Presented at the Second International JSAI-isAI Workshop on Argument for Agreement and Assurance (AAA 2015), Keio University, Kanagawa, Japan, November 2015; Appears in Springer LNAI Vol. 10091, pp. 331–347.

On the Interpretation Of Assurance Case Arguments

John Rushby

Computer Science Laboratory
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025 USA

Abstract. An assurance case provides a structured argument to establish a claim for a system based on evidence about the system and its environment. I propose a simple interpretation for the overall argument that uses epistemic methods for its evidential or leaf steps and logic for its reasoning or interior steps: evidential steps that cross some threshold of credibility are accepted as premises in a classical deductive interpretation of the reasoning steps. Thus, all uncertainty is located in the assessment of evidence. I argue for the utility of this interpretation.

1 Introduction

An assurance case provides an argument to justify certain claims about a system, based on evidence concerning both the system and the environment in which it operates. The claims can be about any system property, such as reliability or security, and thereby generalize the previously established notion of a safety case, where the claim is always about safety. Software assurance, in the form this is understood in the DO-178C guidelines for civil aviation [1], provides an important special case: here the top claim is one of correctness with respect to system requirements (safety of those requirements is established separately using guidelines such as ARP-4754A and ARP-4761).

Assurance cases are standard in many industries (e.g., trains and nuclear power in Europe, and some medical devices in the USA) and are being considered for others, such as civil aircraft, where changes in the operating environment and the pace of that change (e.g., integration of ground and air systems in NextGen, UAVs in civil airspace, increasingly autonomous flight systems) challenge current methods of assurance. Civil aviation has an exemplary record of safety, so there is interest in achieving greater understanding of both existing and new methods for assurance before making changes. In particular, there is work on reconstructing the argument implicit in DO-178C [18], and exploring whether assurance cases could provide the basis for future evolutions of these and related guidelines [2].

Modern safety cases developed from methods used in nuclear power, offshore oil, and process industries, where the case was based on a "narrative" about the design and operation of the plant or system and how its hazards were eliminated or mitigated. Later, as computer control became a larger part of the system, safety cases became more "structured" with an explicit argument organized in a step-by-step manner and often presented in a graphical notation such as CAE (Claims-Argument-Evidence) or GSN (Goal Structuring Notation). This paper is concerned with the *interpretation* of structured assurance case arguments; that is, we ask, what is the meaning of such an assurance case? This question is

a necessary precursor to one that will be addressed in a later paper concerning the *evaluation* of assurance cases: that is, how we can tell if an assurance case truly and adequately justifies its claim.

An assurance case is based on evidence, which is an epistemological concept: that is, it concerns our *knowledge* of the system and its environment. Hence, it seems that the interpretation of a case could or should build, at least in part, on ideas from epistemology, such as those used to formalize scientific theories. On the other hand, an assurance case also employs an *argument*, which is generally viewed as a logical concept. But even within logic, there are different ways of looking at arguments. One perspective focuses on the dialectical, back and forth interpretation of argument; that perspective will be valuable when we turn to the evaluation of assurance cases, but for their basic meaning the classical interpretation of formal logic seems more suitable. However, formal logic deals with deductive validity—that is, truth of the premises must guarantee truth of the conclusion—whereas an assurance case must acknowledge uncertainties in the world and in our knowledge about it, so that truth of the premises may do no more than strongly suggest the conclusion. This is generally referred to as inductive validity (an unfortunate overloading of the term "inductive," which has many other meanings in logic and science) and its interpretation requires a departure from the well-established semantics of classical logic into more contentious areas such as probability logic, fuzzy logic, or evidential reasoning.

Thus, interpretation of assurance cases must reconcile their epistemic and logical aspects, and must acknowledge their inductive character. Furthermore, many industries employ graduated levels of assurance: systems that pose greater risk are subjected to more intense assurance. If this graduation is framed in terms of assurance cases, then it seems that in addition to the inductive validity (sometimes referred to as the cogency) of a case, we must also address the "strength" of that validity. One way in which assurance case arguments may be strengthened is by inclusion of *confidence claims*. These are elements whose falsity would not invalidate the argument but whose truth strengthens it. Clearly, such elements are not part of standard logical interpretations.

Accordingly, some look to rather radical reformulations of the idea of argument, such as Toulmin's treatment [3], or probability logics [4]. In this paper, by contrast, I propose a very simple combination of classical methods and argue for its utility. I present the approach in the following section, provide brief comparison with other methods in Section 3, and conclusions in Section 4.

2 Structure and Interpretation of Assurance Arguments

As noted in the introduction, an assurance case is composed of three elements: a *claim* that states the property to be assured, *evidence* about the system and its environment, and a *structured argument* that the evidence is sufficient to establish the claim. The structured argument is a hierarchical collection of individual argument steps, each of which justifies a local claim on the basis of evidence and/or lower-level subclaims. A trivial example is shown on the left in Figure 1, where a claim $\texttt{C}$ is justified by an argument step $\texttt{AS}_1$ on the basis of evidence $\texttt{E}_3$ and subclaim $\texttt{SC}_1$, which itself is justified by argument step $\texttt{AS}_2$ on

Here, **C** indicates a claim, **SC** a subclaim, and **E** evidence; **AS** indicates a generic argument step, **RS** a reasoning step, and **ES** an evidential step.

