independent from the decision procedure in which they are used. In this work, the first principles are expressed by means of causal rules describing how some states and events are triggered by other known or assumed states and events.

The possible causes of a given set of available evidence are inferred by means of an abductive inference procedure. These causes form the hypothetical scenarios describing plausible crimes. Potential additional evidence that may confirm or contradict these scenarios is then deduced using the same causal rules.

This abductive, first-principles based approach recognises that while the individual scenarios encountered in a major crime investigation may be virtually unique and vary widely, the underlying domain knowledge on evidence and the types of events that create it are not. It also encourages a principled hypothetico-deductive investigative methodology because it hypotheses all (known) possible causes of the available evidence, composes these causes into plausible scenarios and deduces additional evidence from the plausible scenario. This promotes consideration of many scenarios, instead of individual ones, in deciding on future investigative actions. Finally, the approach also allows making expert domain knowledge available to less experienced investigators. As such, the system described in this paper provides a useful means to help address the specific complexities of investigating major (non-volume) crimes outlined in Section 2.

3.2 Architecture

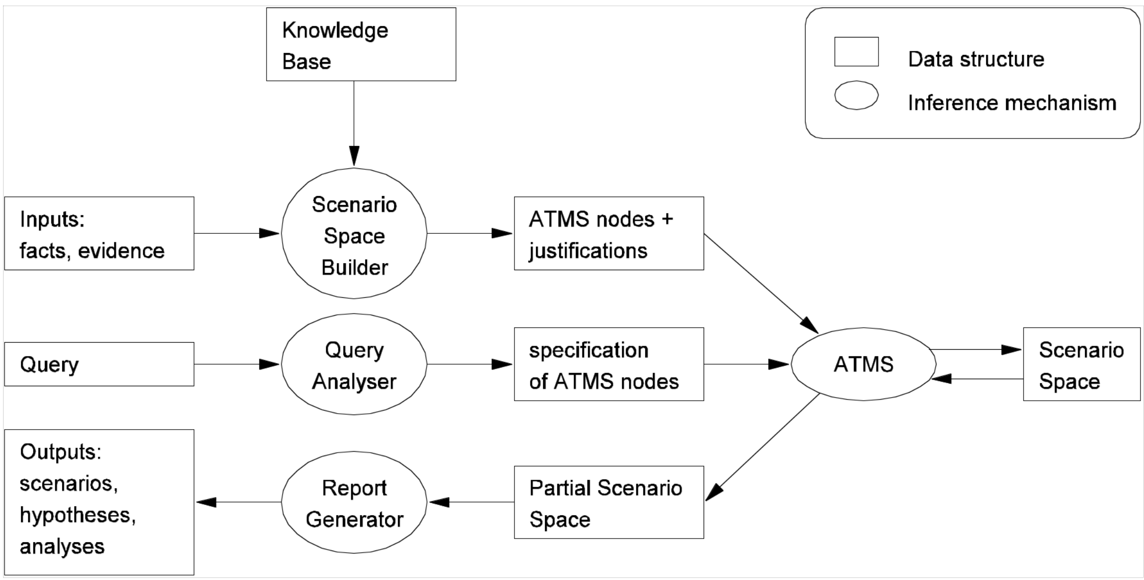


Figure 2: System architecture

The overall architecture of the DSS is show in Figure 2. The central inference mechanism in this architecture is an assumption based truth maintenance system (ATMS). An *ATMS* is a mechanism that enables a problem solver to make inferences under different hypothetical conditions by maintaining the assumptions that each piece of information and each inference depends on [10].

The ATMS is employed to maintain a *scenario space*. The scenario space is a concise data structure that contains all possible scenarios that explain the available evidence. It is initially constructed from the initial set of given *facts* and *evidence* by means of a *knowledge base*.

Once constructed, the scenario space is analysed through a series of *queries*. Queries are questions about the scenario space. Their answers are computed by extracting relevant parts from the scenario space and reported back in an understandable format. To interface between the human and scenario space, a *query analyser* translates standard types of user queries into a specification of ATMS nodes of interest, and a *report generator* provides the means to represent a partial scenario space back to the user.

The next two sections describe the inner workings of this architecture.

4 Knowledge Representation

4.1 Scenarios

Scenarios describe events and situations that may have occurred in the real-world. They form possible explanations for the evidence that is available to the crime investigator and support certain hypotheses under consideration.

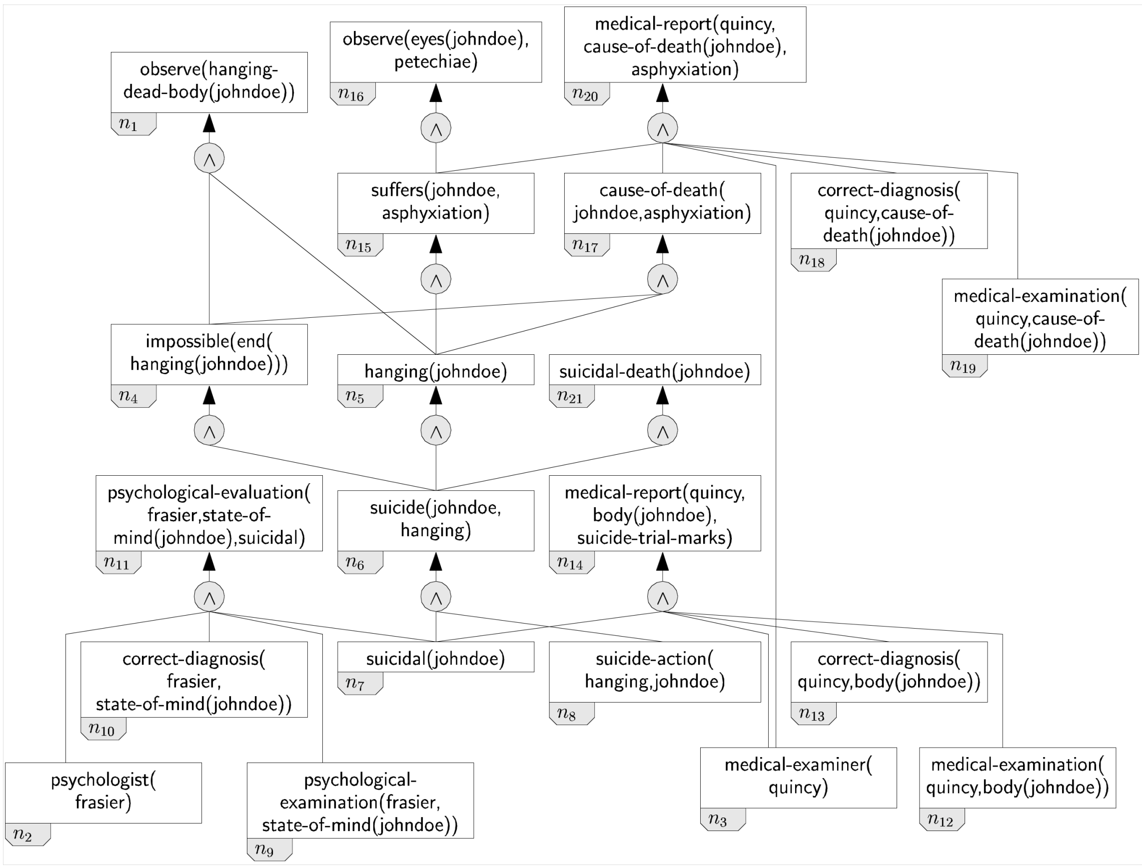


Figure 3: Sample scenario: suicide by hanging

Within the DSS, scenarios are represented by means of predicates denoting events and states, and causal relations between these events and states. The causal relations, which enable the scenarios to explain evidence and support hypotheses, are represented by hyperarcs between nodes containing the predicates. The *causal hypergraphs* shown in Figure 3 represents a sample scenario of a suicide by hanging. This scenario contains five pieces of evidence:

- n_1: A hanging corpse of a person identified as johndoe has been found.
- n_{11}: A report by a psychologist identified as frasier (n_{15}) stating that johndoe may have been suicidal prior to his death.
- n_{14}: The observation of suicide trial marks on the body of johndoe.
- n_{16}: The body of johndoe exhibits signs of petechiae[1] .
- n_{20}: A report by a medical examiner identified as quincy (n_7) stating that the cause of death of johndoe was asphyxiation.

There are many possible combinations of events and states that may lead to this set of evidence, and the scenario of Figure 3 shows one of them. It demonstrates how the first three pieces of evidence may be explained by suicide by hanging. The hanging corpse (n_1) and the assumed cause of death (n_{20}) are the consequents of johndoe's hanging (n_5),

[1]Petechiae are small red to purple spots on the eyes or skin. Petechiae may be caused by certain diseases and asphyxiation.

which he was unable (unwilling) to end (n_4). The petechiae is caused by asphyxiation (n_{15}) resulting from the hanging. johndoe's suicide by hanging requires that johndoe is suicidal (n_7) and the last two pieces of evidence are a consequence of his suicidal state.

Generally speaking, all scenarios considered in this work are represented by means of a hypergraph such as the one presented in Figure 3. Thus, scenarios are formally defined as follows:

Definition 1 A scenario is a directed acyclic hypergraph $\langle V,E \rangle$, where V is a set of events and states, and E is a set of directed hyperarcs. Each hyperarc in E connects a set of events and states from V to another event/state in V.

4.2 Types of information

In an abductive reasoner, different types of information are employed. Some information is certain, i.e. known to be true, whereas other information is uncertain, i.e. merely presumed to be true. Some information is explicable, i.e. causes for its truth can be inferred, whereas other information is inexplicable, i.e. causes for its truth can not be inferred or their explanations are irrelevant. This section describes the types of information considered in this paper and their role in the abductive reasoner, following as much as possible the terminology proposed by Poole [35]

Table 1: Certainty and explicability of information

Explicability	Certainty	
	Certain	Uncertain
Explicable	Fact	Hypothesis and inferred events and states
Inexplicable	Evidence	Assumption

As shown in Table 1, four different types of information can be identified on the basis of these two distinctions.

Facts are pieces of inexplicable, certain information. Typical examples include nodes n_2 and n_3 in the scenario of Figure 3, which denote that frasier is a psychologist and quincy is a medical examiner. These pieces of information are deemed basic truths that need not be explained further. Note that *Investigative actions* performed by an investigator are a special type of fact. They refer to activities by the investigator(s) aimed at collecting additional evidence

Evidence is information that is certain and explicable. Typical examples include nodes n_1 and n_{16} in the scenario of Figure 3, which denote that the hanging corpse of johndoe has been found and that it exhibits petechiae. Evidence is deemed certain because it can be observed by the human user and it is explicable because its possible causes are of interest to the user.

