

Resolving Goal Conflicts via Argumentation-Based Analysis of Competing Hypotheses

Pradeep K. Murukannaiah*, Anup K. Kalia*, Pankaj R. Telang[†], and Munindar P. Singh*

* Department of Computer Science, North Carolina State University, Raleigh, NC 27695-8206, USA

pmuruka@ncsu.edu, akkalia@ncsu.edu, m.singh@ieee.org

[†] Cisco Systems Inc., Research Triangle Park, NC 27709, USA; ptelang@gmail.com

Abstract—A stakeholder’s beliefs influence his or her goals. However, a stakeholder’s beliefs may not be consistent with the goals of all stakeholders of a system being constructed. Such belief-goal inconsistencies could manifest themselves as conflicting goals of the system to be. We propose Arg-ACH, a novel approach for capturing inconsistencies between stakeholders’ goals and beliefs, and resolving goal conflicts. Arg-ACH employs a hybrid of (1) the analysis of competing hypotheses (ACH), a structured analytic technique, for systematically eliciting stakeholders’ goals and beliefs, and (2) rational argumentation for determining belief-goal inconsistencies to resolve conflicts. Arg-ACH treats conflicting goals as hypotheses that compete with each other and the winning hypothesis as a goal of the system to be. Arg-ACH systematically captures the trail of a requirements engineer’s thought process in resolving conflicts.

We evaluated Arg-ACH via a study in which 20 subjects applied Arg-ACH or ACH to resolve goal conflicts in a sociotechnical system concerning national security. We found that Arg-ACH is superior to ACH with respect to completeness and coverage of belief search; length of belief chaining; ease of use; explicitness of the assumptions made; and repeatability of conclusions across subjects. Not surprisingly, Arg-ACH required more time than ACH: although this is justified by improvements in quality, the gap could be reduced through better tooling.

Index Terms—Goal modeling; Goal conflicts; Argumentation; Argument schemes; Analysis of competing hypotheses; ACH

I. INTRODUCTION

The stakeholders of a system to be may have conflicting goals. Such conflicts, if unresolved, lead to an inconsistent specification of the system and the eventual failure of its implementation. Resolving goal conflicts has long been identified as important in requirements engineering. Van Lamsweerde et al. [1] describe techniques to systematically transform (create, delete, or modify) goals to resolve conflicts. Antón and Potts [2] analyze obstacles to goal satisfaction to address conflicts. Robinson [3] and Boehm et al. [4] describe negotiation-based approaches for conflict resolution.

We ask a fundamental question: *what makes stakeholders’ goals conflict?* Although stakeholders have ego-centric views [5], that, in itself, does not make their goals conflict. We must understand how stakeholders process (activate, promote, drop, or suspend) goals to identify the sources of their conflicts. Castelfranchi and Paglieri [6] argue that, in order to process a goal, one must have appropriate beliefs—that is, assumptions about the world. A belief may support, oppose, or be neutral about a goal.

We posit that *goals conflict when stakeholders have contradictory beliefs about supporting or opposing those goals.* That is, goals g_1 and g_2 conflict if a belief b_1 supports g_1 and opposes g_2 , and another belief b_2 supports g_2 and opposes g_1 . For example, let two goals of a meeting scheduler to be developed be *to reduce user effort in scheduling* and *to preserve participants’ privacy*. The two goals conflict given stakeholder beliefs that *participants share their calendars with the scheduler*, which supports the former but opposes the latter goal, and *participants reveal their availability to the scheduler, weekly*, which opposes the former but supports the latter goal. More complex n-ary conflicts may occur, but the exact nature of the conflict is not important here.

We seek to resolve goal conflicts by analyzing stakeholders’ beliefs about various goals. We treat conflicting goals as competing hypotheses and a belief about a goal as a piece of evidence about the corresponding hypothesis. The hypothesis intuition applies here since the hypothesis in question corresponds to a goal the system-to-be should potentially support.

We resolve goal conflicts by employing the Analysis of Competing Hypotheses (ACH) [7], a well-known structured analytic technique. ACH describes systematic steps to create a *diagnosticity matrix*, capturing the relative extent to which stakeholders’ beliefs are inconsistent about the given conflicting goals. Then, we resolve the conflict by adopting the least inconsistent goal from the diagnosticity matrix. A distributed version of ACH [8] can also be used, e.g., when stakeholders negotiate to resolve conflicts.

ACH seeks to capture the trail of a requirements analyst’s thought process in resolving conflicts. However, it falls short in answering three key questions concerning conflict resolution. First, how can we elicit stakeholders’ beliefs about conflicting goals? Such beliefs can be incomplete and ambiguous. Second, how can we compose beliefs? It is possible that individual beliefs are neutral about a goal, but their composition supports or opposes the goal. Third, how can we infer if a (composed) belief supports or opposes a goal? To answer these questions, an analyst must critically examine the goals and beliefs (or lack thereof). ACH offers neither guidance to answer these questions nor a systematic approach for capturing the answers in the diagnosticity matrix. The effectiveness of ACH, therefore, highly depends upon the analyst’s prior knowledge.

To address the limitations of ACH in resolving goal conflicts, we propose Arg-ACH, an argumentation-based method

for the analysis of competing hypotheses. Arg-ACH captures the analyst's mental model of stakeholders' goals and beliefs as arguments. Arg-ACH employs argumentation schemes, i.e., reusable patterns of reasoning, to systematically elicit stakeholders' beliefs about goals, compose beliefs, and infer relationships between beliefs and goals. Argumentation schemes serve as critical thinking aids regardless of the analyst's level of expertise. Also, our method facilitates critical examination of an analyst's rationale, and supports systematic updates in case a belief changes subsequent to the current analysis.

A. Contributions

- (1) We describe how ACH, a leading intelligence analysis technique, can be applied to resolve goal conflicts. Unlike goal transformations and negotiation techniques, ACH offers a novel perspective by resolving goal conflicts based on stakeholders' beliefs about goals.
- (2) We propose Arg-ACH, an extension to ACH incorporating arguments. Arg-ACH assists a requirements analyst in critically examining stakeholders' beliefs and goals to resolve goal conflicts. We demonstrate the benefits of Arg-ACH over ACH in a case study and via a rigorous empirical evaluation involving 20 human subjects.