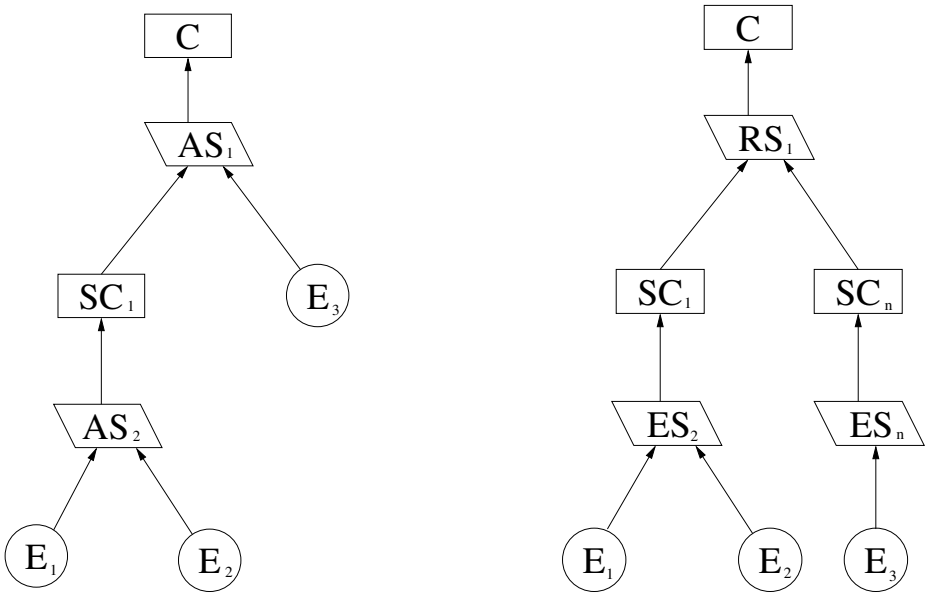

Fig. 1. Converting an Argument from Free (left) to Simple Form (right)

the basis of evidence $\mathtt{E}_1$ and $\mathtt{E}_2$. The figure is generic and not representative of any specific notation, although its element shapes are from GSN (but the arrows are reversed, as in CAE). Note that a structured argument may not be a tree because subclaims and evidence can support more than one argument step.

Observe that the argument step $\mathtt{AS}_1$ uses both evidence $\mathtt{E}_3$ and a subclaim $\mathtt{SC}_1$. We will see later how to provide an interpretation to such "mixed" argument steps, but it is easier to understand the basic approach in their absence. By introducing additional subclaims where necessary, it is straightforward to convert arguments into *simple form* where each argument step is supported either by subclaims (i.e., a *reasoning step*) or by evidence (i.e., an *evidential step*), but not by a combination of the two (many assurance cases will already have this form—it is natural in GSN, for example). In Figure 1, the "mixed" or free argument on the left is converted to simple form on the right by introducing a new subclaim $\mathtt{SC}_n$ and new evidential argument step $\mathtt{ES}_n$ above $\mathtt{E}_3$. Argument steps $\mathtt{AS}_1$ and $\mathtt{AS}_2$ are relabeled as reasoning step $\mathtt{RS}_1$ and evidential step $\mathtt{ES}_2$, respectively.

The key to our approach is that the two kinds of argument step are interpreted differently. Specifically, evidential steps are interpreted epistemically, using ideas grounded in probability, while reasoning steps are interpreted in logic: subclaims supported by evidential steps that cross some threshold of credibility are accepted as premises in a classical deductive interpretation of the reasoning steps. We now consider these two kinds of argument steps in more detail.

2.1 Evidential Steps

Evidential steps are the bridge between our concepts about both the system and its environment, which we express as subclaims, and our observations concerning these, which we document as evidence; in other words, they represent our *knowledge* about the system and its environment. What it means to really know something is the topic of epistemology, a branch of philosophy that dates back to the ancient Greeks and provides much insight but no generally accepted treatment. Our focus is a rather more specific than the general theory of knowledge: we want to know what it means for evidence to support a claim.

The intuitive idea is that the evidence in support of a hypothesis or claim should be "weighed" and the hypothesis accepted as a "settled fact" if that weight exceeds some threshold. Modern treatments of this topic derive from the work of I. J. (Jack) Good who frames it in terms of probabilities and Bayesian inference [5]. Good began his work in the codebreaking activity at Bletchley Park during the Second World War and reference [6] recounts some of this history. Recent developments of these ideas are found in Bayesian Epistemology [7] and their application to the theory of science is known as Bayesian Confirmation Theory [8]. Related ideas are developed also in legal theory [9].

