Exploratory Interaction with a Bayesian Argumentation System

Ingrid Zukerman, Richard McConachy, Kevin Korb and **Deborah Pickett**
School of Computer Science and Software Engineering
Monash University
Clayton, Victoria 3168, AUSTRALIA
{ingrid,ricky,korb,debbiep}@csse.monash.edu.au

Abstract

We describe an interactive system which supports the exploration of arguments generated from Bayesian networks. In particular, we consider key features which support interactive behaviour: (1) an attentional mechanism which updates the activation of concepts as the interaction progresses; (2) a set of exploratory responses; and (3) a set of probabilistic patterns and an Argument Grammar which support the generation of natural language arguments from Bayesian networks. A preliminary evaluation assesses the effect of our exploratory responses on users' beliefs.

1 Introduction

An ideal interactive argumentation system would allow a user to respond to an argument with a full counterargument, and it would allow the argumentation process to go on indefinitely, producing a series of arguments and counterarguments. In this paper, we describe an interactive version of our argumentation system NAG (*Nice Argument Generator*), which enables a user to explore the impact of different beliefs and propositions on arguments generated from Bayesian networks (BNs). Such an exploratory capability constitutes a significant step towards allowing a user to interact freely with an argumentation system.

NAG interacts with a user through a web interface. The interaction starts when NAG presents to the user background information for a particular scenario and a hyper-text rendering of the argument generated by NAG in support of a goal proposition. NAG attempts to generate arguments that are *nice*, meaning that they are both normatively correct and persuasive for the target audience. To this effect it consults two models during argument generation: a normative model, which contains our best understanding of the domain of discourse, and a revisable model of the user's beliefs. These models are not necessarily synchronized, since the user and NAG may have different beliefs and may apply different inference patterns. Thus, operationally, a nice argument is one that achieves a desired degree of belief in the goal in each of these models.

After presenting an argument, NAG allows the user to perform the following operations: (1) select a new goal proposition; (2) ask NAG to argue for or against a proposition in the current argument; (3) ask NAG to include or exclude a proposition; (4) ask NAG a *what about* question, i.e., to consider the effect of a proposition on the argument; and (5) ask NAG a *what if* question, i.e., to consider a hypothetical change in the belief in a proposition. Each response results in the generation of a (revised) argument which incorporates the user's request. The user can either retain this argument or revert to the previous situation. These operations enable the user to examine different aspects of an argument in piece-meal fashion.

The argument generation process is described in [Zukerman et al., 1998; McConachy et al., 1998]. The focus of this paper is on the interactive capabilities of the system. In particular, we consider key features which support these capabilities: (1) an attentional mechanism which updates the activation of concepts as the interaction with the user progresses (Section 4); (2) a set of operations for exploring the argument (Section 5); and (3) a set of probabilistic patterns and an accompanying *Argument Grammar*, which extend patterns such as Toulmin's (1958) to enable the generation of natural language arguments from BNs (Section 6). These features provide a foundation for enabling users to argue with NAG.

In the next section, we show a sample interaction with NAG. In Section 3, we describe the argument generation process. We then consider NAG's knowledge representation and attentional mechanisms, followed by its exploratory operations and argumentation patterns. Finally, we discuss the results of a preliminary evaluation, review related research and present concluding remarks.

2 Sample Interaction

In this section, we describe an actual interaction with a user. Our domain is a fictitious murder scenario, which was chosen to minimize the influence of a user's pre-existing beliefs. The interaction started when the user was given the following preamble (which contains all the available observables).

Scratchy, a notorious spy from Vulcan, was found murdered in his bedroom, which is in the second storey of his house.

Indentations were found on the ground right outside Scratchy's window, and they were observed to be circular. In addition, one set of footprints was found outside Scratchy's window, but the bushes were undisturbed.

Itchy and Scratchy were long-term enemies, and Itchy's fingerprints were found on the murder weapon (a gun discovered at the scene and registered to Itchy).