Assumptions are uncertain and inexplicable information. Typical examples include nodes n_{19} (quincy determines the cause of death of johndoe), n_{18} (quincy makes the correct diagnosis of the cause of death of johndoe) and n_7 (johndoe was suicidal). Generally speaking, it is not possible to rely solely on facts when speculating about the plausible causes of the available evidence. Ultimately, the investigator has to presume that certain information at the end of the causal paths is true, and such pieces of information are called assumptions. In this work, three types of assumptions are distinguished:

- *Default assumptions* describe information that is normally presumed to be true. In theory, the number of plausible scenarios that explain a set of available evidence is virtually infinite, but many of these scenarios are based on very unlikely presumptions. Default assumptions aid in the differentiation between such scenarios by expressing the most likely features of events and states in a scenario. A typical example of a default assumption is the presumption that a doctor's diagnosis of the cause of death of person is correct (e.g. n_{18}).
- *Conjectures* are the unknown causes of certain feasible scenarios (e.g. n_7). Unlike default assumptions, conjectures are not employed to differentiate between the relative likelihood of scenarios.
- Uncommitted *investigative actions*, i.e. possible but not yet performed activities aimed at collecting additional evidence, are also treated as assumptions. At any given stage in the investigation, it is *uncertain* which of the remaining uncommitted investigative actions will be performed. The reasoning required to perform such an action involves looking at its consequences instead of its causes, and therefore they are *not* (causally) *explicable*. As such, investigative actions assume a similar role as default assumptions and conjectures: i.e. they are employed to speculate about the plausible (observable) consequences of a hypothetical scenario.

The information in the remaining category is uncertain and explicable. It includes *uncertain states*, such as n_4 (johndoe was unable to end his hanging), *uncertain events*, such as n_{15} in Figure 3 (johndoe asphyxiated) and *hypotheses*, such as n_{21} (johndoe's death was suicidal).

4.3 Scenario fragments

The objective of this work is to automatically generate scenarios that could have caused the available evidence in an investigation. This is a difficult task since there may be many, potentially rare, scenarios that can explain the unique circumstances of an individual case.

The approach proposed here is based on the observation that the constituent parts of the scenarios are not normally unique to that scenario. The scenario of Figure 3, for instance, describes that the asphyxiation of johndoe causes petechiae on the body of johndoe. This causal relation applies to most humans, irrespective of whether the asphyxiation occurs in the context of a hanging or a suicide. Thus, the causal rule,

$$\text{asphyxiation}(P) \rightarrow \text{petechiae}(\text{eyes}(P))$$

is generally applicable and can be instantiated in all scenarios involving evidence of petechiae or possible asphyxiation of person.

Thus, the knowledge base consists of a set of such causal rules, called *scenario fragments*. For example, the rule

```
if {
  suffers(P,C),
  cause-of-death(C,P),
  medical-examiner(E)
} assuming {
  determine(E,cause-of-death(P)),
  correct-diagnosis(E,cause-of-death(P))
} then {
  cod-report(E,P,C)
}
```

states that if a person `P` suffers from ailment or injury `C`, `C` is the cause of death of `P`, and there is a medical examiner `E`, and assuming that `E` determines the cause of death of `P` and makes the correct diagnosis, then there will be a piece of evidence in the form of a cause of death report indicating that according to `E`, the cause of death of `P` is `C`.

The causal relations between assumptions, states and events are formalised in *scenario fragments*.

Definition 2 A *scenario fragment* μ is a tuple $\left\langle V, V^s, V^t, \Phi^s, \Phi^t, A \right\rangle$ where

- $V = \{v_1,...,v_l\}$, $V^s(m) = \{p_1^s,...p_m^s\}$ and $V^t(m) = \{v_1^t,...,v_n^t\}$ are sets of *variables*,
- $\Phi^s(m) = \{\phi_1^s,...,\phi_v^s\}$ is a set of relations, called *preconditions*, whose free variables are elements of $V \square V^s$,
- $\Phi^t(m) = \{\phi_1^t,...,\phi_w^t\}$ is a set of relations, called *postconditions*, whose free variables are elements of $V \square V^t$, and
- $A(m) = \{a_1,...,a_t\}$ is a set of relations, called *assumptions*

such that for i=1,...,w:

$$\forall v_1,\ldots,\forall v_l, \forall v_1^s, \ldots \forall v_m^s, \exists v_1^t, \ldots \exists v_n^t, \left(\phi_1^s \wedge \ldots \wedge \phi_v^s \rightarrow \left(a_1 \wedge \ldots \wedge a_t \rightarrow \phi_i^t\right)\right)$$

Hence, the aforementioned sample model fragment matches this definition 4.3 as follows:

$V^s = \{\}$

$V^t = \{\}$

$V = \{\}$

$\Phi^s =$ `{suffers(P,C),cause-of-death(P,C),medical-examiner(E)}`

$A =$ `{determine(E,cause-of-death(P)), correct-diagnosis(E,cause-of-death(P))}`

$\Phi^t =$ `{cod-report(E,P,C)}`

4.4 Inconsistencies

Some states and events are inconsistent with one another. For example, a person can not kill himself both with such an intention (i.e. in a suicide) and without this intention (i.e. in an accidental self-killing). Such knowledge is represented by means of *inconsistencies*.

For instance, the following inconsistency states that a person `P` can not both commit suicide and be hanged by someone else:

```
inconsistent {
    suicide-action(hanging,P),
    is-hanged(P)}
```

Similarly, the following inconsistency states that a person `P` can not commit an action `A` to kill him/herself and as an autoerotic activity:

```
inconsistent {
    suicide-action(A,P),
    autoerotic-action(A,P)}
```

Formally, inconsistencies are defined as follows:

Definition 3 An inconsistency is a tuple $\langle V, \Phi \rangle$ where $V = \{v_1, \ldots v_l\}$ is a set of variables and $\Phi = \{\phi_1, \ldots, \phi_v\}$ is a set of relations, whose free variables are elements of P, such that:

$$\forall v_1, \ldots, \forall v_l, \left(\bigwedge_{\phi_i \in \Phi} \phi_i \right) \rightarrow \perp$$

4.5 Knowledge base

The knowledge base in the system's architecture of Figure 2 consists, at least, of the following constructs:

- *Property definitions* describe which types of predicate correspond to a symptom, fact, hypothesis or investigative action.
- A set of *scenario fragments* describing reusable component causal relations from which the scenarios are composed.
- A set of *inconsistencies* describing which combinations of states and events are impossible.

It is assumed that the set of scenario fragments does not contain any cycles as these may lead to perpetual creation of new instances of the same predicates. Within the domain of major crime investigations, this is a realistic assumption as there are usually identifiable causes for the available physical evidence. Consequently, existing work on evidence evaluation tends to use acyclic structures, such as Bayesian Networks and argument trees.

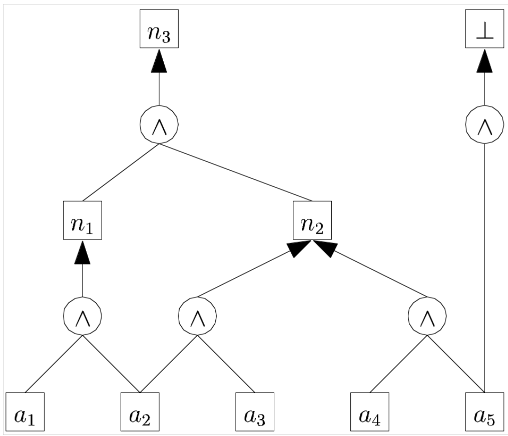
5 Inference mechanisms

The DSS employs two types of inference. One inference mechanism generates plausible crime scenarios from evidence. Once these crime scenarios are available, a set of inference mechanisms analyses them to help the investigator decide on the course of the investigation.

Since there are potentially many scenarios that produce the same evidence, and because many of these scenarios are minor variations of one another, an efficient means of storing and reasoning with them is both necessary and feasible. An assumption based truth maintenance system (ATMS) is employed in this paper to store the scenarios and the conditions under which they are valid. A summary of the functionality of the ATMS is presented in Section 5.1.

5.1 Assumption based truth maintenance

An ATMS is a mechanism that maintains how each piece of inferred information depends on presumed information and facts, and how inconsistencies arise. This section summarises the functionality of an ATMS as it is employed in this work. For more details, the reader is referred to the original papers [10, 11].



n_3 victim's body shows signs of petechiae

n_1 victim asphyxiated

n_2 victim suffered poisoning

a_1 perpetrator employed strangulation

a_2 perpetrator attempted to kill victim

a_3 perpetrator employed poisoning

a_4 victim consumed excessive dose

a_5 victim medicated self

Figure 4: Sample ATMS

In an ATMS, each piece of information of relevance to the problem solver is stored as a *node*. Some pieces of information are not known to be true and cannot be inferred from other pieces of information. The plausibility of these is determined through the inferences made from them. In the ATMS, they are represented by a special type of node, called *assumption*. Figure 4a is a graphical representation of a sample ATMS with 3 nodes (n_1, n_2, n_3) and 5 assumptions $(a_1, ..., a_5)$ and Figure 4b shows a sample interpretation of the nodes in this ATMS.