When we have evidence E supporting a hypothesis or claim C, it seems plausible that our procedure should be to assess $P(C \mid E)$ and to accept C when this probability exceeds some threshold. Unfortunately, assessment of $P(C \mid E)$ poses difficulties. All the quantities under consideration here are subjective probabilities that express human judgement [10] and even experts find it difficult to directly assess a quantity such as $P(C \mid E)$. Furthermore, the significance of $P(C \mid E)$ depends on our prior assessment $P(C)$, which could be one of ignorance (or, in law, prejudice). Rather than attempt directly to assess $P(C \mid E)$, it seems that we should factor the problem into alternative quantities that are easier to assess and of separate significance.

The basic idea of Good and others is that the strength or "weight" of evidence is some function of $P(E \mid C)$. This is related to $P(C \mid E)$ by Bayes' Theorem but seems easier to assess (that is, it seems easier to estimate the likelihood of concrete observations, given a claim about the world, than vice-versa). Furthermore, what we are really interested in is the ability of E to discriminate between C and its negation $\neg C$, so the quantities we should look at are the difference or ratio (or logarithms of these) between $P(E \mid C)$ and $P(E \mid \neg C)$. Such quantities are referred to as *confirmation measures* and may be said to weigh C and $\neg C$ "in the balance" provided by E.

There is no agreement in the literature on the best confirmation measure: Fitelson [11] considers several and makes a strong case for Good's measure $\log \frac{P(E \mid C)}{P(E \mid \neg C)}$, Tentori and colleagues [12] perform an empirical comparison and generally approve of Kemeny and Oppenheim's measure $\frac{P(E \mid C) - P(E \mid \neg C)}{P(E \mid C) + P(E \mid \neg C)}$, while Joyce [13] argues that different measures serve different purposes.

In criminal law, there is (or was until recently) a reluctance to convict the innocent, even at the price of acquiting some who are guilty, so Gardner-Medwin [14] suggests that appropriate probabilistic criteria for conviction are those that

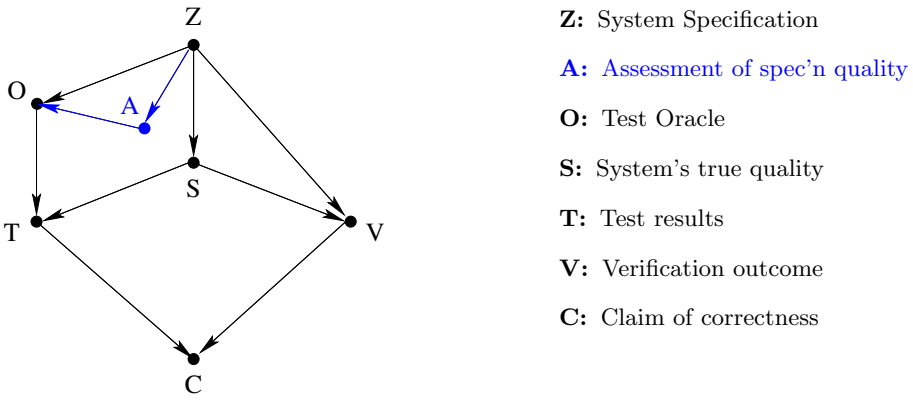

Fig. 2. BBN for Testing and Verification Evidence

indicate the evidence could very likely have arisen if the defendant is guilty but not if they are innocent—and confirmation measures have this property.

It is a topic for debate whether the criteria for acceptance of evidential steps in assurance cases should use a confirmation measure (so that, as in a criminal trial, we can reduce the chance of accepting a false claim, even at the price of rejecting some good ones) or one that more directly assesses the claim (thereby maximizing utility but possibly accepting some false claims).

My own view is that the final decision is a human judgement that should consider several quantities and measures. It is not necessary to attach numerical estimates to the probabilities nor to actually evaluate the measures, but understanding the basis for their construction can inform our judgement. This judgement is more difficult when several items of evidence are combined to support a subclaim: the items may not be independent so accurate analysis and evaluation requires more sophisticated probabilistic modeling techniques, such as Bayesian Belief Networks (BBNs) and their tools. Again, informal rather than quantitative modeling and analyses may be used in practice, but it is useful to hone our judgement with numerical experiments that allow sensitivity and "what-if" explorations. An example of such an exploration is presented below.

Bayes' Theorem is the principal tool for analyzing conditional subjective probabilities: it allows a prior assessment of probability to be updated by new evidence to yield a rational posterior probability. It is difficult to calculate over large numbers of complex conditional (i.e., interdependent) probabilities, but usually the dependencies are relatively sparse and can conveniently be represented by a graph (or "net"—the term used in BBNs) in which arcs indicate dependencies. An example, taken from [15], is shown above in Figure 2. This represents a "multi-legged" evidential argument step in which evidence from testing is combined with that from formal verification. The nodes of the graph represent judgments about components of the argument step and the arcs indicate dependencies between these (ignore, for the time being, the arcs associated with A and shown in blue).