Itchy, who is the head of the INS, has a ladder with oblong supports, which he was planning to use to paint his house.

Poochie, the town's mayor, has a cylindrically-supported ladder which is available only to him.

The user was then asked for his degree of belief in the goal proposition (*Itchy did not kill Scratchy*) and in Itchy's means, opportunity and motive to kill Scratchy, and also for his confidence in these judgments. The user indicated that it is *rather likely* that Itchy had a motive, *quite likely* that he had the means, and *less likely than not* that he had the opportunity to kill Scratchy or that he killed Scratchy.[1] NAG then generated an initial argument in support of the goal, taking into account the user's beliefs. Notice however, that the presented degrees of belief are NAG's.

Very probable hatred between Itchy and Scratchy imply a very probable motive to kill Scratchy.

Itchy's firing of the gun and Itchy's gun being used to kill Scratchy imply a means to murder Scratchy.

Circular indentations outside Scratchy's window, cylindrical supports to Poochie's ladder and availability of Poochie's ladder only to Poochie imply Poochie almost certainly was outside Scratchy's window, which together with Itchy's ladder almost certainly not being outside Scratchy's window and a single person being outside Scratchy's window implies Itchy very probably not being outside Scratchy's window.

The circular indentations outside Scratchy's window imply Scratchy very probably being shot from outside the window. This together with Itchy very probably not being outside Scratchy's window imply a very probable lack of opportunity to murder Scratchy.

Even though Itchy very probably had a motive to kill Scratchy and Itchy had the means to murder Scratchy, the very probable lack of opportunity to murder Scratchy implies Itchy's very probable innocence.

After this argument, the user retained his belief in Itchy's motive to kill Scratchy, but reduced his belief in Itchy's means to kill Scratchy to *rather likely*. He also reduced his belief in Itchy's opportunity to murder Scratchy and Itchy's guilt to *quite unlikely*, which are closer to NAG's normative beliefs than the user's initial beliefs after the preamble.

The exploratory interaction started with a request to exclude the proposition *Itchy fired the gun*, which resulted in a stronger argument for Itchy's innocence, where the paragraph regarding Itchy's means to murder Scratchy was omitted, and the concluding paragraph was replaced as follows.

Even though Itchy very probably had a motive to kill Scratchy and Itchy could have had the means to murder Scratchy, the very probable lack of opportunity to murder Scratchy implies Itchy's almost certain innocence.

The user returned to the original argument (without retaining the exclusion), and asked NAG to support the proposition *Only one person was outside Scratchy's window*. NAG generated the following sub-argument.

A single set of footprints outside Scratchy's window implies a single person being outside Scratchy's window.

The user then requested that this sub-argument be included in the main argument, and asked a *What if* question which explored the effect of a belief of *even chance* in the proposition *Itchy was outside Scratchy's window*. This resulted in an argument which yielded a weak belief in the goal.

Itchy and Scratchy very probably hated each other; therefore, Itchy very probably had a motive to kill Scratchy.

Itchy fired the gun and Itchy's gun was used to kill Scratchy; hence, Itchy had the means to murder Scratchy.

Circular indentations were found outside Scratchy's window; hence, Scratchy very probably was shot from outside the window.

If Itchy maybe was outside Scratchy's window and Scratchy very probably was shot from outside the window, Itchy possibly had no opportunity to murder Scratchy.

Despite a very probable motive to kill Scratchy and the means to murder Scratchy, the possible lack of opportunity to murder Scratchy implies Itchy's possible innocence.

After the exploratory interaction, the user was asked post-test questions regarding his belief in the goal proposition and in Itchy's means, opportunity and motive for killing Scratchy. The user did not change his beliefs as a result of the interaction (but three of these beliefs were already close to NAG's).

3 Argumentation Process

Figure 1 shows the modules of NAG. The Strategist, which receives as input a goal proposition, drives the argumentation process and produces an argument in support of this goal.[2] This process is influenced by the context, which changes at each turn in the interaction due to a user's request and the argument NAG generates to address this request.