Inferences between pieces of information are maintained within the ATMS as inferences between the corresponding nodes. In its extended form (see [11] or [25]), the ATMS can take inferences, called *justifications* of the form $n_i \wedge \ldots \wedge n_j \wedge \neg n_k \wedge \ldots \wedge \neg n_l \rightarrow n_m$, where $n_i, ..., n_j, n_k, ..., n_l, n_m$ are nodes (and assumptions) representing things that the problem solver is interested in. The sample ATMS of Figure 4 contains the following justifications:

$$a_1 \wedge a_2 \rightarrow n_1 \qquad\qquad n_1 \rightarrow n_3$$

$$a_2 \wedge a_3 \rightarrow n_2 \qquad\qquad n_2 \rightarrow n_3$$

$$a_4 \wedge a_5 \rightarrow n_2$$

An ATMS can also take justifications, called *nogoods* that have lead to an inconsistency, i.e. justifications of the form $n_i \wedge \ldots \wedge n_j \wedge \neg n_k \wedge \ldots \wedge \neg n_l \rightarrow \perp$. The latter nogood implies that at least one of the statements in $\{n_i, ..., n_j, \neg n_k, ..., \neg n_l\}$ must be false. The sample ATMS of Figure 4 contains the following nogood:

$$a_5 \rightarrow \perp .$$

Based on the given justifications and nogoods, the ATMS computes a *label* for each (non-assumption) node. A label is a set of *environments* and an environment is a set of assumptions. An environment *A* depicts a possible world where all the assumptions in *A* are true. The label *L*(*n*) of a node *n* describes all possible worlds in which *n* can be true. For reasons of efficiency and effectiveness, the label computation algorithm of the ATMS guarantees that each label is:

- *Sound*: Each environment describes a possible world that logically entails the node. In other words, the presumption that all assumptions in an environment from the label of a node are true is a sufficient condition to derive that node. Formally, *L*(*n*) is sound if

$$\forall E \in L(n), \left[\left(\wedge_{n_i \in E} n_i \right) \wedge \left(\wedge_{\neg n_i \in E} \neg n_i \right) \right] ?\ n$$

- *Consistent*: No environment in the label of a node describes an impossible world (i.e. a world from which $\perp$ logically follows). Formally, *L*(*n*) is consistent if

$$\forall E \in L(n), \left[\left(\wedge_{n_i \in E} n_i \right) \wedge \left(\wedge_{\neg n_i \in E} \neg n_i \right) \right] - \perp$$

- *Complete*: The label describes all possible worlds. In other words, if there is a consistent conjunction of assumptions that entail the node, then the set of those assumptions or a subset is included in the label of the node. Formally, a label is complete if

$$\forall E, \exists E' \in L(n), \left[\left(\wedge_{n_i \in E} n_i \right) \wedge \left(\wedge_{\neg n_i \in E} \neg n_i \right) ?\ n \right] \rightarrow \left(E' \subseteq E \right)$$

- *Minimal*: The label does not contain possible worlds that are less general than one of the other possible worlds it contains (i.e. environments that are supersets of other environments in the label). Formally, a label is minimal if

$$\forall E \in L(n), \neg \exists E', \left[\left(\wedge_{n_i \in E'} n_i \right) \wedge \left(\wedge_{\neg n_i \in E'} \neg n_i \right) ?\ n \right] \wedge \left(E' \subseteq E \right)$$

In the sample ATMS of Figure 4, the labels of the nodes are as follows:

$$L(n_1) = \{\{a_1, a_2\}\}$$
$$L(n_2) = \{\{a_2, a_3\}\}$$
$$L(n_3) = \{\{a_1, a_2\}, \{a_2, a_3\}\}$$
$$L(\perp) = \{\{a_5\}\}$$

The concepts of soundness, consistency, completeness and minimality can be illustrated by means of the the label of n_3:

- $L(n_3)$ is sound because it can be shown that n_3 follows from both environments:

  $$a_1 \wedge a_2 \rightarrow n_1 \rightarrow n_3$$
  $$a_2 \wedge a_3 \rightarrow n_2 \rightarrow n_3$$

- $L(n_3)$ is consistent because neither $\{a_1,a_2\}$ nor $\{a_2,a_3\}$ entails $\perp$.
- n_3 is entailed by each of the following consistent environments:

  $$\{a_1, a_2\}, \{a_2, a_3\}, \{a_1, a_2, a_3\}, \{a_1, a_2, a_4\}, \{a_2, a_3, a_4\}, \{a_1, a_2, a_3, a_4\}$$

  Note that $\{a_5\}$ is an inconsistent environment because $\{a_5\} \in L(\perp)$. Therefore, environments that entail n_3 but include a_5, such as $\{a_4, a_5\}$ are excluded from the above list of environments. Because each of environments in the above list is a superset of one of the environments in $L(n_3)$, $L(n_3)$ is said to be complete.
- Finally, $L(n_3)$ is minimal because $\{a_1, a_2\} \not\subset \{a_2, a_3\}$ and $\{a_2, a_3\} \not\subset \{a_1, a_2\}$.

5.2 Synthesis of the scenario space

5.2.1 Intuition

The goal of the scenario space builder is to construct plausible crime scenarios by instantiating the knowledge base of scenario fragments and inconsistencies into an ATMS. This is accomplished in four phases:

1. *Initialisation phase*: An ATMS that contains one node per piece of available evidence is created.
2. *Backward chaining phase*: The ATMS is extended by adding all plausible causes of the available evidence. For each possible unification of a consequent of a model fragment with a node already in the ATMS,
   - the antecedents and assumptions of that model fragment are instantiated,
   - a node is added to the ATMS for each antecedent instance that does not already have one,
   - an assumption node is added to the ATMS for each assumption instance that does not already have one,
   - a justification is added to the ATMS, from the nodes corresponding to the antecedent and the assumption nodes corresponding to the assumptions, to the node corresponding to the consequent.

   This process is repeated until all possible unifications of individual model fragment consequents with nodes in the ATMS are exhausted. After the backward chaining phase, all plausible scenarios explaining the available evidence are instantiated in the ATMS.

3. *Forward chaining phase*: The ATMS is then extended by adding all possible consequences of the plausible scenarios. For each possible unification of the set of antecedents of a model fragment with a set of nodes already in the ATMS,
   - the assumptions and consequents of that model fragment are instantiated,
   - an assumption node is added to the ATMS for each assumption instance that does not already have one,
   - a node is added to the ATMS for each consequent instance that does not already have one,
   - for each consequent instance, a justification is added to the ATMS, from the nodes corresponding to the antecedent and the assumption nodes corresponding to the assumptions, to the node corresponding to the consequent instance.

   This process is repeated until all unifications of model fragment antecedents with sets of nodes in the ATMS are exhausted. After the forward chaining phase, all possible consequences of plausible scenarios, including potential evidence and hypotheses, are instantiated in the ATMS.
4. *Consistency phase*: In this final phase, inconsistent combination of states and events are denoted as nogoods. This involves instantiating the inconsistencies from the knowledge base based on information in the ATMS and marking them as justifications for the nogood node.

5.2.2 Formal algorithm

This approach is formalised by algorithm 5.1. The algorithm, **generateScenarioSpace**(O,F,**S**,**I**) takes a set of evidence O, a set of facts F and a knowledge base containing a set of scenario fragments **S** and a set of inconsistencies **I** as its inputs. It expands on an existing composition modelling algorithm devised for the automated construction of ecological models [28].

Algorithm 5.1 : generateScenarioSpace$(O, F, \mathbf{S}, \mathbf{I})$

comment: Initialisation phase

$\theta \leftarrow$ new ATMS;

for each $e \in O$ **do** ADD-NODE(θ, e);

for each $f \in F$

do $\begin{cases} \text{ADD-NODE}(\theta, \mathrm{f}); \\ \text{ADD-JUSTIFICATION}(\theta, f, \{\}); \end{cases}$

comment: Backward chaining phase

for each $\langle V, V^s, V^t, \Phi^s, \Phi^t, A \rangle \in \mathbf{S}, \exists\sigma, \text{MATCH}(\langle V \cup V^t, \Phi^t \rangle, \theta, \sigma)$

do $\begin{cases} \mathrm{J} \leftarrow \varnothing; \\ \textbf{for each } v \in V^s \textbf{ do } \sigma \leftarrow \sigma \cup \{v / \text{GENSYM}(\ )\}; \\ \textbf{for each } \phi \in \Phi^s \\ \quad \textbf{do } \begin{cases} n \leftarrow \text{ADD-NODE}(\theta, \sigma\phi); \\ J \leftarrow J \cup \{n\}; \end{cases} \\ \textbf{for each } a \in A \\ \quad \textbf{do } \begin{cases} n \leftarrow \text{ADD-ASSUMPTION}(\theta, \sigma\mathrm{a}); \\ J \leftarrow J \cup \{n\}; \end{cases} \\ m \leftarrow \text{ADD-NODE}(\theta, (\sigma\langle V, V^s, V^t, \Phi^s, \Phi^t, A \rangle)) \\ \text{ADD-JUSTICIATION}(\theta, m, J); \\ \textbf{for each } \phi \in \Phi^t \textbf{ do } \text{ADD-JUSTIFICATION}(\theta, \eta(\sigma\phi), \{m\}); \end{cases}$

comment: Forward chaining phase

for each $\langle V, V^s, V^t, \Phi^s, \Phi^t, A \rangle \in \mathbf{S}, \exists\sigma, \text{MATCH}(\langle V \cup V^s, \Phi^s \rangle, \theta, \sigma)$

do $\begin{cases} \mathrm{J} \leftarrow \varnothing; \\ \textbf{for each } \phi \in \Phi^s \textbf{ do } J \leftarrow J \cup \{\sigma\phi\}; \\ \textbf{for each } a \in A \\ \quad \textbf{do } \begin{cases} n \leftarrow \text{ADD-ASSUMPTION}(\theta, \sigma\mathrm{a}); \\ J \leftarrow J \cup \{n\}; \end{cases} \\ m \leftarrow \text{ADD-NODE}(\theta, (\sigma\langle V, V^s, V^t, \Phi^s, \Phi^t, A \rangle)) \\ \text{ADD-JUSTIFICATION}(\theta, m, J); \\ \textbf{for each } v \in V^t \textbf{ do } \sigma \leftarrow \sigma \cup \{v / \text{GENSYM}(\ )\}; \\ \textbf{for each } \phi \in \Phi^t \\ \quad \textbf{do } \begin{cases} n \leftarrow \text{ADD-NODE}(\theta, \sigma\phi); \\ \text{ADD-JUSTIFICATION}(\theta, n, \{m\}); \end{cases} \end{cases}$

comment: Consistency phase

for each $\langle V, \Phi \rangle \in \mathbf{I}, \exists\sigma, \text{MATCH}(\langle V, \Phi \rangle, \theta, \sigma)$

do $\begin{cases} J \leftarrow \varnothing; \\ \textbf{for each } \phi \in \Phi \textbf{ do } J \leftarrow J \cup \{\sigma\phi\}; \\ \text{ADD-NOGOOD}(\theta, \mathrm{J}); \end{cases}$

The algorithm works as follows:

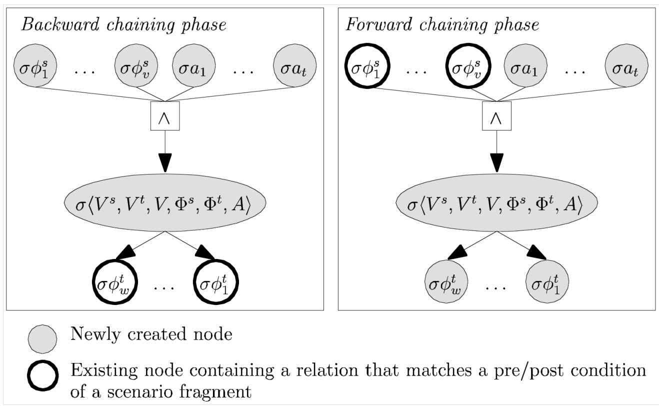


Figure 5: Scenario fragment instantiation in the ATMS

1. *Initialisation phase*: Here, an ATMS θ is created and initialised by adding a node for each given piece of evidence in *O* and one for each given fact in *F*. Each node corresponding to a fact is justified by the empty set, thus indicating that they are true under all circumstances.
2. *Backward chaining phase*: All combinations of possible events and states that can possible produce the given pieces of available evidence *O* are reconstructed. That is, for each scenario fragment whose postconditions match relations in the ATMS θ (that is, the scenario fragments $\langle V, V^s, V^t, \Phi^s, \Phi^t, A\rangle$ for which a substitution σ exist that maps the postconditions Φ^t to relations referred to by nodes in the ATMS), a new set of nodes and justifications is added to the ATMS as follows:
   - A node *m* is added to θ denoting the application of the scenario fragment. Each node *n* to which the scenario fragment was matched (i.e. each node $\eta(\sigma\phi)$ with $\phi \in \Phi^t$) is justified in the ATMS by $n \leftarrow m$.
   - For each variable $v \in V^S$ a new constant *c* is created and the substitution $\{v/c\}$ is added to σ.
   - A node denoting $\sigma\phi$ is added for each precondition $\phi \in \Phi^S$ and an assumption denoting σa is added for each assumption $a \in A$. The conjunction of these newly created nodes is added as a justification of node *m* (the node denoting the instantiation of the scenario fragment).