B. Organization

Section II describes ACH and applies it for resolving goal conflicts in a sample scenario. Section III describes Arg-ACH and demonstrates how it helps in systematically eliciting stakeholders' beliefs and constructing arguments from beliefs to support or oppose goals. Sections IV and V describe the design of our empirical study and the findings from it. Section VI presents related work and Section VII the conclusions.

II. ANALYSIS OF COMPETING HYPOTHESES (ACH)

Heuer [7] proposed ACH to address intelligence analysis scenarios in which an analyst must choose among alternative hypotheses. Here, the term *hypothesis* is used in a broad sense to mean a potential outcome, explanation, or conclusion. A major objective of ACH is to reduce *confirmation bias*: the tendency of people to seek evidence to support a hypothesis they tentatively hold and discount evidence opposing that hypothesis [9]. ACH requires an analyst to evaluate a piece of evidence against each hypothesis before moving on to the next piece of evidence. Also, the conclusion drawn via ACH depends on the amount of inconsistent (opposing) evidence the analyst finds for each hypothesis, not the amount of consistent (supporting) evidence.

Resolving goal conflicts for a system-to-be parallels choosing from among competing hypotheses: a conflicting goal is a competing hypothesis and a belief supporting or opposing a goal is a piece of evidence. Therefore, we can adapt the ACH steps, as described below, to resolve goal conflicts.

(1) Identify conflicting goals. Conflicting goals need to be mutually exclusive. Consider all reasonable possibilities without judging their likelihoods. Identifying conflicting goals can be a complex task in itself. In intelligence analysis, techniques

such as structured brainstorming are used to identify competing hypotheses. Here, we assume that we are given conflicting goals and that our task is to resolve the conflict.

(2) Identify significant beliefs. A belief is significant if it supports or opposes one of the conflicting goals. Additionally, consider all logical deductions of the beliefs that support or oppose a conflicting goal.

(3) Create a diagnosticity matrix. The matrix shows the goals as columns and beliefs (original or deduced) as rows. A matrix cell shows if the belief is consistent, strongly consistent, inconsistent, strongly inconsistent, or neutral with respect to the goal corresponding to the cell. Here, a consistent belief supports a goal and an inconsistent belief opposes a goal.

(4) Refine the diagnosticity matrix. Reconsider the previously identified goals in light of the beliefs elicited so far. Combine or split goals, or introduce new goals as necessary. Identify beliefs corresponding to new goals and remove those not relevant anymore. Update the inconsistency scores.

(5) Draw tentative conclusions. For each goal, find the number of beliefs that are inconsistent with the goal (double-count strongly inconsistent beliefs). This is the inconsistency score of the goal (ACH includes other scoring methods such as *weighted inconsistency*, which we omit here for brevity). Then, the goal with the least inconsistency score is most desirable by the stakeholders of the system to be. Verify if this choice of goal reflects your judgment. If not, then identify additional beliefs and reconsider your assumptions.

(6) Analyze sensitivity. Analyze the sensitivity of the tentative conclusion with respect to the critical beliefs that drive the conclusion. Identify if the assumptions underlying the conclusion are questionable. Further, identify if there are alternative explanations or interpretations for each belief. Here, you may decide that additional analysis is required. If so, go back to a previous step as necessary.

(7) Report conclusions. Discuss the relative desirability of each goal and describe any additional considerations and assumptions underlying the overall conclusions. Next, identify milestones for future observation that may indicate that events are taking a different course than expected.

A. An Example Scenario

We describe a hypothetical scenario in which stakeholders' goals conflict. We demonstrate ACH by applying it to resolve this conflict. Imagine a sociotechnical system responsible for defending a city against terrorist attacks. Further, let the Train Authority (TA) and the Hotel Authority (HA) be two stakeholders in the system who administer trains and hotels in the city, respectively. In the wake of a recently foiled terrorist attack, the stakeholders have the following goals.

- (1) TA: *Enhance security on trains (trains for short).*
- (2) HA: *Enhance security at hotels (hotels for short).*

Further, let us assume that these goals conflict since the city can only adopt one of these goals due to budget constraints.

Now, a requirements analyst at the city must resolve the conflict, promoting one of the two goals.

B. ACH Solution to the Example Scenario

We skip the first step of ACH since we are already given the conflicting goals. In the second step, suppose we elicit the following beliefs from stakeholders. That is, assume that the analyst finds the following beliefs to start with.

- (1) US Intelligence Agency suspects an attack on trains.
- (2) British Intelligence Agency suspects an attack on hotels.
- (3) Trains were recently installed with security systems from the XYZ company.

Given these goals and beliefs, the next step in ACH is to construct a diagnosticity matrix such as the one shown in Table I, where a belief can be inconsistent (I), consistent (C), or neutral (N) with respect to a goal.

TABLE I: ACH diagnosticity matrix

		Goals	
		Trains	Hotels
Beliefs	US Intel suspects an attack on trains	C	I
	British Intel suspects attack on hotels	I	C
	Trains were recently installed with security systems from the XYZ company	I	N
Inconsistency score		-2	-1

Next, the analyst computes the *inconsistency score* for each goal. An individual score of I contributes -1 to the inconsistency score, whereas individual scores of C and N contribute 0. Table I shows the inconsistency scores for each goal (column sum). Based on these scores, the analyst may promote the goal to enhance security at hotels. The analyst can potentially stop here, assuming the conclusion matches his or her intuition. If not, the analyst must refine the diagnosticity matrix until it reflects his or her intuition.

III. ARGUMENTATION-BASED ACH (ARG-ACH)

ACH describes the steps for analyzing conflicting goals. However, executing the steps of ACH is nontrivial. For example, Step 2, i.e., to identify significant beliefs, largely relies on an analyst's rational thinking and domain knowledge.

We introduce Arg-ACH, which incorporates argumentation in the ACH process to assist an analyst in executing the steps of ACH. Unlike formal logic, argumentation is the study of informal reasoning generally employed by humans. Argumentation can be effective in resolving goal conflicts since it is often humans (requirements analysts or stakeholders) who perform the resolution. Argumentation has been successfully applied in areas such as legal reasoning, commonsense reasoning, decision making, and negotiation [10].

A. Incorporating Arguments in ACH

Below, we describe two techniques that help an analyst incorporate arguments in ACH. Note, however, that an analyst applying Arg-ACH operates under the broader rubric of ACH. The techniques below help the analyst compose beliefs, relate beliefs and goals, and determine the extent to which a belief supports or opposes a goal.