The argument generation process is composed of a sequence of generation-analysis cycles [Zukerman et al., 1998]. The Strategist first invokes the Attentional Mechanism (Section 4) to activate propositions which are related to the items that are currently salient (including the goal). These propositions form an initial *Argument Graph* – a network with nodes that represent propositions and links that represent the inferences that connect these propositions. This Argument Graph is passed to the Generator, which consults the user model and the normative model to add information to the graph. The inferences and propositions that are compatible in both models and are related to the propositions in the current Argument Graph are incorporated into the graph. The resultant Argument Graph is then passed by the Strategist to the Analyzer in order to evaluate its niceness.

To assess the persuasive power of an argument represented by an Argument Graph, the Analyzer determines its effect on the belief in the goal proposition in the user model (Section 4). The normative correctness of the argument is assessed by determining its effect on the belief in the goal in the normative model. If the Analyzer reports that the Argument Graph is not nice enough, e.g., the belief in the goal is not high enough in one or both of the models, the Argument Graph is returned to the Strategist, and another generation-analysis cycle is performed: the Strategist re-activates the Attentional Mechanism to expand the reasoning context, followed by a re-activation of the Generator and then the Analyzer. This process iterates until a nice Argument Graph is built, or NAG is unable to continue, e.g., because it failed to find further evidence. In this case, NAG performs the following actions in turn. First, it tries to enhance the argument by including propositions that are believed only in the normative model. If the resulting argument achieves the intended belief

[1] The descriptive names for the probabilities are based on those described in [Elsaesser, 1987].

[2] The interface allows the user to select a goal proposition that NAG should argue for. However, for our evaluation the initial goal proposition is determined in advance, and the user is allowed to select subsequent goals.

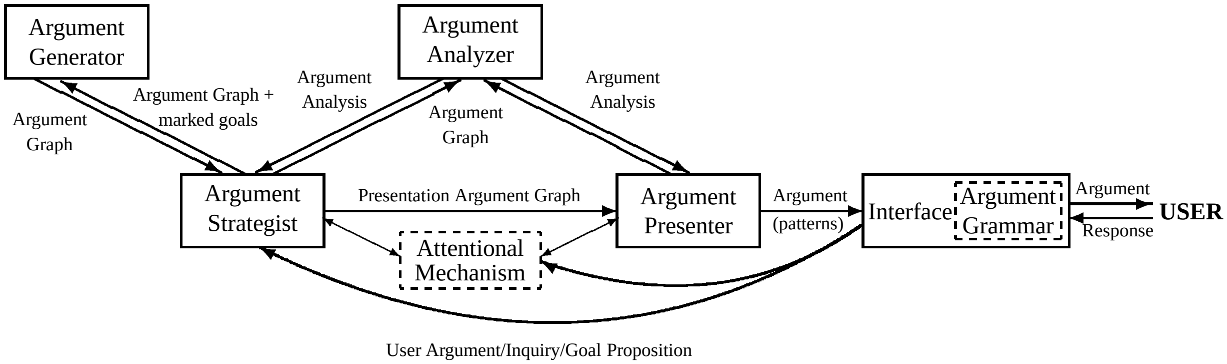

Figure 1: System architecture.

in the goal in the normative model, it is retained. However, the added propositions are not immediately incorporated into the user model; they will be incorporated if the user accepts them implicitly (by failing to question their inclusion in the argument). If this argument fails, NAG attempts to argue for the negation of the goal. To this effect, it first tries a nice argument, and then an argument which includes additional normative propositions. Finally, if this fails too, NAG selects the strongest argument it could generate, either for the goal or its negation. These actions were chosen, instead of simply giving up, based on suggestions made by users who tried a previous version of NAG. Note that failure can still happen if the user's requirements over-constrain NAG, e.g., the user demands the inclusion of a particular proposition in an argument for a goal that is not related to this proposition (Section 5).