   The resulting nodes and justifications are shown graphically in Figure 5. Initially, θ is only populated with the pieces of evidence given in *O* and the algorithm works its way backwards to determine the potential sources of those pieces of evidence as described in the knowledge base of scenario fragments **S**.
3. *Forward chaining phase*: All the pieces of evidence and hypotheses that can be consequences of plausible scenarios generated in the backward chaining phase are extrapolated. For each scenario fragment whose preconditions match relations in the ATMS θ (i.e each scenario fragments $\langle V, V^S, V^t, \Phi^S, \Phi^t, A\rangle$ for which a substitution σ exist that maps the postconditions in Φ^s to relations referred to by nodes in the ATMS), a new set of nodes and justifications is added to the ATMS as follows:

- For each variable $v \in V^t$ a new constant c is created and the substitution $\{v/c\}$ is added to σ.
- A node denoting $\sigma\phi$ is added for each postcondition $\phi \in \Phi^t$ and an assumption denoting σa is added for each assumption $a \in A$.
- A node m is added to θ denoting the application of the scenario fragment. This new node is justified by the conjunction of the instances of the relations in Φ^s and A:

$$m \leftarrow \left[\left(\wedge_{(\phi \in \Phi^s)} \sigma\phi\right) \wedge \left(\wedge_{(a \in A)} \sigma a\right)\right]$$

  Each postcondition instance is justified by the new node m.

The resulting nodes and justifications are also shown graphically in Figure 5.

4. *Consistency phase*: In the final stage, the inconsistencies are instantiated and reported to the ATMS θ. More specifically, for each inconsistency whose relations match relations in the ATMS θ (that is, each inconsistency $\langle V,\Phi \rangle$ for which a substitution σ exist that maps the relations in Φ to relations referred to by nodes in the ATMS), a nogood $\perp \leftarrow \left(\wedge_{(\phi \in \Phi)} \sigma\phi\right)$ is created.

The scenario instantiation algorithm employs a function MATCH($\langle V,\Phi \rangle,\theta,\sigma$) to find instances of the relations in the scenario fragments and inconsistencies. The function takes the following arguments: 1) a set of free variables V, 2) a set of relations Φ whose free variables are elements of V, 3) the ATMS under construction θ and a substitution σ. The substitution σ maps each variable in $v \in V$ to a constant σv and each relation $\phi \in \Phi$ to a grounded relation $\sigma\phi$ where the variables are substituted by constants.

The function is true if for each relation $\phi \in \Phi$, a node exists in the ATMS θ that denotes the grounded relation $\sigma\phi$. Formally, MATCH($\langle V,\Phi \rangle,\theta,\sigma$) is deemed true if

$$\left(V = \{v_1,\ldots,v_p\}\right) \wedge \left(\sigma = \left(v_1 / o_1,\ldots,v_p / o_p\right)\right) \wedge \left(\forall \phi_j \in \Phi, \eta\left(\sigma\phi_j\right) \in \theta\right)$$

where η is the function that maps grounded relations of interest to our problem solver (the crime scenario instantiator) to nodes and assumptions in ATMS.

The scenario space generation algorithm can be illustrated by showing how it can be employed to reconstruct the scenario introduced in Figure 3. Assume that the system is given one piece of evidence observe(hanging-dead-body(johndoe)) and two facts psychologist(frasier) and medical-examiner(quincy). The *initialisation phase* of the algorithm will simply create an ATMS with nodes corresponding to that piece of evidence and those two facts. As the facts are justified by the empty set, they are deemed true in all possible worlds. The result of the initialisation phase is shown in Figure 6.

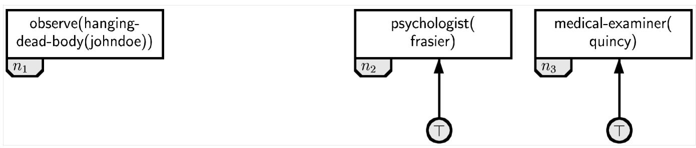


Figure 6: Scenario space generation: initialisation phase

The *backward chaining phase* then expands this initial scenario space by generating plausible causes of the available evidence by instantiating the antecedents and assumptions of scenario fragments whose consequences match nodes already in the scenario space. For example, the consequent of scenario fragment

```
if   { hanging(P),
       impossible(end(hanging(P))) }
then { observe(hanging-dead-body(P)) }
```

matches the piece of evidence already in the scenario space, and this allows the creation of new nodes corresponding to hanging(johndoe) and impossible(end(hanging(johndoe))) and a justification from the latter two nodes to the former. The result of the backward chaining phase is shown in Figure 7.

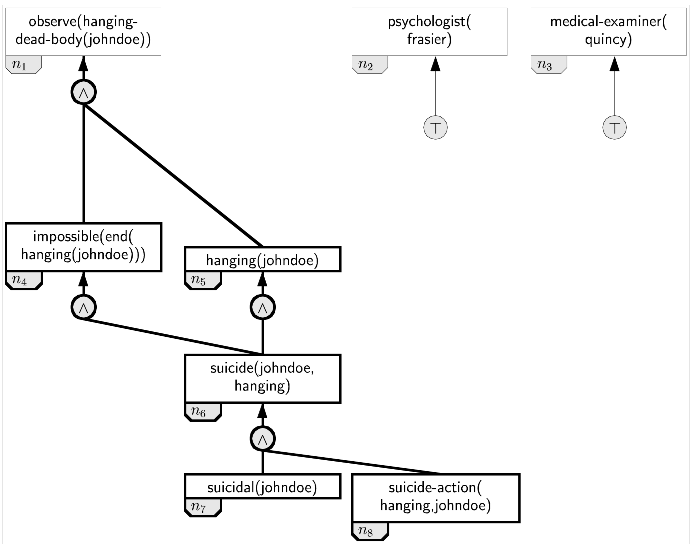


Figure 7: Scenario space generation: backward chaining phase

The *forward chaining phase* expands the scenarios created during the backward chaining phase with additional evidence that can be produced by them and the hypotheses they entail. The result of the forward chaining phase is shown in Figure 8.