(1) Construct arguments. An argument consists of a conclusion and a set of premises such that the conclusion can be inferred from the premises. Our objective in constructing arguments is to identify the relationships between stakeholders' beliefs and goals. To do so, we employ argument schemes.

An argumentation scheme [11] is an inferential structure, i.e., reusable pattern, of arguments. A scheme specifies a conclusion, a set of premises, and a set of critical questions. Its conclusion can be inferred from its premises. The critical questions evaluate the argument by probing into its weak points. For example, the argument from *expert opinion* is a scheme in which one accepts an expert's assertion provided the critical questions fail to uncover a weakness in the argument. This scheme's structure is shown below.

Major Premise: Source E is an expert in a subject domain S containing a proposition A .

Minor Premise: E asserts that proposition A is true (false).

Conclusion: A is true (false).

Critical Questions:

- CQ₁: How credible is E as an expert?
- CQ₂: Is E an expert in the field that A is in?
- CQ₃: What did E assert that implies A ?
- CQ₄: Is E personally reliable as a source?
- CQ₅: Is A consistent with what other experts assert?
- CQ₆: Is E 's assertion based on evidence?

Given a set of domain-specific and domain-independent schemes, an analyst can construct arguments as follows.

- Treat a goal as the scheme's conclusion. Find beliefs corresponding to the scheme's premises.
- In case a belief is not found for some of the scheme's premises, make reasonable assumptions based on prior knowledge but specify all assumptions explicitly.
- Find beliefs that can answer the critical questions.
- Accept the argument generated from the scheme if all of its premises are supported by the beliefs and assumptions.

(2) Assign belief scores. An analyst can, potentially, construct multiple arguments for each goal, but may not believe in each argument to the same extent. To characterize an analyst's *belief* in an argument, we employ Jøsang et al.'s [12] notion of belief measure that is based on a triple: *belief* (B), *disbelief* (D), and *uncertainty* (U), such that $B + D + U = 1$. We refer to the triple as a BDU score. Also, we distinguish an analyst's *belief* from a stakeholder's belief by italicizing the former.

The B, D, U scores represent the analyst's *belief*, *disbelief*, and *uncertainty* about the argument. For example, consider a medical study conducted by a reputed university, which concludes that obesity leads to recurrence of breast cancer. If Alice, a patient, is obese, then her physician may assign a BDU score of $\langle B: 0.8, D: 0.1, U: 0.1 \rangle$ to an argument concluding that *Alice will have breast cancer recurrence* based on the evidence. Consider another patient Carol, who is borderline obese, but has a large tumor size. Assuming there is no information on how tumor size affects recurrence, Carol's physician's may assign a BDU score of $\langle B: 0.3, D: 0.2, U: 0.5 \rangle$ to the argument concluding *breast cancer recurrence for Carol*.

A conclusion may have multiple arguments supporting and multiple arguments opposing it. We compute the overall BDU score of the conclusion as follows: the overall B score is the average over the B scores of the supporting and D scores of the opposing arguments; the overall D score is the average over the D scores of the supporting and B scores of the opposing arguments; and the overall U score is $1 - B - D$. For example, for the conclusion that Alice will have breast cancer recurrence, if there is a supporting argument with the BDU score $\langle B: 0.8, D: 0.1, U: 0.1 \rangle$ and an opposing argument with the BDU score $\langle B: 0.4, D: 0.5, U: 0.1 \rangle$, the overall BDU score for the conclusion is $\langle B: 0.65, D: 0.25, U: 0.1 \rangle$. Finally, the analyst can choose between alternative conclusions based on the conclusions' *belief* scores.

Assigning BDU scores in Arg-ACH and inconsistency values in ACH are both subjective. Whereas the ACH matrix provides little information on why an analyst assigned inconsistency values a certain way, Arg-ACH captures the analyst's rationale in the form of arguments. The answers to critical questions in the argument schemes prompt an analyst to consider factors such as credibility and reliability of belief sources, assumptions made, and ambiguity in the evidence in deciding BDU scores for a conclusion.

B. Arg-ACH Solution to the Example Scenario

Starting from the goals and beliefs in our example scenario, an analyst can construct the arguments shown in Figure 1. An argument with a "+" supports and an argument with a "-" opposes its conclusion. The analyst can fit given beliefs to one of the relevant argumentation schemes. Assume that the relevant schemes for our example scenario are argument from *expert opinion* and argument from *analogy* [11].

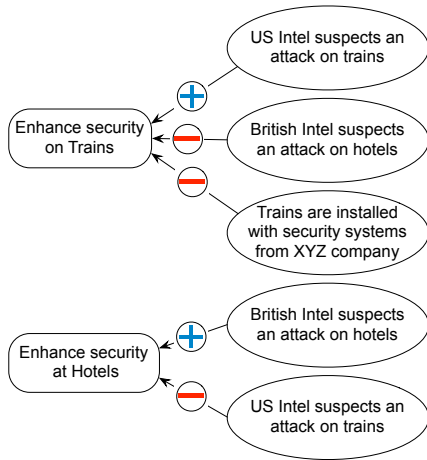


Fig. 1: Argument graph with initial beliefs.

Figure 2 shows how an analyst may fit beliefs into schemes. For brevity, we show only two beliefs fitted in schemes. The gray ovals show some (not all, for brevity) critical questions associated with the corresponding scheme. The critical questions prompt an analyst to consider scenarios he or she may not have imagined, otherwise. For example, in the first argument,