Once an argument is generated, the Argument Graph is passed to the Presenter, which selects an argumentation strategy, e.g., premise-to-goal or goal-to-premise, and removes (easily inferred) propositions from the argument [McConachy et al., 1998]. After each removal, the Presenter activates the Analyzer to check whether the belief in the goal proposition has not been significantly altered, and the Attentional Mechanism to determine whether the argument can still be followed by the user. After the Presenter determines that no more propositions can be removed from the argument, it extracts Bayesian reasoning patterns from the Argument Graph, and passes them to the interface, which uses the Argument Grammar to generate a textual version of the argument, and presents it in hyper-text form (Section 6).

4 Knowledge Representation and Attentional Mechanism

Both the user model and the normative model rely on a BN as their primary representational tool, each model using a different BN. We have chosen BNs because of their ability to represent normatively correct reasoning under uncertainty. During argument generation, the Generator may draw upon additional knowledge bases, such as rule-based systems, to add information to the BNs as needed. During argument analysis, the Analyzer performs BN propagation on the portions of the normative and user models which correspond to the Argument Graph and are connected to the goal.[3]

In order to simplify the argument generation process and to model more accurately a user's cognitive abilities, we apply the Attentional Mechanism, which focuses upon the portion of the BN in each model that may be relevant to the current argument.[4] Hence, each of the user model and the normative model at any given time contains a single Bayesian subnetwork that is in focus. The structural intersection of these subnetworks forms the Argument Graph. For example, consider a situation where NAG's normative model believes that Poochie's ladder was outside Scratchy's window (P1) and that it is available only to Poochie (P2). In addition, the normative model has an inference pattern whereby these propositions have implications about Poochie being outside Scratchy's window (P3). Now, let us assume that the user model shares the normative model's belief in P1, is uncertain about P2, and also believes that Poochie was seen at the local toy store on the night of the murder (P4). Further, the user model has an inference pattern whereby P1, P2 and P4 affect the belief in P3. In this example, P1 and P2 and their link to P3 are included in the Argument Graph, but P4 is excluded, since it does not appear in the normative model. When analyzing the Argument Graph, propagation is done twice, once for the Bayesian subnetwork in the normative model and once for that in the user model (in our example, the probability calculations for the user model are marginalized over P4 to match the inference pattern in the normative model). Thus, we assess the niceness of the same argument with respect to both the user model and the normative model, thereby determining its persuasiveness and normative correctness respectively. In this example, NAG's argument for P3 will achieve a strong belief in this proposition in the normative model, but only a middling belief in the user model.

A plausible model of attentional focus requires NAG to follow associative links that do not necessarily represent evidential or causal relationships. We incorporate such links into our user and normative models by building one hierarchical

[3]It came late to our attention that the behaviour of our propagation algorithm departs from that of, for example, the Bayesian propagation algorithm in [Pearl, 1988], in particular in its treatment of loops. This departure does not seem to affect NAG's argumentative behaviour in the examples run to date. Nonetheless, we shall implement the standard propagation procedure in the near future.

[4]As shown in [Zukerman et al., 1998], the Attentional Mechanism can greatly reduce argument generation times. For instance, generation times were halved for our sample BNs, which contained between 20 and 120 nodes; the BNs for the example discussed in this paper contain 23 nodes.

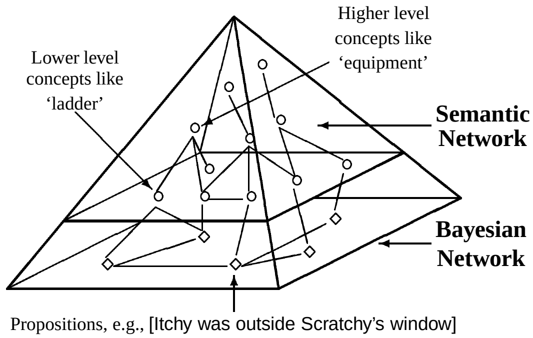

Figure 2: Semantic network on top of a Bayesian network.

semantic network (SN) on top of the user-model BN, and one on top of the normative-model BN (Figure 2). When performing attentional modeling, links in both the SN and the BN are followed. However, the resulting Argument Graph contains only propositions and links from the BNs.