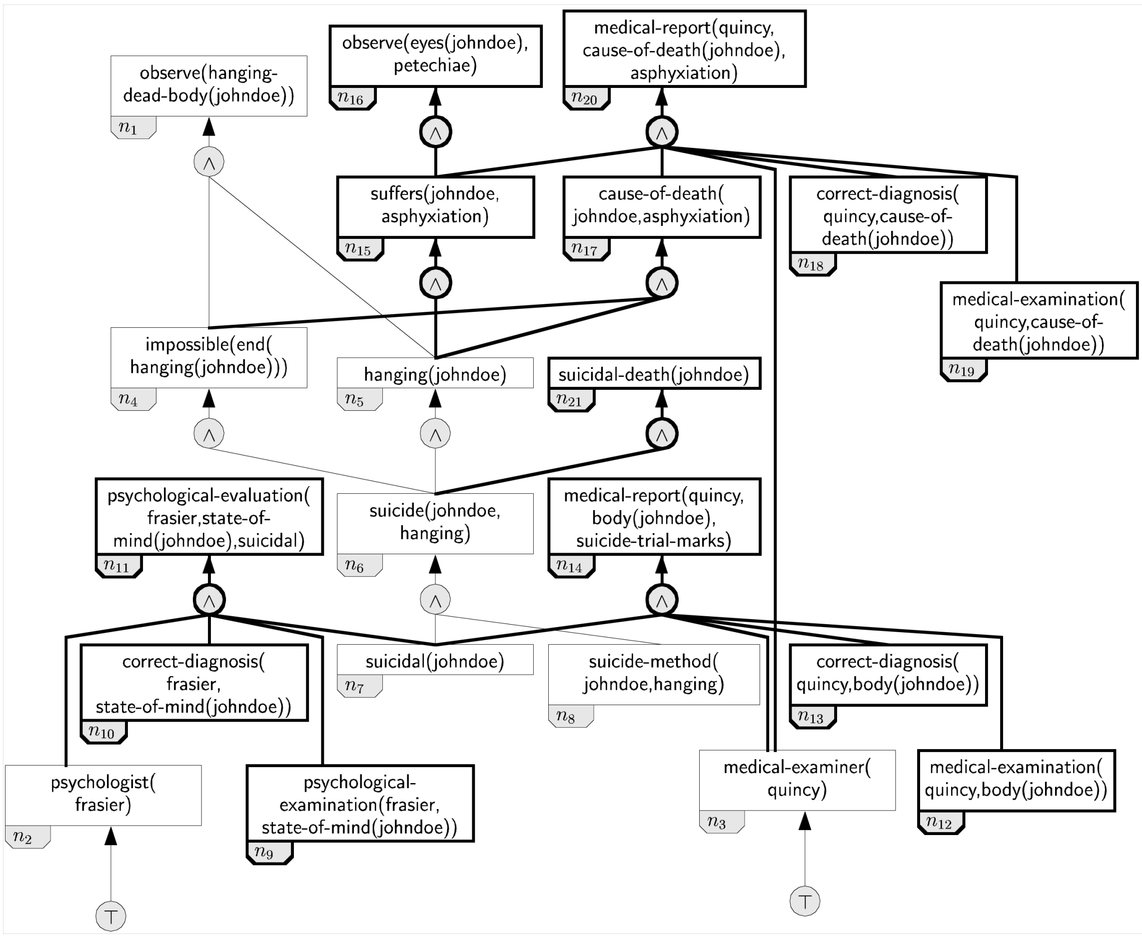


Figure 8: Scenario space generation: forward chaining phase

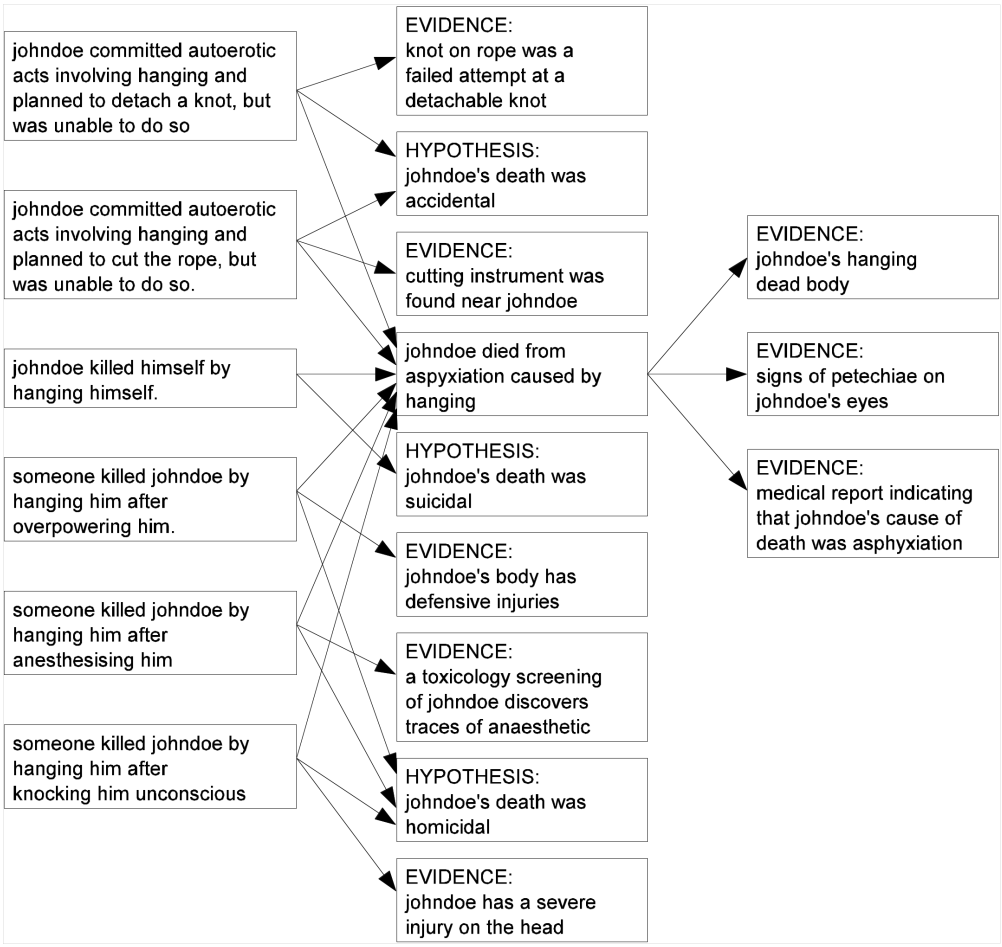


Figure 9: Overview of the scenario space

The scenario space as a whole is to large to display here. Therefore, Figure 9 presents an informal overview of the information contained in the scenario space. The blocks on the lefthand side of the figure represent sets of nodes and justifications between them that correspond to a sufficient explanation for a death by hanging. Note that, although these blocks appear to be separated, they do have a number of nodes and justification in common. Therefore, the scenario space does not only contain the 6 primitive scenarios described in Figure 9, but also combinations of them. The middle and righthand columns for Figure 9 show the possible pieces of evidence that may follow from certain scenario and the hypotheses that logically follow from scenarios. This information will be employed in the analysis phase of the application to generate useful decision support information for the human crime investigator.

5.2.3 Outline analysis of complexity

The synthesis of a scenario space of plausible crime scenarios is an important feature that makes this approach unique and enables it to answer important queries. However, the generation of a space of all scenarios that may explain the available evidence also raises concerns regarding the time and space complexity of the approach. A formal analysis of the complexity of this type of algorithm is rather sophisticated because its performance is dependent upon a substantial number of structural features of both the knowledge base and the initial set of evidence. Therefore, an outline discussion of the time and space complexity of the scenario space generation algorithm is presented instead.

The algorithm essentially performs a fixed sequence of instructions and produces a small set of nodes and justifications for each match of a scenario fragment. Therefore, its time and space requirements are proportional to the number of matches of scenario fragments, and its complexity arises from two factors: the discovery of matches of scenario fragments, and the processing of the application of scenario fragments with respect to a particular match.

Because the set of model fragments is assumed not to contain cycles, the pattern matching procedure can be organised by traversing an ordered list of model fragments, thereby processing all the instantiations of each model fragment in one go without the need for a conflict resolution mechanism. If there are m model fragments and n sets of participants and relations matching the source-participants and structural conditions, the time complexity of the backward and forward chaining iterations is $O(m \times n)$. Note that n is very hard to establish, but in general, it depends on the number of similar objects and relations in the given scenario and the degree of reusability of model fragments, which are both domain dependent.

The actual application of scenario fragments involves creating new nodes and adding justifications for the new and some existing nodes. Although the creation of new nodes requires negligible effort, the addition of justifications to nodes necessitates an update of their label, and of the nodes that they justify, in order to ensure that all labels remain sound, complete, consistent and minimal (as explained in Section 5.1). As argued in [25, 28], the dominant factor driving the computational effort required by label propagation in this case is the depth of the scenario space (i.e. the length of the longest path from an assumption or fact to a piece of evidence or hypothesis it entails), and the time complexity of scenario space synthesis is exponential with regard to this factor.

In order to determine whether this issue is a serious one, a significant number of case studies would have to be performed. However, discussions with crime investigators and forensic scientists, as well as the ongoing development of a sample knowledge base, indicate that crime scenarios tend to be explainable with causal arguments of manageable lengths (i.e. one or two digit numbers).

Storing the space of all possible scenarios may impose significant storage requirements. However, when compared to other approaches that store a space of models such as the Graph of Models [1], the approach presented here is far more economical. Because only scenario fragments are instantiated rather than entire scenarios, the common parts of different scenarios need only be stored once. In theory, if a scenario space represents m different scenarios that have $c\%$ in common, and each model has a space requirement of s, then the space complexity can be as low as $O(c \times s + (1-c) \times m \times s)$, rather than $O(m \times s)$ when all modes are stored explicitly. For large values of m and when c is not too close to 0, $c \times s + (1-c) \times m \times s \ll m \times s$.

5.3 Analysis of the scenario space

The ATMS θ constructed by the algorithm described in Section 5.2 contains a space of all scenarios that can be constructed with the knowledge base and that produce the given set of evidence O. This section shows how the information contained in this ATMS can be exploited to answer the first three types of query mentioned in Section 3.1. The approach taken herein involves translating queries into formal ATMS nodes and justifications, thus enabling the existing ATMS label propagation to answer the queries of interest.

Formally, any consistent conjunction of assumptions that entails all pieces of evidence (in θ) constitutes a possible world for the case under investigation. Any set of assumption W such that

$$\left[\bigwedge_{(a \in W)} a, \theta \nvdash \bot\right] \wedge \left[\bigwedge_{(a \in W)} a, \theta \quad e\right]$$

and the consequences of those assumptions describe a plausible crime scenario. Therefore, all the scenarios contained in the ATMS can be retrieved by computing the label for the conjunction of the pieces of evidence in O. Let n_O be an additional node added to θ and justified as:

$$n_O \leftarrow \bigwedge_{(e \in O)} \eta(e)$$

Then, the label $L(n_O)$ contains all the environments from which plausible scenarios can be produced. This knowledge enables the decision support system to answer the following questions:

- *Which hypotheses are supported by the available evidence?* Every hypothesis that follows from a plausible scenario is supported by the available evidence. That is, a hypotheses h is supported by the evidence if it follows from an environment of the label of n_O:

  $$\exists W \in L(n_O), \bigwedge_{(a \in W)} a, \theta \; ? \; \eta(h) \qquad \textbf{Eq. 1}$$

  where $\eta(e)$ refers to the node that denotes e in θ. Thus, if the label of h is not empty, then that hypothesis is supported by the available evidence.

  For instance, in the scenario space generated for the ongoing example, the labels of both n_0 and the node η(suicidal-death(johndoe) (i.e. the node representing the hypothesis that the death of johndoe was suicidal) include the following environment:

  $$W_{\text{suicide}} = \{\text{suicidal}(\text{johndoe}), \text{suicide-action}(\text{hanging}, \text{johndoe})\}$$

  According to Eq. 1 it therefore follows that the suicidal death hypothesis is supported in this case.

- *What additional pieces of evidence can be found if a certain scenario/hypothesis is true?* All the states and events, including pieces of evidence, that are logical consequence states and events in plausible scenarios are generated in the forward chaining phase of the algorithm. Therefore, θ will contain nodes representing pieces of evidence that are produced in certain scenarios but were not collected in O. As with the hypotheses, the labels of these nodes describe the environments (and hence, the scenarios) under which these pieces of evidence are expected.

  Unlike hypotheses, evidence can not be considered a logical consequence of a plausible scenario. Indeed, evidence is normally the result of an investigative action as well as an interpretation by a human investigator, some laboratory equipment or a combination of both. The former is represented by an investigative action assumptions whereas the latter is described by one or more default assumptions. In order to incorporate these considerations in the analysis, let $\mathbf{I}_\theta$

denote the set of all investigative actions in the scenario space θ and let **D** denote the set of all default assumptions in the scenario space θ.