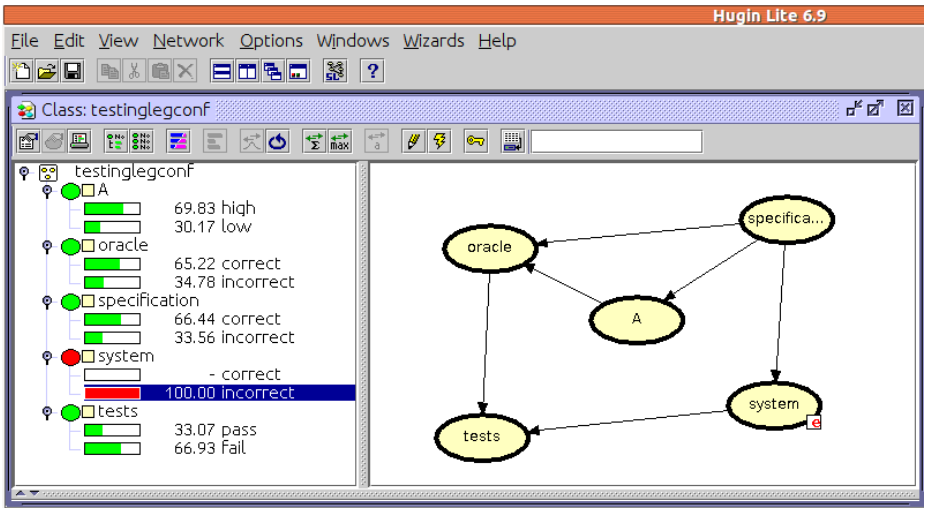

Fig. 3. Hugin Analysis of BBN for Testing Evidence Alone

The nodes of the graph actually represent random variables but we can most easily understand the construction of the graph by first considering the artifacts from which these are derived. Here, Z is the system specification; from this are derived the actual system S and the test oracle O. Tests T are dependent on both the oracle and the system, while formal verification V depends on the system and its specification. The claim of correctness C is based on both the test results and the formal verification.

For reasons explained later, I think it is best to treat the verification and testing "legs" of the evidence separately, so let us focus on the testing leg alone (i.e., ignore everything involving V and C); this is shown represented inside the BBN tool *Hugin Expert* [16] in Figure 3. Here, the interpretation of Z is a random variable representing correctness of the system specification: it has two possible values: `correct` (i.e., it achieves the requirements established for the system) or `incorrect`. The assessor must attach some prior probability distribution to these (e.g., 99% confidence it is `correct`, vs. 1% that it is `incorrect`).

S is a variable that represents the true (but unknown) quality of the system, stated as a probability of failure on demand (that is, failure wrt. requirements). This probability depends on Z: we might suppose that it is 0.99 if Z is `correct`, but only 0.5 if it is `incorrect`.

O is a variable that represents correctness of the test oracle; this is derived in some way from the specification Z and its probability distribution will be some function of the correctness of Z (e.g., if Z is `correct`, we might suppose it is 95% probable that O is `correct`, but if Z is `incorrect`, then it is only 2% probable that O is `correct`).

T is a Boolean variable that represents the outcome of testing. It depends on both the oracle O and the true quality of the system S. Its probability distribu-

tion over these is represented by a joint probability table such as the following, which gives the probabilities that the test outcome is judged successful.

Correct System		Incorrect System	
Correct Oracle	Bad Oracle	Correct Oracle	Bad Oracle
100%	50%	5%	30%

In this example, only T is directly observable. Using a BBN tool such as Hugin, it is possible to conduct "what if" exercises on this model to see how prior estimates for the conditional probability distributions of the various nodes are updated by evidence. In particular, Hugin allows the user to manipulate the values of some variables and observe the impact on others. In Figure 3, we have hypothesized the system is incorrect (indicated by the red bar, and set by double-clicking on the value) and can see that the conditional probability that testing succeeds (i.e., $P(E \mid \neg C)$ for this example) is 33.07%. If the system is assumed correct, the probability that testing succeeds (i.e., $P(E \mid C)$) is 98.53%. Hence, the Kemeny-Oppenheim confirmation measure is 0.49. We can also examine the probability of a correct system, given that testing succeeds (i.e., $P(C \mid E)$), which evaluates to 99.49%, or given that it fails (i.e., $P(C \mid \neg E)$), which is 59.21%.

We see that in this model the assumed prior distributions are such that testing has rather poor evidential weight: it is rather likely that an incorrect system will be accepted or that a rejected system is in fact correct. Further inspection and experimentation will show that part of the explanation is that the modeled test oracle is of low quality. The variable O has strong impact on the test outcome T but is not itself observed or evaluated. We might suppose that reliability of the testing procedure would be improved if we could assess the quality of the test oracle and require this to exceed some threshold. However, it is not easy to see how this artifact can be assessed directly, so an alternative might be to assess the quality of the specification Z, since this has a large impact on the quality of the oracle.