To begin attentional simulation, NAG obtains an initial set of salient concepts and propositions from the context of the argument (in our example, the initial context is the preamble describing the murder scenario). As the argument or interaction progresses, new propositions become salient. For example, if the user asks NAG to include a proposition in an argument, this proposition and the concepts in it will become salient. We use activation with decay [Anderson, 1983], spreading from the (clamped) salient objects to determine the focus of attention. This process passes activation through the nodes in the SN and BN in the user model and in the normative model, each node being activated to the degree implied by the activation levels of its neighbours, the strength of association with those neighbours, and its immediately prior activation level (reduced by a time-decay factor). This iterative process ceases when an activation cycle fails to activate a new node. Any nodes in the SN or BN which achieve a threshold activation level at the end of this process are brought into the span of attention.

The context is dynamic, changing with each new request and resulting argument. In addition, the context for the user model may differ from that for the normative model, since NAG's experiences and the user's differ. NAG reasons about the argument and presents it, while the user reads it and reasons about it. After each request, the context for the normative model is composed of the proposition in this request and the propositions and concepts active upon completion of the reasoning process for the last argument. In addition to the proposition in the user's request, the context for the user model contains the propositions and concepts which were activated while the system simulated argument presentation, which NAG does in order to select a presentation order for its argument [McConachy et al., 1998].

5 Interacting with NAG

The interaction with the user influences NAG's argument generation process by changing the attentional focus and the beliefs in NAG's normative and user models. Since NAG has limited reasoning capacity (Bayesian propagation is restricted to the portion of the Argument Graph that is connected to the goal and the KBs are not exhaustively searched), the beliefs in NAG's user model and normative model may be affected by the additional Bayesian updating performed when addressing a user's request. Also, as indicated above, a request involving a particular proposition makes that proposition salient, which together with the argument generated to address this request modifies the context. As a result, when the user chooses to cancel his or her last request and revert to the previous argument, the argument generated by NAG may be slightly different from the previous argument.

NAG supports the following exploratory operations.

Selecting a new goal. This operation allows the user to request NAG to argue for any proposition known to it. Currently, the available propositions (and their negations), i.e., those in the normative and user models, are presented in a pull-down list. Clearly, this requires the interface to know in advance all the propositions accessible to the argumentation process, and so does not scale up. Nonetheless, this approach was adopted because it enables us to test exploratory interactions without introducing the complications attendant to accepting new goals presented in natural language.

As stated above, a newly selected goal is added to the reasoning context and passed to the Strategist, which activates the generation-analysis cycles described in Section 3.

Asking NAG to argue for or against a proposition in the argument. Here too the selected proposition is added to the reasoning context, and the generation-analysis process is activated to produce a sub-argument for or against this proposition. The presumption is that a new sub-argument for this proposition should be stronger than the current sub-argument. The user may then ask NAG to incorporate the resultant sub-argument into the main argument or may retain the previous argument (possibly in a modified form).

Including/excluding a proposition. As when selecting a new goal, a proposition to be included in the argument is selected from the pull-down list. The inclusion of a proposition cancels any previous exclusion of this proposition or its negation. The proposition is added to the reasoning context, and the generation-analysis process is activated in an attempt to produce a revised argument which includes it. The resultant argument may differ substantially from the last argument, in particular if the included proposition has a detrimental effect on the goal, thereby requiring the introduction of additional sub-arguments to maintain the desired level of belief in the goal. If, despite the inclusion of the selected proposition in the context, NAG generates a nice argument without this proposition, then NAG activates additional generation-analysis cycles (currently two) to attempt to link this proposition to the argument. If the proposition still cannot be connected, NAG reports this failure.