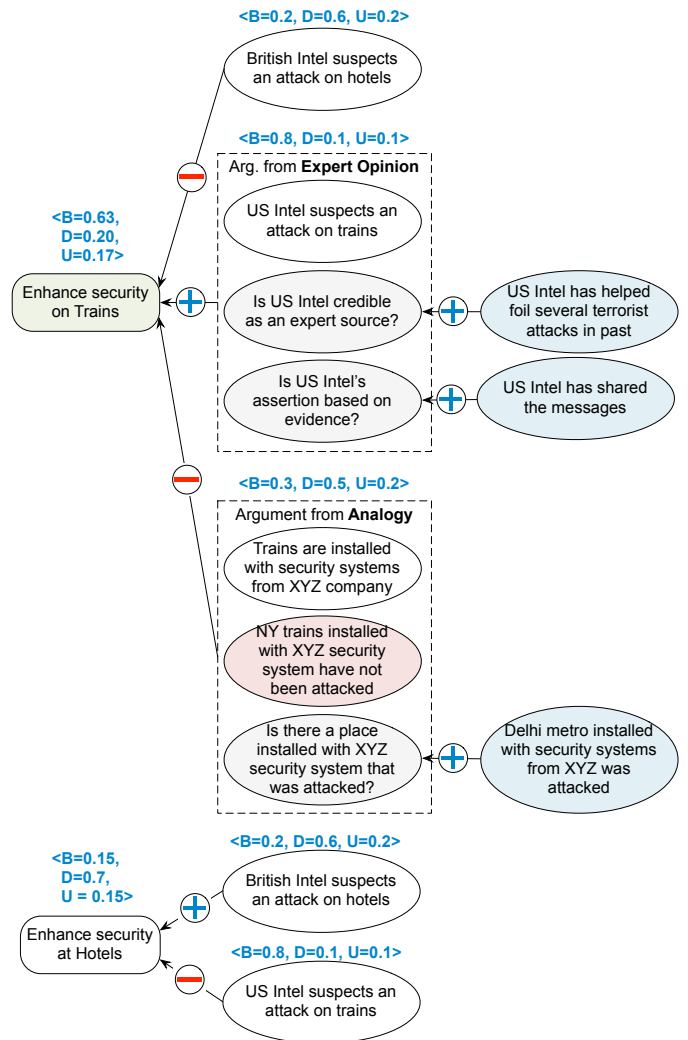


Fig. 2: Argument graph (partial) with beliefs elicited by answering critical questions from argumentation schemes.

the belief *US Intel suspects an attack on trains* supports the goal to *enhance the security on trains*. The expert opinion scheme's critical questions include:

- Is US Intel credible as an expert source?
- Is US Intel's assertion based on evidence?

In order to answer these questions, the analyst must look for more beliefs. In this case, we assume that the analyst finds additional beliefs shown in blue in Figure 2. A critical question may prompt the analyst to make an explicit assumption, e.g., the assumption shown in red ovals.

Once arguments are constructed, the analyst can assign BDU scores. For example, Figure 2 shows BDU scores the analyst may assign in our case study for each argument (shown on top). These scores are based on the analyst's judgment as to the extent to which various arguments are supported.

Finally, we compute the BDU score for each goal by composing BDU scores of different arguments for each goal. Figure 2 shows BDU scores for each goal (shown on top). It is

interesting to note that in the light of new beliefs, the analyst promotes the goal to *enhance security at trains*.

C. Revisiting the Meeting Scheduler Example

The above example about national security demonstrates Arg-ACH’s applicability in a broad sense to the requirements engineering of a sociotechnical system. However, the key ideas of Arg-ACH are equally applicable to traditional software systems. To demonstrate this, we apply Arg-ACH to resolve a conflict in the meeting scheduler example from Section I.

Consider that we are engineering the requirements of a meeting scheduler (called Scheduler). Given participant names and their schedules in some form, Scheduler recommends meeting slots to the participants. Next, let the following be two goals of the stakeholders of Scheduler:

- (1) *to reduce user effort in scheduling meetings*, and
- (2) *to preserve participants’ privacy* by not sharing a participant’s schedule with third parties.

Suppose these goals conflict because of two beliefs that:

- (1) *participants share their calendars with Scheduler*, and
- (2) *participants reveal their availability to Scheduler, weekly*.

Here, it appears that the former belief is consistent with reducing effort but inconsistent with preserving privacy, whereas the latter belief is consistent with preserving privacy but inconsistent with reducing user effort. Thus, given no additional beliefs an ACH solution would yield equal inconsistency values for both goals, thus failing to resolve the conflict.

Then, we argue that sharing calendar may lead to a negative consequence and thus the goal *to preserve privacy* must be promoted. Next, we argue from popular opinion that revealing constraints weekly may not be tiresome (unlike what we had assumed before). To do so, we elicit additional beliefs from stakeholders, e.g., evidence from a poll. Now, considering the original beliefs, our assumptions, and additional beliefs we elicited based on critical questions, we resolve the conflict by promoting the goal *to preserve privacy*.

IV. EMPIRICAL EVALUATION

The foregoing intuitions led us to hypothesize that Arg-ACH yields a better quality analysis of competing hypotheses than ACH. Now, we empirically evaluate this claim.

A. Study Design

We described the terrorist attack scenario we worked out in the previous section, but with an additional conflicting goal *to enhance security at airports* to 20 subjects and asked them to resolve conflicts. The subjects included five undergraduate and 15 graduate students. Our study was approved by our university’s Institutional Review Board (IRB). We collected an informed consent from each subject and provided a payment to each subject completing the study.

We employed the one-factor design with two alternatives—ACH and Arg-ACH. We created two groups of 10 subjects balanced based on the subjects’ education level and familiarity with intelligence analysis techniques as indicated in a pre-participation survey. We trained subjects on ACH or Arg-ACH to resolve conflicts in a breast cancer diagnosis scenario. We asked them to resolve conflicts in the terrorist attack scenario.

All subjects started from three conflicting goals—to enhance security on *trains*, at *hotels*, or at *airports*. We developed a database containing hypothetical beliefs pertaining to terrorist attacks in Mumbai. The subjects were asked to search the database using keywords and to use only what is in the database as beliefs in the analysis. The subjects were allowed to make logical deductions and assumptions where necessary. The beliefs in the database were incomplete and included conflicting information. Our objective was to capture the kind of beliefs an analyst often finds in real analysis scenarios. Figure 4 shows the top keywords searched by subjects.

In case of ACH, subjects employed the PARC ACH tool [13] to construct the diagnosticity matrix, analyze sensitivity, and derive conclusions. In case of Arg-ACH, subjects employed the Carneades Editor [14] to construct arguments and a spreadsheet with embedded macros to assign BDU scores.

The main deliverable for each subject was a report capturing the thought process of the subject in deriving the conclusions. The report for ACH subjects was in the form of a diagnosticity matrix. The report for Arg-ACH subjects was argument graphs and a spreadsheet of BDU scores. Further, subjects answered a post-participation survey to record their conclusions (i.e., the goal they promoted), time they spent on their analysis in minutes, and their perceived difficulty on a scale of 1 to 5.