Consider an environment W in θ that corresponds to a particular scenario in the scenario space. A piece of evidence e can then be expected in the scenario corresponding to W if a if a possible world W_e exist that supports the evidence and where W_e consists only of assumptions from W, investigative action assumptions and default assumptions:

$$\exists W_e \in L(\eta(e)), W_e / W_h \subseteq (\mathbf{I} \cup \mathbf{D}) \qquad \textbf{Eq. 2}$$

This definition can be extended to discover whether a piece of evidence can be expected given a hypothesis h if W is an environment taken from the label of h:

$$\exists W_h \in L(\eta(h)), \exists W_e \in L(\eta(e)), W_e / W_h \subseteq (\mathbf{I} \cup \mathbf{D})$$

In the ongoing example, this technique can establish that in the suicide scenario, the investigator can expect that a forensic psychological evaluation by a given forensic psychologist, say frasier, will determine that johndoe has had a suicidal state of mind prior to his death. Formally, this new piece of evidence is represented by the predicate psychological-evaluation(frasier,suicidal(johndoe)) and its label is

$$\{\{\text{suicidal}(\text{johndoe}),\\ \text{psychological-examination}(\text{frasier}, \text{state-of-mind}(\text{johndoe})),\\ \text{correct-diagnosis}(\text{frasier}, \text{state-of-mind}(\text{johndoe}))\}\}$$

The sole environment in this label consists of one conjecture suicidal(johndoe), which is also in W_{suicidal}, one investigative action assumption psychological-examination(frasier,state-of-mind(johndoe)) and one default assumption correct-diagnosis(frasier,state-of-mind(johndoe)). Following Eq. 2Eq. 2Eq. 2, this implies that the new piece of evidence can be found should the scenario entailed by the environment W_{suicide} be true.

- *What pieces or sets of additional evidence can differentiate between two scenario/hypotheses?* Let W_1 and W_2 be two environments each entailing a scenario, and let h_1 and h_2 be two hypotheses. Then any set of pieces of evidence O' that can be found if W_1 (or h_1) is true, but are inconsistent with W_2 (or h_2) can differentiate between the two scenarios (hypotheses). However, because evidence are the observable consequences of particular scenarios, hard inconsistencies between a scenario and a piece of evidence is very rare.

  An alternative method, and more relaxed, approach to establish evidence collection strategies consists of search for sets of pieces of evidence that logically follow from a given environment W_1 and which do not logically follow from W_2 or any of its consistent supersets.

  In the sample knowledge base devised for this paper, for instance, a medical examination of the body of the victim johndoe may uncover defensive injuries of johndoe. In the current (incomplete) version of the knowledge base, defensive injuries can only be explained by attempts by another person to overpower johndoe in order to murder him. An examination of the eyes of johndoe may reveal that johndoe has petechiae. However, petechiae is a consequence of asphyxiation and can be explained under scenarios describing suicidal, homicidal and accidental deaths. Therefore, it is possible to determine that a search for defensive injuries on the body of the victim is a more effective evidence collection strategy than an examination of the eyes of the victim.

But, although this approach for generating evidence collection strategies can be useful, it is crucially dependent on complete knowledge of all possible (reasonable) causes of plausible states and events. There are two important problems with this approach. Firstly, it provides no means of assessing the quality of the available evidence. Secondly, it provides no means of measuring the relative benefit of investigative actions that is part of an evidence collection strategy that meets the criterion.

In order to deal with these limitations, ongoing research is devising methods to extend the approach described in the paper with a means to measure the quality of the available evidence and the impact of additional investigative actions. The current focus is on the generation of Bayesian Networks to calculate the probability of plausible states and events in the scenario space and the use of information theory to compute entropy in the scenario space as an assessment of quality of evidence before or after evidence collection efforts [27].

6 Application

The theoretical ideas presented in the previous two sections have been developed into a prototype decision support software. This section briefly discusses how this prototype is employed.

6.1 User input

After the initial set up of the application, which involves choosing a knowledge base and starting a new session, the user/investigator must specify which facts and evidence are available in the given case. As it is not reasonable to assume that the user can specify these by means of formal predicates matching those in the knowledge base, the knowledge base contains one or more taxonomies for both facts and evidence.

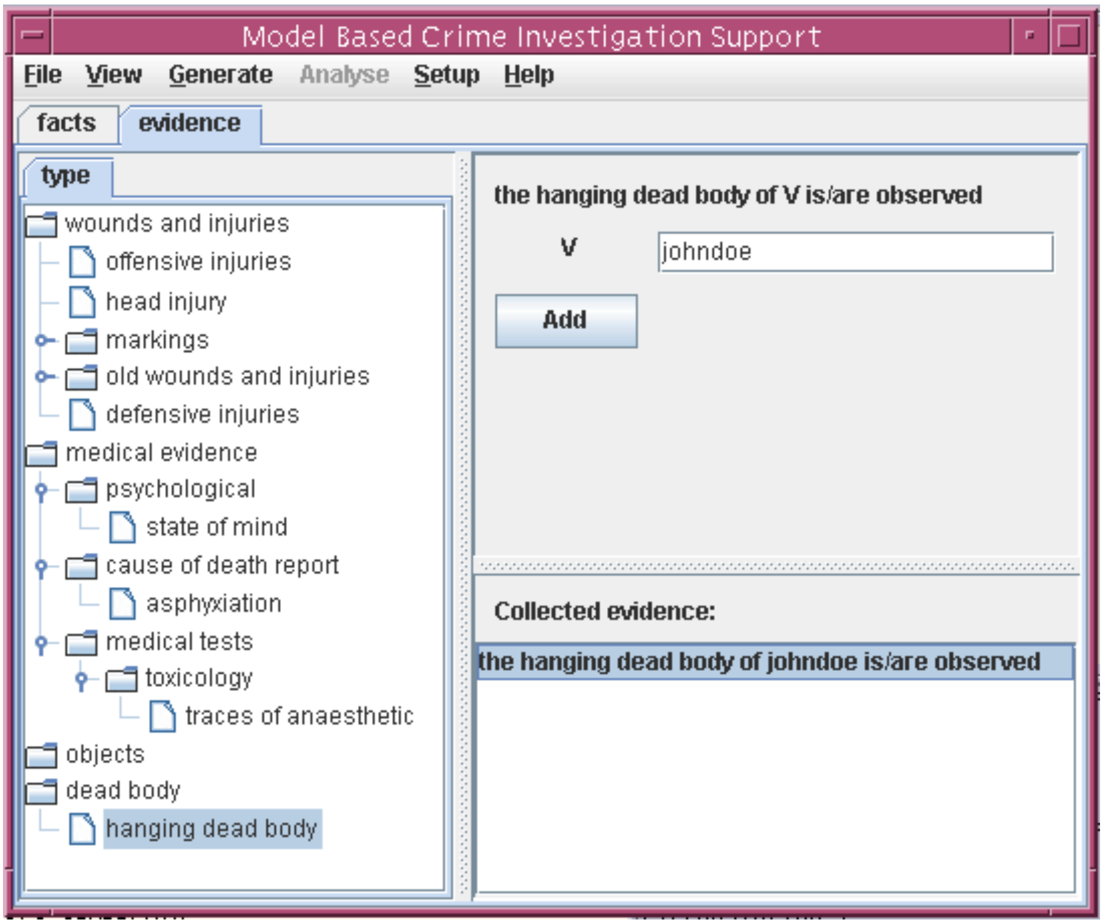


Figure 10: Interface for entering facts/evidence

For ease of reference, multiple taxonomies may organise the same set of facts or evidence according to different perspectives. Within each taxonomy, evidence is organised according to various distinctive attributes, such as the type of object that constitutes the evidence or the evaluation method that generated it. Once the user has found an item that corresponds to the appropriate piece of evidence, (s)he is required to enter some further details to help to uniquely identify the people and objects involved in the piece of evidence. Figure 10 shows a screenshot of the application as a user enters the details of a particular piece of evidence.

6.2 System output

Once all the available evidence and facts have been entered into the system, the user may choose to generate the scenario space. Once entered, three types of analysis become available.

First, the system can display the hypotheses consistent with the available evidence, and which plausible scenarios support them. Figure 11 shows a screenshot of the application where the hypotheses are displayed in a taxonomy. As indicated, the software has identified three hypotheses that are consistent with the evidence: suicidal death, homicidal death and accidental death. Clicking on a hypothesis causes the interface to display the minimal scenarios that support the selected hypothesis, and clicking on one of the displayed scenarios causes that scenario to be shown

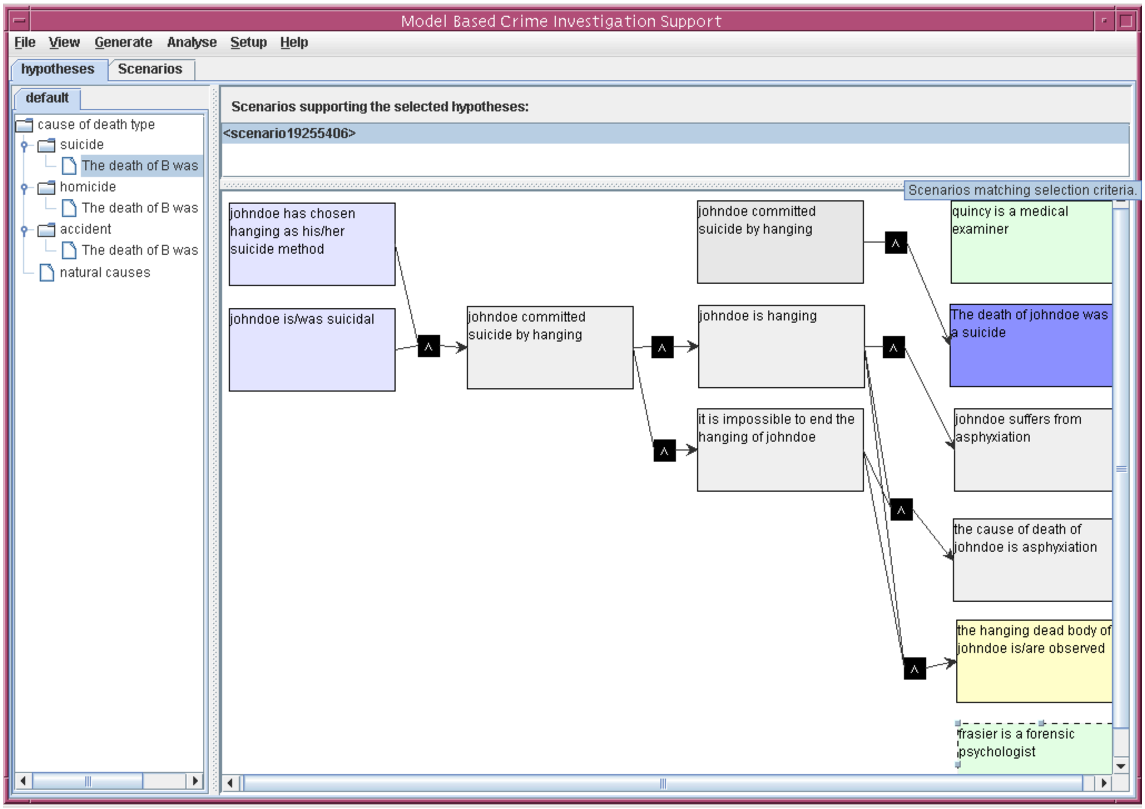


Figure 11: Navigating scenarios according to hypotheses

Currently, scenarios can be visualised in two different ways. The default approach summarises the scenarios by listing the assumptions they are based on and the hypotheses they support. This is a good representation to quickly identify the distinctive features of a scenarios, as it hides the underlying causal reasoning. Another view of a scenario represents a causal hypergraph, similar to the one shown in Figure 3. Causal hypergraphs are particularly suitable for describing causal reasoning, and therefore they are a useful tool to explain a scenario to the user.