Reasoning similar to this may implicitly underlie some of the DO-178C guidelines for software assurance in civil aircraft [1]. For the most critical software, DO-178C specifies 71 assurance "objectives" that must be accomplished and several of these concern the quality of requirements and specifications. For example, its Section 6.3.2.d specifies the objective to "ensure that each low-level requirement can be verified." We can introduce this idea into our model as the variable A in Figure 2 (with dependencies indicated in blue) and similarly in Figure 3. Here A assesses "confidence" that the specification Z is testable and takes values `high` and `low`; we suppose the probability that A is `high` is 95% when Z is `correct` and 20% when it is `incorrect`. There is no arc from A to S because A is not a general evaluation of the specification, just its testability. The probability distribution of O will now depend on both A and Z and we might suppose it takes the following form.

Correct Specification		Incorrect Specification	
High Confidence	Low Confidence	High Confidence	Low Confidence
99%	70%	2%	1%

If we require that A is `high` before we undertake testing, then we find that the probability of accepting an incorrect system is reduced from 33.07% to 13.33% while the probability of accepting a correct system increases from 98.53% to 99.45%. Hence, the Kemeny-Oppenheim confirmation measure improves from 0.49 to 0.76. The probability the system is correct, given that testing succeeds, improves from 99.49% to 99.85% and, if testing fails, the probability the system is correct reduces from 59.21% to 36.33%.

The probabilities and distributions used in this exercise were "plucked from the air" and cannot be considered realistic. It is possible that experts could provide realistic prior distributions for models such as these, and thereby derive credible posterior estimates. However, I do not think that is the main value in these exercises. Rather, I believe that "what-if" explorations help develop understanding of the relationships among variables and, more particularly, can help guide selection of evidence and the informal criteria to be used in deciding when the totality of evidential support allows a claim to be regarded as a "settled fact." Thus, although probabilistic modeling provides the underlying semantics and sharpens our understanding, the evidential steps in an assurance case argument may well be comprised of objectives similar (but better justified) to those developed using informal methods in guidelines such as DO-178C.

When all the objectives of an evidential step are satisfied, its subclaim is accepted as a premise in the logical interpretation of the reasoning steps of the argument, as explained in the following section.

2.2 Reasoning Steps

We have seen that in evidential steps, the separate items of evidence are "combined" to justify truth of the claim concerned. This combination may be performed informally or it can use probabilistic modeling with BBNs, as in the example, where we saw testing evidence combined with "confidence" evidence about testability of the specification.

In contrast, I propose that the subclaims appearing in reasoning steps should be *conjoined* to deliver the truth of their parent claim: that is, the claim in a reasoning step is considered true only if all its subclaims are so.[1] This interpretation could be inductive, that is the conjunction of subclaims strongly *suggests* the claim, or it could be deductive, meaning the conjunction *implies* (or entails, or proves) the claim. Let us accept for the time being that logic does provide the appropriate interpretation for reasoning steps and focus on whether this should be deductive or inductive. I claim it should be deductive and advance two reasons. The first concerns modular reasoning.

Assurance cases are generally very large and cannot truly be comprehended *in toto*: a modular or compositional method is essential. Deductive reasoning steps can be assessed in just such a modular fashion, one step or one claim at a time. First, we check local soundness: that is, for each reasoning step, we must assure ourselves that the conjunction of subclaims truly implies the claim.

[1] Some would allow disjunctions and general logical expressions. My opinion is that these are the hallmarks of evidential—rather than reasoning—steps.

Second, we must check that claims are interpreted consistently between the steps that establish them and the steps that use them; this, too, is a modular process, performed one claim at a time.

In contrast, the first of these is not modular for inductive steps—for when a step is labeled inductive, we are admitting a "gap" in our reasoning: we must surely believe either that the gap is insignificant, in which case we could have labeled the step deductive, or that it is taken care of elsewhere, in which case the reasoning is not modular.

My second reason for deprecating inductive reasoning steps is that there is no effective way to estimate the size of the gap in our reasoning. We may surely assume that any inductive step is "almost" deductive. That is to say, the following generic inductive step

$$p_1 \texttt{ AND } p_2 \texttt{ AND } \cdots \texttt{ AND } p_n \texttt{ SUGGESTS } c \tag{1}$$

would become deductive if some missing (and presumably unknown) subclaim or assumption a (which, of course, may actually be a conjunction of smaller subclaims) were added, as shown below. (It may be necessary to adjust the existing subclaims p_1 to p'_1 and so on if, for example, the originals are inconsistent with a).

$$a \texttt{ AND } p'_1 \texttt{ AND } p'_2 \texttt{ AND } \cdots \texttt{ AND } p'_n \texttt{ IMPLIES } c. \tag{2}$$

If we cannot imagine such a "repair," then surely (1) must be utterly fallacious. It then seems that any estimation of the doubt in an inductive step like (1) must concern the gap represented by a. Now, if we knew anything at all about a it would be irresponsible not to add it to the argument. But since we did not do so, we must be ignorant of a and it follows that we cannot estimate the doubt in inductive argument steps.

If we cannot estimate the magnitude of our doubt, can we at least reduce it? This seems to be the purpose of "confidence claims," but what exactly is their logical rôle? One possibility is that confidence claims eliminate some sources of doubt. For example, we may doubt that the subclaims imply the claim *in general*, but the confidence claims restrict the circumstances so that the implication is true *in this case*. But such use of confidence claims amounts to a "repair" in the sense used above: these claims are really assumptions that should be added to the argument as conventional subclaims (see the discussion of assumptions in Section 2.4 below), thereby making it deductively sound, or at least less inductive.