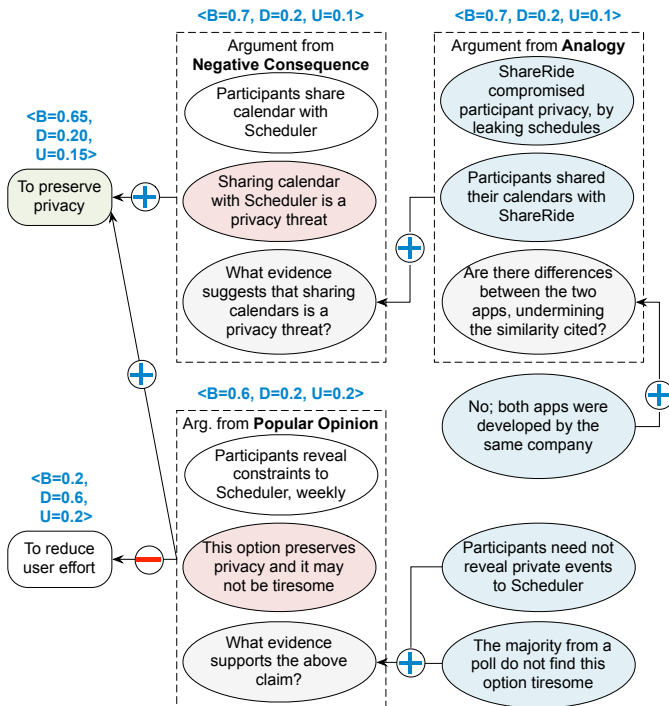


Fig. 3: Argument graph (partial) for the meeting scheduler example.

Figure 3 shows an Arg-ACH solution to resolve the conflict associated with Scheduler. Here, first, we argue from analogy that sharing calendar with Scheduler is a threat to privacy.

Figure 7 compares the difficulty ratings of Arg-ACH and ACH subjects. We observe that there is no significant difference between difficulty ratings, despite the fact that Arg-ACH subjects spent more time during analysis. Constructing arguments is a significant step in analyzing goals. Whereas Arg-ACH systematically guides an analyst to construct arguments, ACH leaves this to the imagination of the analyst. As the cognitive load of analysis increases with increase in number of goals and beliefs available, we expect Arg-ACH to reduce the difficulty of analysis significantly compared to ACH.

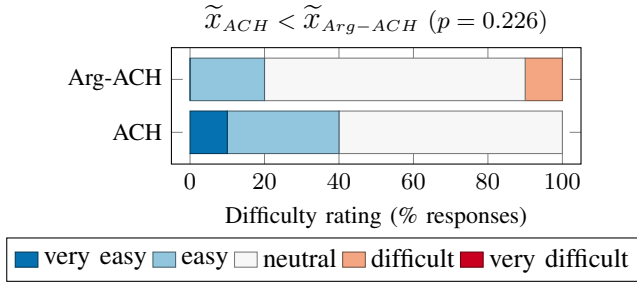


Fig. 7: Perceived difficulty of analysis.

B. Quality

Figure 8 compares the number of beliefs incorporated by Arg-ACH and ACH subjects in the analyses they produced. Although Arg-ACH subjects employed more beliefs than the ACH subjects, the difference was not significant. A potential reason for this could be the limited number of beliefs in our database. To keep the study feasible for subjects to finish, we included only 34 beliefs in the database.

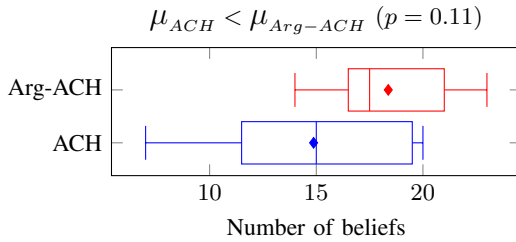


Fig. 8: Belief coverage.

In order to study the extent to which ACH and Arg-ACH support belief composition, we calculated the mean length of belief chains incorporated in each (1) row of the diagnosticity matrix of a subject for ACH, and (2) argument supporting or opposing a hypothesis for Arg-ACH.

As shown in Figure 9, the mean length of belief chains is significantly higher for Arg-ACH than for ACH. This result, though desirable, is not surprising. The only way to compose beliefs in ACH is to incorporate multiple beliefs in the same row of the diagnosticity matrix. Because of this limitation, belief composition in ACH leads to incomprehensible solutions. In contrast, Arg-ACH provides finer control for belief composition as shown in the example of Figure 2.

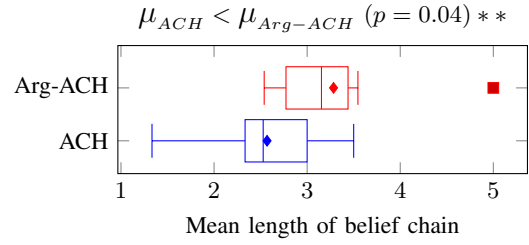


Fig. 9: Belief composition.

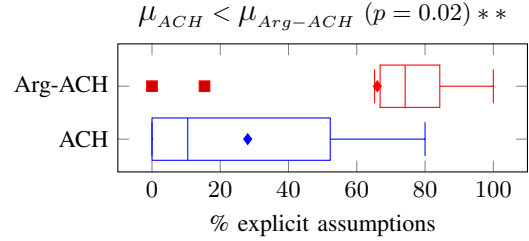


Fig. 10: Explicit assumptions.

We counted explicit and implicit assumptions made by subjects. An explicit assumption is any statement that was not in our database. Implicit assumptions include any unjustified inconsistency scores, BDU scores, or logical deductions as involving an implicit assumption (i.e., the subject must have made an assumption to assign the corresponding score or make the corresponding deduction). As shown in Figure 10, Arg-ACH yields significantly more explicit assumptions than ACH.

C. Repeatability

An analyst's major conclusion from ACH or Arg-ACH is an ordered list of goals, e.g., in our case study, the conclusion can be [trains, hotels, airport] meaning that the goal to *enhance security on trains* is most desired, followed by the goals to enhance security at *hotels* and *airport*. Figure 11 visualizes the 10 conclusions, one from each subject, within each group. We observe that seven Arg-ACH subjects concluded *trains* as most desirable. In contrast, five ACH subjects concluded *trains* and the other five concluded *airports* as most desirable. Thus, there is higher consensus among the Arg-ACH subjects than the ACH subjects about the most desirable goal.