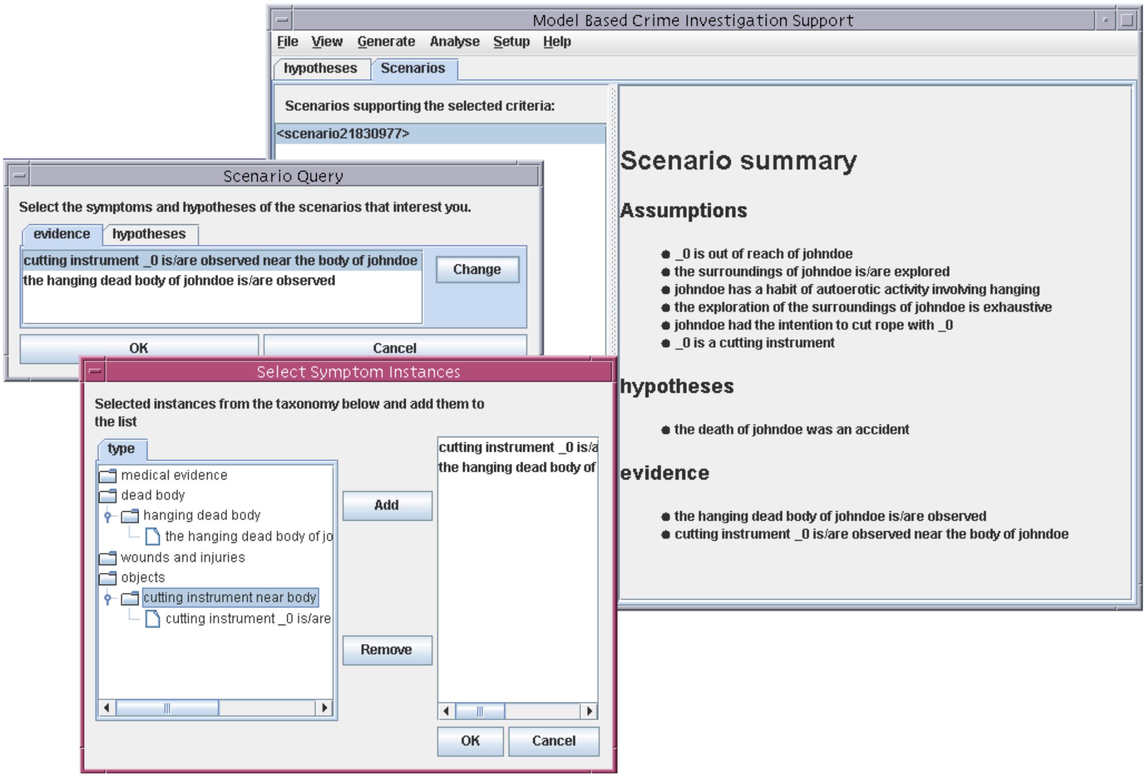


Figure 12: Querying the scenario space

Secondly, the user can query the system for scenarios that produce certain evidence and support certain hypotheses. This is a useful facility for what-if analysis. For example, the investigator might note that a “cutting instrument”, say a knife, has been recovered from the crime scene and wonders whether this rules out accidental death. As Figure 12 demonstrates, the system can answer this type of question by requesting it to search for a scenario that supports the available evidence, the discovery of a knife near the body and the accidental death hypothesis. In response, the system generates such a scenario by suggesting that the victim may have engaged in autoerotic activities and aimed to use the knife to cut the rope.

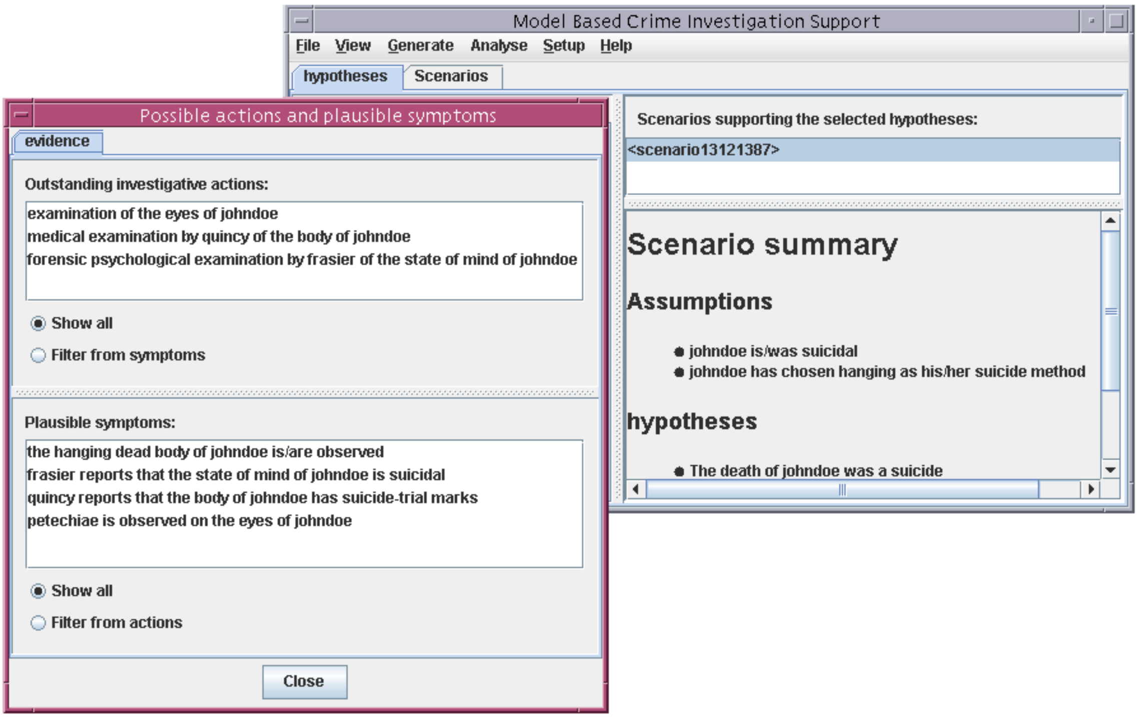


Figure 13: Additional plausible investigative actions

Finally, the decision support system can suggest additional pieces of evidence that may be collected if a given scenario were true. Whenever the user has selected a scenario generated by the system (using one of the aforementioned two facilities), (s)he may request additional evidence that could be discovered if the selected scenario were true. In response, it will display the dialog box shown in Figure 13, which shows the additional evidence and the investigative actions required to uncover it.

7 Related work

Section 2 has highlighted some of the difficulties of major crime investigation. Generally speaking, crime detection and investigation is a complex problem, involving the collection and maintenance of large amounts of data and expert knowledge. As a result, a significant body of research has focused on the development of decision support systems (DSSs) to aid law enforcement agencies in this task. Although a detailed literature review is beyond the scope of this paper, this section presents a brief overview of related work.

7.1 Evidence Evaluation

A number of different strands of research has focused on reasoning about evidence (see [42] for a detailed survey).

Argumentation research has devised methods [46, 51] and tools [7, 30, 48] to model legal reasoning. This has allowed recent studies into the representation and categorisation of forms of legal arguments about evidence [50] and its relation to logics for defeasible argumentation [36]. While the system presented here automatically generates relatively simple representations of scenarios that are not necessarily suitable for use in the courtroom, these causal models, which describe how hypothetical scenarios are related to evidence, may form a useful input in the construction of valid legal arguments.

Probabilistic expert systems employ Bayesian inference to compare two alternative hypotheses, typically a theory suggested the prosecution with one proposed by the defence, based on the available evidence [9]. As such systems rely on prespecified Bayesian Networks [34], they are used to evaluate individual pieces of evidence in terms of how well they support one proposition over another. Typical applications of such systems include the analysis of possible cross-transfer of DNA material [2], and the profiling of mixtures of DNA material [33]. Recently developed methods for compositional modelling of Bayesian Networks [28] will enable the integration of such probabilistic knowledge into this work.

Other important research concerns the validity of evidence rather than its implications. ADVOCATE, for instance, is a expert system designed to evaluate the credibility of eye-witness statements based on the conditions in which their observations were made [4].

7.2 Decision Support Systems in Crime Investigation

In addition to systems that reason about evidence, a number other types of DSS for crime investigation have been developed.

One group of DSSs formalise expert knowledge in the form of a conventional expert system [37]. For example, AREST [3] is an expert system designed for profiling suspects of armed robberies. InvestigAide B&E [47] is an expert systems designed to support the processing and investigation of breaking and entering cases. It supports activities such as gathering and recording case data and provides useful information such as suspect characteristics and similar cases.

Another group of DSSs apply knowledge discovery and data mining techniques to databases containing past cases, police reports and intelligence data. The approaches employed range from data visualisation [20] to the use of more formal statistical analysis [15]. Good examples of mature applications in this area include the COPLINK suite of tools [6, 21] and RECAP [5]. COPLINK is a tool aimed at providing an information extraction facility that integrates data from multiple police forces. RECAP (REgional Crime Analysis Program) is a tool that seeks out patterns of similar modus operandi in an effort to identify organised crimes.

A final group of systems employ Case Based Reasoning (CBR) methods to help investigators discover similar past cases and solution methods that correspond to those past cases. In the context of crime investigation, CBR systems usually perform analysis tasks by means of predefined sets of information. They are particularly suitable for analysing volume crime and repeat victimisation [38, 39]. Typical applications include the categorisation of the risk of electronic commerce transactions [24], the categorisation of crimes and retrieval of cases with similar profiles in burglary [40], the differentiation between hostile intrusions of computer systems and other anomalous transactions [16] and identification of crimes with similar modus operandi and potential repeat offenders [45].

8 Conclusion and future work

This paper has introduced a novel type of decision support system for crime investigation, one capable of generating hypothetical crime scenarios from evidence and supporting the creation of evidence collection strategies. It tackles the problem by means of a knowledge base that captures the first principles underlying human understanding of how events can generate certain pieces of evidence. An algorithm has been presented that employs this knowledge base to abductively create plausible causes of individual pieces of evidence and compose them into a network that represents a space of plausible scenarios. The paper has also described a number of techniques to analyse a generated scenario space to support the formulation of evidence collection strategies.