The logical rôle of other kinds of confidence claims is less clear; one possibility is that they serve to justify the reasoning involved. Some justification *why* the conjunction of subclaims is believed to suggest (or imply) the claim is, of course required, but I would expect it to take the form of a narrative explanation (as it does in CAE, for example), rather than a claim. On the other hand, Hawkins *et al* [17] propose that confidence claims are removed from the main safety argument but linked to it through assurance claim points (ACPs) at restricted locations to yield "assured safety arguments" that have the flavor of justification. However, although this improves the readability of arguments that use confidence claims, Hawkins *et al* provide no guidance on how to assess their contribution.

Thus, there seems to be no established or proposed method to assess the contribution of confidence claims to inductive reasoning steps. In my opinion, their

use opens Pandora's Box, for there is no way to determine that we have "enough" and thus a temptation to employ complex, but still inductive, reasoning steps, buttressed with numerous confidence claims "just in case."

My opinion is that inductive reasoning sets too low a bar and confidence claims do nothing to raise it. Hence, I recommend that reasoning steps should be deductive, for then it is very clear what their evaluation must accomplish: it must review the content and justification of the step and assent (or not) to the proposition that its subclaims truly imply the claim. There is no rôle for confidence claims in deductive reasoning steps and other superfluous subclaims are likely to complicate rather than strengthen the assessment. Hence, the requirement for deductive soundness encourages the formulation of precise subclaims and concise arguments.

2.3 Complete Arguments

We have considered evidential steps and reasoning steps separately, now we need to put them together. For arguments that are in simple form, this is easy because they are composed of just those two kinds of steps. The interpretation of a complete argument in simple form is a deductive logical interpretation in which evidentially-supported subclaims are treated as premises that are interpreted epistemically and accepted as true when their weight of evidence is considered to have crossed some threshold (which may assessed either by probabilistic modeling and analysis, or by informal judgment grounded on such modeling). Observe that although the reasoning steps are deductive, the evidential steps admit doubt, and hence the overall argument is inductive.

We will say that an argument in simple form is *sound* if its reasoning steps are deductively valid and its evidential steps all cross the thresholds established for their claims to be accepted. It seems plausible that the "weight" established for those thresholds could be used to assess the *strength* of a sound argument. We will consider that topic shortly but first consider arguments that are not in simple form.

2.4 Assumptions, and Arguments Not In Simple Form

I propose two approaches for arguments that are not in simple form. One is to convert them to simple form by the transformation suggested in Figure 1. The other is to attempt a direct interpretation by treating subclaims appearing in "mixed" argument steps as assumptions. The generic inductive step (1) could be augmented by an assumption a to make it deductive and then be written as

$$p_1 \texttt{ AND } p_2 \texttt{ AND } \cdots \texttt{ AND } p_n \texttt{ IMPLIES } c, \texttt{ ASSUMING } a.$$

Assumptions are generally treated as additional premises, so this is interpreted as

$$a \texttt{ IMPLIES } (p_1 \texttt{ AND } p_2 \texttt{ AND } \cdots \texttt{ AND } p_n \texttt{ IMPLIES } c),$$

which simplifies under the laws of logic to

$$a \texttt{ AND } p_1 \texttt{ AND } p_2 \texttt{ AND } \cdots \texttt{ AND } p_n \texttt{ IMPLIES } c.$$

Thus, we see that any subclaim can be interpreted as an assumption. This observation might seem trivial, but there are cases where it is useful. In particular,

p_1 might only "make sense" if a is true. For example, in some notations a term such as $\frac{y}{x}$ does not "make sense" unless $x \neq 0$, so we should not even inspect the expression corresponding to p_1 until we know the assumption a is true. Since all the subclaims in an assurance case argument must be true if we are to conclude its top claim, we could allow each subclaim to be interpreted under the assumption that all other subclaims are true. However, it can require additional analysis to ensure there is no circularity in this reasoning, so a useful compromise is to impose a left-to-right reading (this strategy is employed effectively in some predicatively-subtyped languages, such as PVS). In concrete terms, this means that each subclaim or item of evidence named in an argument step is evaluated assuming the truth of all subclaims appearing earlier in the argument.

The challenge of "mixed" argument steps such as $\texttt{AS}_1$ on the left of Figure 1 is whether to interpret them epistemically, like evidential steps, or logically like reasoning steps. Now that we understand assumptions, my suggestion is that the combination of evidence appearing in the step should be interpreted epistemically, under an assumption comprised of the conjunction of subclaims appearing in the same step. Thus, for example, $\texttt{AS}_1$ on the left of Figure 1 would be interpreted as an evidential step in which the evidence $\texttt{E}_3$ is evaluated under assumptions represented by the subclaim $\texttt{SC}_1$. This is effectively the same interpretation as for the transformed argument on the right of Figure 1: there, the evidential step $\texttt{ES}_n$ can use $\texttt{SC}_1$ as an assumption when interpreting $\texttt{E}_3$ since it appears earlier in the argument.