The above observation is qualitative and considers first position (most desirable goal), only. In order to objectively measure the consensus across positions, we employed the Kemeny and Snell distance [15] (d_{KS}), a popular metric to measure similarity between ordered preferences. The d_{KS} metric considers similarity between all ordered pairs of choices across alternative preferences. The smaller the distance the more similar the preferences are. For example:

$$\begin{aligned}
 d_{KS}([\text{trains, hotels, airport}], [\text{trains, hotels, airport}]) &= 0 \\
 d_{KS}([\text{trains, hotels, airport}], [\text{trains, airport, hotels}]) &= 2 \\
 d_{KS}([\text{trains, hotels, airport}], [\text{airport, hotels, trains}]) &= 6
 \end{aligned}$$

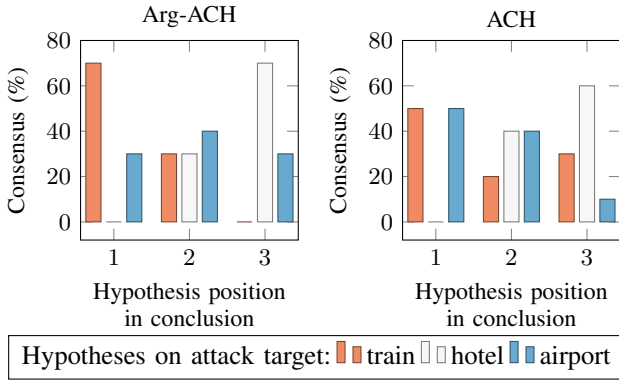


Fig. 11: Consensus among conclusions.

Considering $d_{KS}(A, B)$ values above, we can infer that there is total consensus between the preferences in the first set above, but no consensus at all in the third set since all six ordered pairs of choices are dissimilar in the third set.

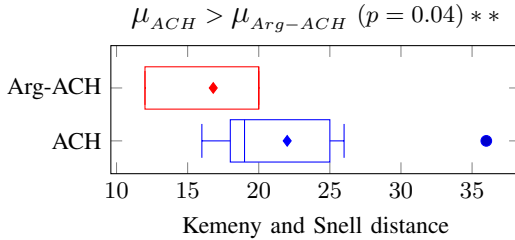


Fig. 12: Distance between conclusions.

To compute consensus between N subjects in a group, we create an $N \times N$ matrix, where the element (i, j) represents d_{KS} between the subjects S_i and S_j . We compute mean d_{KS} of the group as the mean of all the values in the matrix for that group. As shown in Figure 12, the mean d_{KS} is significantly lower for Arg-ACH than for ACH, indicating that there is significantly more consensus among Arg-ACH conclusions than among ACH conclusions.

D. Threats to Validity

We find three primary threats to the validity of our results.

Generalizing to expert analysts. All our subjects were new to both ACH and argumentation. Thus, our results may not generalize to seasoned analysts. However, we speculate that even experts may overlook details some of the time. How Arg-ACH can help them remains to be studied.

Generalizing to a larger belief base. Our belief database included only 34 beliefs. This was necessary so that subjects could finish the analysis in a reasonable amount of time. It remains to be seen if our results can be reproduced for a larger belief base. We expect that the benefits of our method will be more pronounced as the available beliefs (and associated incompleteness and ambiguity) increase.

Influence of tools used. The ACH subjects used the PARC ACH tool and the Arg-ACH subjects used the Carneades Editor for analysis. The interface of the PARC tool is

similar to that of a spreadsheet and was familiar to subjects. However, the subjects were not familiar with the interface of the Carneades Editor. This could have influenced the time spent by the subjects. How the time spent would change if the subjects are given enough time to master the argument editor remains to be studied.

VI. RELATED WORK

We describe works related to conflict resolution from the requirements engineering, multiagent systems, and intelligence analysis domains.

A. Requirements Engineering

Van Lamsweerde et al. [1] propose formal techniques and heuristics to detect conflicts from specifications of goals from multiple stakeholders. Their techniques include: (1) computing preconditions to derive the negated goal assertion backwards from other assertions, (2) detecting conflicts between goals by matching goals with existing domain-specific divergence patterns, and (3) providing rules learnt from past experiences to identify conflict patterns. Whereas these techniques emphasize detecting conflicts at the goal level, we resolve conflicts considering beliefs supporting or opposing goals.

Antón and Potts [2] infer goals from abstract or incomplete requirements and find more complete requirements from the inferred goals. Their method involves asking systematic questions to improve requirements, relaxing initial goals using obstacles, and exploring different scenarios. A key idea behind both Antón and Potts' method and Arg-ACH is to prompt stakeholders for eliciting additional beliefs. However, in contrast to broad-brush guidelines the former method offers, argumentation schemes in Arg-ACH provide specific critical questions to ask. Further, based on answers to these questions, Arg-ACH guides an analyst to resolve conflicts.

Robinson [3] describes a negotiation-based approach to resolve stakeholders' conflicting goals to produce a unified specification. Robinson considers: (1) a distributive technique where stakeholders negotiate and reach an agreement based on their utility functions, and (2) an integrative technique where stakeholders collect opposing proposals, exchange communications to learn each others' motivations, and jointly search a variety of proposals. Ideas from Arg-ACH can be easily incorporated into a negotiation-based approach, where in addition to goals and constraints, stakeholders can exchange arguments. Distributed ACH [8] is ideal for this purpose.

Massacci and Zannone [16] apply Secure Tropos to a detect conflicts between functional and security requirements in a case study. They capture security requirements for each stakeholder at both social (roles and positions in a system) and individual (entitlements, goals, objectives, and responsibilities) levels. Arg-ACH is complementary to Massacci and Zannone's approach: whereas the latter seeks to detect conflicts at an early phase of development, Arg-ACH seeks to resolve conflict by systematically capturing beliefs that led to the conflict.