The work presented here has some limitations that must be addressed in future work. Most importantly, the DSS is not yet able to produce evidence collection strategies. Evidence collection strategies are effective if they are able to reduce the number and diversity of most likely scenarios. As such, this feature requires a means of measuring the likelihood of scenario, which is not present yet in the existing work.

A number of different approaches are possible. The conventional approach, based on Shannon's information theory [44], is the maximal entropy reduction technique, amongst others, applied to classifying faults in the model based diagnosis literature [12]. This approach requires that the likelihood is measured by probabilities and this can be accomplished by extending the existing approach with a compositional modeller for generating Bayesian Networks, such as the one presented in [27].

The use of Bayesian inference methods not only allows a straightforward extension of the work presented here, it is also very similar to the evidence evaluation methodology favoured by the forensics statistics community and some major forensic laboratories such as the Forensic Science Service [9]. However, it is also rather controversial [41]. Therefore, symbolic reasoning methods, such as symbolic preference handling methods [13], can be employed as alternative method of measuring the likelihood of scenarios. However, new methodologies for evaluating the effectiveness of evidence collection strategies must be developed if the likelihood of scenarios is assessed by a symbolic approach.

Another important extension involves the representation of events and states in time and space. Although the discussion in this paper has avoided these issues, the presumed spatial and temporal location in which hypothetical states and events occur can have a significant effect on the likelihood of scenarios. In order to incorporate such considerations in the present work, the existing DSS will have to be extended with representational formalisms that can describe qualitatively distinct locations of objects and events in time and space, as well as corresponding inference mechanisms. Event calculus [29] and situational calculus [31] are examples of potential approaches to implement this feature.

Finally, it should be pointed out that while the example knowledge base employed to illustrate the ideas in the paper contains some realistic and original example, it is by no means complete. In future work, this example will be extended substantially and a tool will be developed to do edit, verify and validate this and other knowledge bases.

Acknowledgements

This work has been supported in part by EPSRC grant GR/S63267. Both authors are grateful to Colin Aitken, Allan Jamieson, John Kingston, David Lucy, Qiang Shen and John Zeleznikow for useful discussion, whilst taking the full responsibility of the views expressed here.

References

[1] S. Addanki, R. Cremonini, and J.S. Penberthy. Graph of models. *Artificial Intelligence*, 51:145–177, 1991.

[2] C. Aitken, F. Taroni, and P. Garbolino. A graphical model for the evaluation of cross-transfer evidence in DNA profiles. *Theoretical Population Biology*, 63(3):179–190, 2003.

[3] A.B. Badiru, J.M. Karasz, and R.T. Holloway. Arest: Armed robbery eidetic suspect typing expert system. *Journal of Police Science and Administration*, 16(3):210–216, 1988.

[4] M. Bromby, M. MacMillan, and P. McKellar. A common-kads representation for a knowledge based system to evaluate eyewitness identification. *International Review of Law Computers and Technology*, 17(1):99–108, 2003.

[5] D.E. Brown. The regional crime analysis program (RECAP): A framework for mining data to catch criminals. In *Proceedings for the 1998 International Conference on Systems, MAN, and Cybernetics*, pages 2848–2853, 1998.

[6] H. Chen, J Schroeder, R.V. Hauck, L. Ridgeway, H. Atabakhsh, H. Gupta, C. Boarman, K. Rasmussen, and A.W. Clements. COPLINK Connect: Information and knowledge management for law enforcement. *Decision Support Systems: Special Issue on Digital Government*, 34(3):271–285, 2002.

[7] J. Conklin and M. Begeman. gIBIS: a hypertext tool for exploratory policy discussion. *ACM Transactions on Office Information Systems*, 4(6):303–331, 1988.

[8] L. Console and P. Torasso. A spectrum of logical definitions of model-based diagnosis. *Computational Intelligence*, 7(3):133–141, 1991.

[9] R. Cook, I. Evett, G. Jackson, P. Jones, and J. Lambert. A model for case assessment and interpretation. *Science and Justice*, 38:151–156, 1998.

[10] J. de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28:127–162, 1986.

[11] J. de Kleer. A general labeling algorithm for assumption-based truth maintenance. In *Proceedings of the 7th National Conference on Artificial Intelligence*, pages 188–192, 1988.

[12] J. de Kleer and B.C. Williams. Diagnosing multiple faults. *Artificial Intelligence*, 32:97–130, 1987.

[13] J. Delgrande, T. Schaub, H. Tompits, and K. Wang. A classification and survey of preference handling approaches in nonmonotonic reasoning. *Computational Intelligence*, 20(2):308–334, 2004.

[14] D. Dixon. Police investigative procedures. In C. Walker and K. Starmer, editors, *Miscarriage of Justice. A Review of Justice in Error*, 1999.

[15] D. Dzemydiene and V. Rudzkiene. Multiple regression analysis in crime pattern warehouse for decision support. In *Proceedings of the 13th International Conference on Database and Expert Systems Applications*, pages 249–258, 2000.

[16] M. Esmaili, R. Safavi-Naini, B. Balachandran, and J. Pieprzyk. Case-based reasoning for intrusion detection. In *IEEE Twelfth Annual Computer Security Applications Conference*, pages 214–223, 1996.

[17] B. Falkenhainer and K. Forbus. Compositional modeling: finding the right model for the job. *Artificial Intelligence*, 51:95–143, 1991.

[18] L. Festinger. *A Theory of Cognitive Dissonance*. Row Peterson, 1957.

[19] S. Greer. Miscarriages of criminal justice reconsidered. *Modern Law Review*, 58:71, 1994.

[20] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT Press, Boston, Massachusetts, 2001.

[21] R.V. Hauck, H. Atabakhsh, P. Ongvasith, H. Gupta, and H. Chen. Using coplink to analyze criminal justice data. *IEEE Computer*, 35(3):30–37, 2002.

[22] B. Irving and C. Dunningham. Human factors in the quality control of CID investigations and a brief review of relevant police training. *Royal Commission on Criminal Justice Research Studies*, 21, 1993.

[23] A. Jamieson. A rational approach to the principles and practice of crime scene investigation: I. principles. *Science & Justice*, 44(1):3–7, 2004.

[24] C. Jung, I. Han, and B. Suh. Risk analysis for electronic commerce using case-based reasoning. *International Journal of Intelligent Systems in Accounting, Finance & Management*, 8:61–73, 1999.

[25] J. Keppens. *Compositional Ecological Modelling via Dynamic Constraint Satisfaction with Order-of-Magnitude Preferences*. PhD thesis, The University of Edinburgh, 2002.

[26] J. Keppens and Q. Shen. On compositional modelling. *Knowledge Engineering Review*, 16(2):157–200, 2001.

[27] J. Keppens and Q. Shen. Causality enabled compositional modelling of bayesian networks. In *Proceedings of the 18th International Workshop on Qualitative Reasoning about Physical Systems*, pages 33–40, 2004.

[28] J. Keppens and Q. Shen. Compositional model repositories via dynamic constraint satisfaction with order-of-magnitude preferences. *Journal of Artificial Intelligence Research*, 21:499–550, 2004.

[29] R. Kowalski and M. Sergot. A logic-based calculus of events. *New Generation Computing*, 4:67–95, 1986.

[30] R. Loui, J. Norman, J. Alpeter, D. Pinkard, D. Craven, J. Linsday, and M. Foltz. Progress on room 5: a testbed for public interactive semi-formal legal argumentation. In *Proceedings of the 6th Internation Conference on Artificial Intelligence and Law*, pages 207–214, 1997.

[31] J McCarthy and P.J. Hayes. Some phylosophical problems from the standpoint of artificial intelligence. In Meltzer, B. and Michie, D., editors, *Machine Intelligence*, volume 4. Edinburgh University Press, 1969.

[32] M. McConville, A. Saunders, and R. Leng. *The Case for the Prosecution*. Routledge London, 1991.

[33] J. Mortera, A. Dawid, and S. Lauritzen. Probabilistic expert systems for dna mixture profiling. *Theoretical Population Biology*, 63(3):191–205, 2003.

[34] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan-Kaufmann, 1988.

[35] D. Poole. Explanation and prediction: An architecture for default and abductive reasoning. *Computational Intelligence*, 5(2):97–110, 1989.

[36] H. Prakken, C. Reed, and D. Walton. Argumentation schemes and generalisations in reasoning about evidence. In *Proceedings of the 9th Internation Conference on Artificial Intelligence and Law*, 2003.

[37] E.C. Ratledge and J.E. Jacoby. *Handbook on Artificial Intelligence and Expert Systems in Law Enforcement*. Greenwood Press, Westport, Connecticut, 1989.

[38] M. Redmond and A. Baveja. A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, 141:660–678, 2002.

[39] M. Redmond and C. Blackburn Line. Empirical analysis of case-based reasoning and other prediction methods in a social science domain: Repeat criminal victimization. In *Proceedings of the 5th International Conference on Case-Based Reasoning*, pages 452–464, 2003.

[40] O. Ribaux and P. Margot. Inference structures for crime analysis and intelligence: the example of burglary using forensic science data. *Forensic Science International*, 100:193–210, 1999.

[41] B. Robertson and T. Vignaux. *Interpreting Evidence: Evaluating Forensic Science in the Courtroom*. John Wiley and Son Ltd., 1995.

[42] D. Schum. *The Evidential Foundations of Probabilistic Reasoning*. Northwestern University Press, 1994.

[43] S. Sedly. Whose justice? In *London Review of Books*, page 6, 23 June 1993.

[44] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.

[45] J. Toland and B. Rees. Applying case-based reasoning to law enforcement. *International Association of Law Enforcement Intelligence Analysts Journal*, 15, 2005.

[46] S. Toulmin. *The Uses of Argument*. Cambridge University Press, 1958.

[47] L. Valcour. Investigate B& E: Break & enter expert system. Technical Report TR-11-97, Canadian Police Research Centre, 1997.

[48] B. Verheij. Automated argumentation assistance for lawyers. In *Proceedings of the 7th Internation Conference on Artificial Intelligence and Law*, pages 43–52, 1999.

[49] C. Walker and K. Starmer, editors. *Miscarriage of Justice. A Review of Justice in Error*. Blackstone Press, 1999.

[50] D. Walton. Argumentation and theory of evidence. *New Trends in Criminal Investigation and Evidence*, 2:711–732, 2000.

[51] J. Wigmore. *The Principles of Judicial Proof*. Little, Brown, and company, 2nd

edition, 1931.