2.5 Graduated Assurance

DO-178C recognizes that aircraft software deployed in different functions may pose different levels of risk and it accepts reduced assurance for that which poses less risk. For example, the number of assurance objectives is reduced from 71 for Level A software (that with the potential for a "catastrophic" failure condition) to 69 for Level B, 62 for Level C, and 26 for Level D, and the number of objectives that must be performed "with independence" is likewise reduced from 33 to 21, 8, and 5, respectively. This is an example of *graduated assurance*, and it is found in similar form in many standards and guidelines.

On the one hand, this seems very reasonable, but on the other it poses a serious challenge to the idea that an assurance case argument should be sound. We may suppose that the Level A argument is sound, but how can the lower levels be so when they deliberately remove or weaken some of the supporting evidence and, presumably, the implicit argument associated with them?

There seem to be three ways in which an explicit assurance case argument can be weakened in support of graduated assurance. First, we could simply eliminate certain subclaims or, equivalently, provide no evidence for them. This surely renders the full argument unsound: any deductive reasoning step that employs the eliminated or trivialized subclaim cannot remain deductively sound with one of its premises removed (unless there is redundancy among them, in which case the original argument should be simplified). This approach reduces a deductively sound argument to one that is, at best, inductive, and possibly unsound; consequently, I deprecate this approach.

Second, we could eliminate or weaken some of the evidence supplied in support of selected subclaims. This is equivalent to lowering the thresholds on what constitutes a "settled fact" and does not threaten the soundness of the argument, but does seem to reduce its strength. Intuitively, the *strength* of a sound assurance case argument is a measure of its evidential support—that is the threshold weights that determine settled facts.

It could be argued that there is surely no difference between lowering the threshold for evidential support of a given claim and using that same evidence to provide strong support for a weaker claim, which could then provide only inductive support to those reasoning steps that use it—yet I approve the former and deprecate the latter. My justification is pragmatic: we have a rigorous procedure for evaluating deductive reasoning steps but not inductive ones.

Third, we could restructure the argument. Holloway's reconstruction of the argument implicit in DO-178C [18] suggests that this underpins the changes from Level C to Level D of DO-178C. The Low Level Requirements (LLR) and all their attendant objectives are eliminated in going from Level C to Level D; the overall strategy of the argument, based on showing correctness of the executable code with respect to the system requirements, remains the same, but now employs a single step from high level requirements to source code without the LLR to provide an intermediate bridge. This also seems a valid form of weakening. Notice that evidence that is common to Levels C and D could use the same thresholds so the strength of their common parts will be the same; yet it seems clear that Level C is a stronger argument than Level D. Thus, it appears there is more to the strength—or, more accurately, the persuasiveness—of an argument than deductive validity and evidential thresholds. Level C is a bigger argument and has more evidence than Level D, but this is a crude measure; it seems more credible that the persuasive strength of an argument is related to its ability to withstand challenges, which is an idea we will return to briefly in the conclusion.

3 Comparisons With Other Approaches

Other approaches proposed for the interpretation of assurance arguments fall into three classes; I consider each in turn.

The first are probabilistic interpretations: for example, [19], which applies Dempster-Shafer analysis to complete assurance cases, and [20] which uses BBNs in a similar way. These methods are insensitive to the logical content of reasoning steps, so in effect they flatten the argument by removing subclaims so that only evidence is left. But this takes us back to approaches such as DO-178C, where all we have is a collection of evidence, and loses the essence of argument-based assurance. That is the reason that I chose to separate the testing leg of Figure 3 from the multi-legged assurance case in Figure 2: rather than combine all available evidence, I consider it better to evaluate testing and verification evidence separately, and then develop a logical argument for their joint use that would consider and mitigate their strengths and weaknesses.

A second approach is that of Toulmin [3]. Papers on assurance cases frequently cite Toulmin but do not spell out how his methods should be used.

Toulmin's approach is radical and challenges some of the fundamentals of logic: namely, that the validity of our reasoning can be assessed separately from the truth of our premises. This may have some appeal in highly contested areas such as religion or ethics where participants might disagree on basic principles, but seems less appropriate for assurance cases where disagreements concern reasoning, evidence, and the interpretation of these.

A third class of approaches builds on methods from the field of argumentation and agreement [21], including defeasible reasoning and argumentation structures. I am entirely sympathetic to the use of these ideas, in particular the notion of a "defeater," to evaluate the quality or persuasiveness of an assurance case, but I do not think they offer new insight into the basic interpretation of a case.

4 Conclusions

I have proposed a two-part process for interpretation of assurance case arguments: evidential or leaf steps are interpreted epistemically by methods that can be grounded in probability, even if performed informally, while interior or reasoning steps are interpreted in deductive logic. The overall argument is inductive (i.e., admits uncertainty) but all uncertainty is located in assessment of evidence.