Elahi and Yu [17] describe criteria for a conceptual modeling technique to model and analyze security trade-offs among

competing goals of stakeholders. Their model incorporates goals (alternative security trade-offs), actors (stakeholders), and security specific concepts (threats, vulnerabilities, and safeguards). In addition to modeling with similar constructs, Arg-ACH describes a systematic way to choose an alternative among many. Albeit, criteria described by Elahi and Yu can be valuable for developing security domain-specific argumentation schemes, which Arg-ACH can exploit.

Jureta et al. [18] propose Goal Argumentation Method (GAM) to incorporate arguments in goal modeling. In GAM, stakeholders choose appropriate requirements and transform these requirements to goal models based on arguments. A key motivation behind both GAM and Arg-ACH is to record stakeholders' decision process in coming up with requirements. However, the two approaches differ in that GAM incorporates custom arguments, whereas Arg-ACH advocates pre-refined domain-independent and domain-specific argumentation schemes as thinking aids for the analyst.

Jureta et al. [19] propose *Techne*, a modeling language applicable during early phases of requirements engineering. *Techne* provides candidate solutions to stakeholders for choosing desirable requirements and assures that the chosen requirements are consistent and satisfy domain properties. In *Techne*, stakeholders elicit requirements based on their psychological modes (beliefs, desires, intentions) and evaluate requirements by modelling relations over requirements (inference, conflict, preference, is-mandatory and is-optional). However, these relations are binary and as Jureta et al. note, in *Techne* it is not possible to enter rationale for an analyst's preference. In contrast, Arg-ACH captures an analyst's confidence in beliefs via BDU scores. Further, Arg-ACH captures the rationale for BDU scores in the form of answers to critical questions.

Ingolfo et al. [20] propose an argumentation-based approach to achieve compliance of system requirements to a given law. In this approach, stakeholders consider two inputs: a requirements model and a law model. Using these models, stakeholders iteratively discuss and revise requirements till they comply with the law model. Stakeholders revise requirements based on arguments provided for or against each requirement. Whereas Ingolfo et al. exploit argumentation for achieving compliance, Arg-ACH exploits argumentation for resolving conflicts among stakeholders' goals.

Maidenn et al. [21] incorporate arguments into i^* goal modelling. Specifically, they include satisfaction arguments, which relate domain knowledge with system specifications to produce requirements. Compared to Maidenn et al.'s approach, we use arguments to elicit stakeholder beliefs and the relationship between beliefs and goals to produce requirements and goals.

B. Multiagent Systems

Castelfranchi and Pagileri [6] argue that belief dynamics and goal processing are closely related and that goal processing is determined by belief revision. That is activating, promoting, dropping, or suspending a goal needs modification of appropriate beliefs. Arg-ACH is motivated by similar intuitions. However, Castelfranchi and Pagileri's work is limited to the

perspective of a single stakeholder, where the stakeholder resolves conflicts between its goals based on efforts required to allocate resources to achieve a goal (cost beliefs) and priority to be set to achieve a goal (preference beliefs). In contrast to their work, we focus on a system with multiple stakeholders and resolve goal conflicts by incorporating argumentation.

Sycara's [22] PERSUADER framework resolves goal conflicts through negotiation, where a mediator enters into negotiation between two stakeholders in conflict. The mediator proposes and modifies the compromises and utilities between the stakeholders till the stakeholders come to a final agreement. In contrast, Arg-ACH does not require a mediator, but incorporates stakeholders' beliefs as arguments to resolve conflicts.

Murukannaiah and Singh [23] describe Xipho, which employs contexts as a basis for resolving conflicting goals of an actor. Xipho addresses a special type of conflicts that arise based on an actor's beliefs about context, whereas Arg-ACH can be generalized to any type of beliefs.

Kakas and Moraitis [24] propose a formal framework for automated decision making for agents that is based on argumentation. Amgoud and Vesic [25] describe an abstract argumentation-based framework for decision making, where a decision maker constructs arguments for or against a candidate decision (hypothesis), and then compares arguments for various candidate decisions in order to rank those candidates. Our work is similar to these approaches in that we also address the problem of decision making, however by human analysts. Further, we propose a methodology for constructing and evaluating arguments for various hypotheses.

C. Intelligence Analysis

Pope and Jøsang [26] incorporate subjective logic in the analysis of competing hypotheses in order to reduce analysts' cognitive effort, and to produce high-quality analysis. Their method calculates likelihoods of hypotheses and evidence diagnosticity based on the analyst input. We share the same objective as Pope and Jøsang, and employ the BDU measures from subjective logic for arguments. Unlike their method, the argumentation schemes in our method guide the analyst in finding the relevant evidence for constructing robust arguments. Further, unlike Pope and Jøsang, we validate our claims about high-quality reports via an empirical study.

Wheaton and Chido [27] propose Structured ACH (SACH). SACH is an enhancement over ACH, where the analyst successively analyzes a hypothesis and refines it into more specific hypotheses. Our argument representation can naturally support such hierarchical analysis of hypotheses.

Valtora et al. [28] extend ACH to incorporate Bayesian networks. Whereas they claim to generalize ACH, incorporating Bayesian networks introduces new challenges. What network structure and prior distributions do we assume? More importantly, how do we capture the rationale behind those assumptions? Also, probabilities based purely on statistical data aggregation may not be comprehensible to humans.

Williams and Williamson [29] present a framework for breast cancer prognosis that combines Bayesian networks with

argumentation. They employ a Bayesian network constructed using the patients' data to perform the prognosis, and the argumentation framework to develop explanation of the prognosis. It will be interesting to combine Williams and Williamson's approach with ours—the Bayesian network can guide the analyst in constructing the arguments. A key concern, then, would be to study if such arguments make sense to humans.

VII. CONCLUSIONS AND DIRECTIONS

We introduce Arg-ACH, a novel argumentation-based approach for resolving conflicts among stakeholder goals of a system to be. Arg-ACH guides an analyst to systematically capture his or her rationale in deriving a conclusion (a goal). Our empirical evaluation shows that Arg-ACH yields high-quality analysis reports. Specifically, Arg-ACH improves belief coverage and composition, and requires an analyst to make his or her assumptions explicit. We find that Arg-ACH fares better than ACH in terms of repeatability, a desirable characteristic for any method. We find that Arg-ACH requires more time for analysis than ACH, though justified by improvements in quality. This speaks to the need for better tool support to enhance the efficiency of Arg-ACH.