A proposition to be excluded is selected from the current argument. The exclusion of a proposition drops from consideration both the proposition itself and its negation, and cancels any previous inclusion of this proposition or its negation. NAG then tries to construct an argument without the excluded proposition; that is, the proposition is removed from the BNs (as are those of its ancestors that have no other connections to the argument). The children of the excluded proposition may then become premises, in which case uninformed priors for them are used. If the children have other parents, their conditional probabilities are marginalized over the excluded proposition, so that Bayesian propagation can proceed without it. Although any probabilistic impact of excluded propositions

Figure 3: Sample Bayesian network.

is avoided, exclusion has the opposite effect on attentional processing: asking someone to not think of unicorns has the opposite effect. Thus, excluded propositions are *added* to the reasoning context, which may induce NAG to incorporate related propositions in the argument.

As usual, the user may retain the resulting argument, or revert to the previous (possibly modified) argument.

Considering the effect of a proposition (*what about*). For this operation NAG returns only the reasoning path which connects the selected proposition to the goal (rather than returning the entire argument for the goal). This allows the user to inspect the effect of an isolated factor on the goal. To perform this operation, NAG adds the selected proposition to the reasoning context, and activates the generation-analysis process. The subgraph that connects the selected proposition to the goal is then returned for presentation. The user can choose to revert to the previous (possibly modified) argument, or to include the examined proposition in the argument.

Considering a hypothetical change in the belief in a proposition (*what if*). In this operation NAG changes the degree of belief in the selected proposition and converts the proposition into a premise. These changes, which take place in both the user model and the normative model, are *temporary*, since the user is explicitly requesting hypothetical reasoning. After producing a revised argument in light of the hypothetical belief, NAG reinstates the original beliefs in both the user model and the normative model and re-generates the previous argument. As usual, the change in context may result in the previous argument being modified.

Undoing changes. The user may undo the inclusion or exclusion of propositions and the incorporation of sub-arguments for or against a proposition. Each undo operation brings a proposition into focus. After the user completes his or her undoing actions, the generation-analysis process is reactivated, and NAG presents the resulting argument.

6 Argumentation Patterns and Grammar

The generation of natural language output from a BN requires the identification of probabilistic patterns which yield specific argumentation patterns. The patterns we have considered so far are: *explain-away* [Pearl, 1988], *neutralize*, *contradict*, *cause*, *evidence* and *imply* (which is a generic pattern that is used when the others are not applicable). When the interface receives an ordered list of patterns, it activates the corresponding productions in our Argument Grammar to generate natural language output. The formulas presented below identify probabilistic patterns with reference to the simple BN in Figure 3 (A and B may be groups of nodes).

Explain away. Reflects a situation where there are several potential explanations for a proposition, and finding out that one of these explanations is likely reduces the probability of the others. Its probabilistic requirements are:

$P(C|A) > P(C)$ and $P(C|B) > P(C)$, which means that A and B are potential explanations for C; and

$P(A|BC) < P(A|C)$ and $P(B|AC) < P(B|C)$, which means that given C, A explains away B and vice versa. Finally, we require $P(A|BC) <$ *threshold* and $P(B|AC) <$ *threshold*, where *threshold* is a probability indicative of an unlikely event; this ensures that the explaining-away effect has a useful impact on the argument.

<u>Explain-away</u>($\{A\}, \{B\}, C$,*strength?, direction?*):
Given(*type$_a$?*) Antecedent(C,*type$_a$?*),
Antecedent($\{A\}$,nominal) Imply(*strength?,direction?,type$_c$?*)
Consequent($\{B\}$,*type$_c$?*).
`Given that the grass is wet, the fact that the sprinklers were on last night implies that it probably did not rain.`

<u>Contradict-short</u>($\{A\}, \{B\}, \{C\}$,*strength?,direction?,cause?*):
Although(*type$_a$?*) Antecedent($\{A\}$,*type$_a$?*),
Antecedent($\{B\}$,nominal)
Imply(*strength?,direction?,cause?,type$_c$?*)
Consequent($\{C\}$,*type$_c$?*).
`Although Itchy's fingerprints were found on the gun, Itchy's alibi implies his innocence.`

Figure 4: Sample productions in the Argument Grammar.

Neutralize. Reflects a situation where some of the antecedents of an implication undermine the effect of others on the consequent, and the posterior belief in the consequent remains largely unchanged from its prior. That is, B neutralizes A if $P(C|AB) \cong P(C)$ and $P(C|A) < P(C)$.

Contradict. Similar to Neutralize, but one of the antecedents wins. That is, $P(C|AB) > P(C)$ and $P(C|A) < P(C)$.

Imply. Reflects a situation where the current beliefs in the antecedents increase the belief in the consequent. That is, $P(C|AB) > P(C)$.

Cause. Like Imply, but the relations have a "cause" tag.

Evidence. Like Imply, but the relations run in the opposite direction to the links in the BN.

Figure 4 illustrates two productions in our Argument Grammar with reference to Figure 3; the words in sans serif are invocations of production rules. Explain-away takes as input two lists of propositions ($\{A\}$ and $\{B\}$), and one singleton (C), which is the pivot on which the explanation hinges. Contradict-short takes two lists of antecedents ($\{A\}$ against the consequents and $\{B\}$ in favour), and one list of consequents, $\{C\}$. This production is used when $\{A\}$ contains only one or two propositions. The *direction?* and *cause?* parameters of the productions are obtained from tags in the BNs. For example, *cause?*=+ and *direction?*='forward' indicate a causal relation, while *direction?*='backward' indicates an evidential relation. The *strength?* parameter is obtained from the normative-model BN, since NAG states its own degrees of belief.

The Argument Grammar is feature-based. The productions in Figure 4 illustrate the *type* feature, which indicates whether a proposition should be realized in sentential or nominal form. For instance, the "despite" realization of Although requires a nominal continuation (e.g., "Despite Itchy's hatred for Scratchy"), while as shown in Figure 4, the "although" realization requires a sentential continuation. Our grammar assumes that each proposition has at least one realization (nominal or sentential); it can produce a nominal realization from a sentential one and vice versa by performing simple manipulations. To expedite the interaction, currently each node in the BNs has a hand-generated sentential or nominal realization. These realizations may be replaced by appropriate calls to grammars such as SURGE [Elhadad and Robin, 1996].

7 Evaluation

We performed a preliminary evaluation of the interactive features of NAG by testing the system with 16 current and former members of our department, using the murder scenario. The subjects reported their degrees of belief in four propositions and their confidence in their judgments immediately after reading the preamble, then after NAG's initial argument for Itchy's innocence, and after interacting with NAG using our exploratory operations. The propositions considered pertained to Itchy's guilt and his means, motive and opportunity to kill Scratchy. The most substantial change in belief after interacting with NAG concerned Itchy's guilt: 7 people reduced their belief in Itchy's guilt, while 3 people retained their belief that Itchy was probably innocent and 4 people remained uncertain. The mean degree of belief dropped from 41% after the initial argument to 36% after interacting with NAG. Even though these results are not statistically significant (due to the small sample size and high variance), they suggest that our exploratory interactions were helpful in persuading users to accept NAG's arguments. We hypothesize that the modest effect of these interactions was partly due to NAG's limited knowledge (several users indicated that additional factors should have been considered in NAG's arguments). Interestingly, the users' confidence dropped after the initial argument (possibly because NAG's argument contradicted some of the users' intuitions), but increased overall after the exploratory interactions. In most cases, the increased confidence accompanied user beliefs which agreed with or moved towards the beliefs in NAG's normative model.

8 Related Research

Our work extends traditional explanation capabilities for expert systems, e.g., [Buchanan and Shortliffe, 1985], in that it uses BNs as its main representation formalism and supports the exploration of arguments in ways that complement the justifications generated by earlier systems.