An important direction for future work is supporting collective intelligence. For example, multiple analysts can collaborate to resolve conflicts. Arg-ACH is well-suited for such scenarios since argumentation-based reports are easy for humans to understand and scrutinize. Another direction for future work is automation. Specifically, information retrieval techniques can automate generating goals [30] and extracting beliefs for each goal. The critical questions from argument schemes can guide the automated search. Given the BDU scores of conflicting goals, Arg-ACH employs a simple heuristic to resolve the conflict: to choose the goal with the highest B score as most desirable by the stakeholders. An interesting direction is to explore alternative techniques, e.g., [31], for composing and comparing the BDU scores.

ACKNOWLEDGEMENT

Thanks to the US Department of Defense for support through the NCSU Laboratory of Analytic Sciences, and Nirav Ajmeri and the anonymous reviewers for useful comments.

REFERENCES

- [1] A. van Lamsweerde, R. Darimont, E. Letier, "Managing conflicts in goal-driven requirements engineering," *IEEE TSE*, vol. 24, no. 11, pp. 908–926, 1998.
- [2] A. Antón, C. Potts, "The use of goals to surface requirements for evolving systems," in *Proc. ICSE*, 1998, pp. 157–166.
- [3] W. Robinson, "Negotiation behavior during requirements specification," in *Proc. ICSE*, 1990, pp. 268–276.
- [4] B. Boehm, P. Bose, E. Horowitz, M. Lee, "Software requirements negotiation and renegotiation aids: A Theory-W based spiral approach," in *Proc. ICSE*, 1995, pp. 243–253.
- [5] W. Robinson, "Integrating multiple specifications using domain goals," *SIGSOFT Softw. Eng. Notes*, vol. 14, no. 3, pp. 219–226, Apr. 1989.
- [6] C. Castelfranchi, F. Paglieri, "The role of beliefs in goal dynamics: Prolegomena to a constructive theory of intentions," *Synthese*, vol. 155, no. 2, pp. 237–263, 2007.
- [7] R. Heuer Jr., R. Pherson, *Structured Analytic Techniques for Intelligence Analysis*. CQ Press, 2014.
- [8] G. Convertino, D. Billman, P. Pirolli, J. Massaro, J. Shrager, "The cache study: Group effects in computer-supported collaborative analysis," *CSCW*, vol. 17, no. 4, pp. 353–393, 2008.
- [9] C. Wickens, J. Hollands, *Engineering Psychology and Human Performance*, 2000.
- [10] J. Bentahar, B. Moulin, M. Bélanger, "A taxonomy of argumentation models used for knowledge representation," *Artif. Intell. Rev.*, vol. 33, no. 3, pp. 211–259, 2010.
- [11] D. Walton, C. Reed, F. Macagno, *Argumentation Schemes*. 2008.
- [12] A. Jøsang, "A logic for uncertain probabilities," *IJUFKS*, vol. 9, no. 3, pp. 279–311, 2001.
- [13] PARC AI Team, "ACH_{1.1}: A tool for analyzing competing hypotheses," 2005.
- [14] T. Gordon, "Carneades tools for argument (re)construction, evaluation, mapping and interchange," 2014.
- [15] J. Kemeny, J. Snell, *Mathematical Models in Social Sciences*. MIT Press, 1972.
- [16] F. Massacci, N. Zannone, "Detecting conflicts between functional and security requirements with secure Tropos: John Rusnak and the Allied Irish Bank," in *Social Modeling for Requirements Engineering*, 2011, pp. 337–362.
- [17] G. Elahi, E. Yu, "A goal oriented approach for modeling and analyzing security trade-offs," in *ER, LNCS*, 2007, vol. 4801, pp. 375–390.
- [18] I. Jureta, S. Faulkner, P. Schobbens, "Clear justification of modeling decisions for goal-oriented requirements engineering," *RE*, vol. 13, no. 2, pp. 87–115, 2008.
- [19] I. Jureta, A. Borgida, N. Ernst, J. Mylopoulos, "Techne: Towards a new generation of requirements modeling languages with goals, preferences, and inconsistency handling," in *Proc. RE*, 2010, pp. 115–124.
- [20] S. Ingolfo, A. Siena, J. Mylopoulos, A. Susi, A. Perini, "Arguing regulatory compliance of software requirements," *Data & Knowledge Engineering*, vol. 87, no. 0, pp. 279–296, 2013.
- [21] N. Maiden, J. Lockerbie, D. Randall, S. Jones, D. Bush, "Using satisfaction arguments to enhance i* modelling of an air traffic management system," in *Proc. RE*, 2007, pp. 49–52.
- [22] K. Sycara, "Resolving goal conflicts via negotiation," in *Proc. AAI*, 1988, pp. 245–250.
- [23] P. Murukannaiah, M. Singh, "Xipho: Extending Tropos to engineer context-aware personal agents," in *Proc. AAMAS*, 2014, pp. 309–316.
- [24] A. Kakas, P. Moraitis, "Argumentation based decision making for autonomous agents," in *Proc. AAMAS*, 2003, pp. 883–890.
- [25] L. Amgoud, S. Vesic, "On the use of argumentation for multiple criteria decision making," in *Advances in Computational Intell.*, 2012, vol. 300, pp. 480–489.
- [26] S. Pope, A. Jøsang, "Analysis of competing hypotheses using subjective logic," in *Proc. ICCRTS*, 2005, pp. 126–135.
- [27] K. Wheaton and D. Chido, "Structured analysis of competing hypotheses: Improving a tested intelligence methodology," *Competitive Intelligence Magazine*, vol. 9, no. 6, pp. 12–15, 2006.
- [28] M. Valtorta, M. Jiangbo, H. Goradia, and J. Huang, "Extending Heuer's analysis of competing hypotheses method to support complex decision analysis," in *Proceedings of the International Conference on Intelligence Analysis*, 2005.
- [29] M. Williams, J. Williamson, "Combining argumentation and bayesian nets for breast cancer prognosis," *J Log. Lang. Inf.*, vol. 15, no. 1-2, pp. 155–178, 2006.
- [30] S. Sohrabi, O. Udrea, A. Riabov, "Hypothesis exploration for malware detection using planning," in *Proc. AAI*, 2013, pp. 883–889.
- [31] Y. Wang, M. Singh, "Formal trust model for multiagent systems," in *Proc. IJCAI*, 2007, pp. 1551–1556.