NAG is most similar in scope to the interactive argumentation system IACAS [Vreeswijk, 1994]. Like NAG, IACAS allows the user to add or remove information from its arguments. However, IACAS does not model attentional focus or tailor its arguments to the user.

Several researchers have incorporated context into dialogue systems, e.g., [Jönsson, 1995]. However, since these are often information providing systems, they use context mainly to further specify the user's requirements. In contrast, NAG uses the user's request and immediately preceding argument to determine the focus of attention, which in turn affects the argument generation process.

NAG's analysis-generation cycle resembles the abductive mechanism used in [Hobbs et al., 1993] to infer implicit information, and the propose-evaluate-modify cycle used in [Chu-Carroll and Carberry, 1995] to initiate information-sharing subdialogues. However, unlike these systems, NAG uses Bayesian reasoning and generates complete arguments.

9 Conclusion

We have described the interactive capabilities of our argumentation system, NAG, which support the exploration of different aspects of its arguments. The integration of these exploratory operations into our system required the adaptation of NAG's argumentation process to accommodate a user's requests, the incorporation of dynamic context changes into our attentional mechanism, and the identification of probabilistic patterns which support the selection of appropriate natural language patterns.

Our preliminary evaluation shows that the exploratory operations presented in this paper can improve the understanding of NAG's arguments. These operations provide a foundation for more elaborate forms of argumentation.

Acknowledgments

This research was supported in part by grant A49600323 from the Australian Research Council.

References

[Anderson, 1983] Anderson, J. R. (1983). *The Architecture of Cognition*. Harvard University Press, Cambridge, Massachusetts.

[Buchanan and Shortliffe, 1985] Buchanan, B. G. and Shortliffe, E. H. (1985). *Rule-based expert systems: The MYCIN experiments of the Stanford heuristic programming project*. Addison-Wesley Publishing Company.

[Chu-Carroll and Carberry, 1995] Chu-Carroll, J. and Carberry, S. (1995). Response generation in collaborative negotiation. In *Proceedings of the Thirty-Third Annual Meeting of the Association for Computational Linguistics*, pages 136–143.

[Elhadad and Robin, 1996] Elhadad, M. and Robin, J. (1996). An overview of SURGE: A reusable comprehensive syntactic realization component. Technical Report 96-03, Department of Mathematics and Computer Science, Ben Gurion University, Beer Sheva, Israel.

[Elsaesser, 1987] Elsaesser, C. (1987). Explanation of probabilistic inference for decision support systems. In *Proceedings of the AAAI-87 Workshop on Uncertainty in Artificial Intelligence*, pages 394–403, Seattle, Washington.

[Hobbs et al., 1993] Hobbs, J. R., Stickel, M. E., Appelt, D. E., and Martin, P. (1993). Interpretation as abduction. *Artificial Intelligence*, 63(1-2):69–142.

[Jönsson, 1995] Jönsson, A. (1995). Dialogue actions for natural language interfaces. In *IJCAI95 – Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1405–1411, Montreal, Canada.

[McConachy et al., 1998] McConachy, R., Korb, K. B., and Zukerman, I. (1998). Deciding what not to say: An attentional-probabilistic approach to argument presentation. In *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society*, pages 669–674, Madison, Wisconsin.

[Pearl, 1988] Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann Publishers, San Mateo, California.

[Toulmin, 1958] Toulmin, S. (1958). *Uses of Argument*. Cambridge University Press, Cambridge.

[Vreeswijk, 1994] Vreeswijk, G. (1994). IACAS: An interactive argumentation system. Technical Report CS 94-03, Department of Computer Science, University of Limburg.

[Zukerman et al., 1998] Zukerman, I., McConachy, R., and Korb, K. B. (1998). Bayesian reasoning in an abductive mechanism for argument generation and analysis. In *AAAI98 – Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 833–838, Madison, Wisconsin.