A natural objection to this proposal is that it may be very difficult to construct strictly deductive reasoning steps, and still harder to assemble these into a complete argument. One response is that this may accurately reflect the true difficulty of our enterprise—assurance is hard—so that simplifications achieved through inductive reasoning will be illusory. Note that, while the top claim and some of the evidential subclaims may be fixed (by regulation and by available evidence, respectively), we are free to choose the others. Just as formulation of good lemmas can simplify a mathematical proof, so skillful formulation of subclaims may make a deductive assurance argument tractable. I speculate that software assurance cases, where the top claim is correctness, may lend themselves more readily to deductive arguments than other cases, where the top claim is a system property such as safety. Experiments are needed to evaluate these claims.

This approach is simple, even obvious, but I have not seen it explicitly described elsewhere. Haley and colleagues [22] describe a method where reasoning steps are evaluated in formal logic (they call this the *Outer Argument*) while evidential steps are evaluated informally (they call this the *Inner Argument*). They acknowledge that the inner argument concerns "claims about the world" [22, pp. 140] but use Toulmin's approach rather than explicitly epistemic methods.

The proposed interpretation provides criteria for assessing the soundness of an assurance case argument and the strength of its evidential support. But it does not provide a means to evaluate whether the total argument is adequately convincing and persuasive. For that, I believe methods from dialectics and defeasible reasoning may be suitable and I will address these in a subsequent paper.

Acknowledgments. This work was partially funded by NASA under a contract to Boeing, and by SRI International. I benefited from many suggestions by Michael Holloway, our NASA contract monitor. Thoughtful comments by the anonymous reviewers improved the presentation of this material.

References

1. RTCA, Washington, DC: DO-178C: Software Considerations in Airborne Systems and Equipment Certification. (2011) 1, 7
2. Rushby, J.: The interpretation and evaluation of assurance cases. Tech Report SRI-CSL-15-01, Computer Science Lab, SRI International, Menlo Park, CA (2015) 1
3. Toulmin, S.E.: The Uses of Argument. Cambridge Univ. Press (2003) Updated edition (the original is dated 1958). 2, 12
4. Adams, E.W.: A Primer of Probability Logic. Center for the Study of Language and Information (CSLI), Stanford University (1998) 2
5. Good, I.J.: Probability and the Weighing of Evidence. Charles Griffin, London, UK (1950) 4
6. Good, I.J.: Weight of evidence: A brief survey. In Bernardo, J., et al., eds.: Bayesian Statistics 2: Proceedings of the Second Valencia International Meeting, Valencia, Spain (1983) 249–270 4
7. Bovens, L., Hartmann, S.: Bayesian Epistemology. Oxford Univ. Press (2003) 4
8. Earman, J.: Bayes or Bust? A Critical Examination of Bayesian Confirmation Theory. MIT Press (1992) 4
9. Dawid, A.P.: Bayes's theorem and weighing evidence by juries. In Swinburne, R., ed.: Bayes's Theorem. Proceedings of the British Academy (2002) 71–90 4
10. Jeffrey, R.: Subjective Probability: The Real Thing. Cambridge U. Press (2004) 4
11. Fitelson, B.: Studies in Bayesian Confirmation Theory. PhD thesis, Department of Philosophy, University of Wisconsin, Madison (2001) 4
12. Tentori, K., Crupi, V., Bonini, N., Osherson, D.: Comparison of confirmation measures. Cognition **103** (2007) 107–119 4
13. Joyce, J.M.: On the plurality of probabilist measures of evidential relevance. In: Bayesian Epistemology Workshop of the 26th International Wittgenstein Symposium, Kirchberg, Austria (2003) 4
14. Gardner-Medwin, T.: What probability should a jury address? Significance **2** (2005) 9–12 4
15. Littlewood, B., Wright, D.: The use of multi-legged arguments to increase confidence in safety claims for software-based systems: a study based on a BBN analysis of an idealised example. IEEE Trans. on Software Eng. **33** (2007) 347–365 5
16. HUGIN Expert: Hugin home page. (Retrieved 2015) http://www.hugin.com/. 6
17. Hawkins, R., Kelly, T., Knight, J., Graydon, P.: A new approach to creating clear safety arguments. In Dale, C., Anderson, T., eds.: Advances in System Safety: Proceedings of the Nineteenth Safety-Critical Systems Symposium, Southampton, UK, Springer (2011) 3–23 9
18. Holloway, C.M.: Explicate '78: Discovering the implicit assurance case in DO-178C. In Parsons, M., Anderson, T., eds.: Engineering Systems for Safety. Proceedings of the 23rd Safety-critical Systems Symposium, Bristol, UK (2015) 205–225 1, 12
19. Zeng, F., Lu, M., Zhong, D.: Using D-S evidence theory to evaluation of confidence in safety case. Journal of Theoretical and Applied Information Technology **47** (2013) 184–189 12
20. Denney, E., Pai, G., Habli, I.: Towards measurement of confidence in safety cases. In: Fifth International Symposium on Empirical Software Engineering and Measurement (ESEM), Banff, Canada, IEEE Computer Society (2011) 380–383 12
21. Ossowski, S., ed.: Agreement Technologies. Law, Governance and Technology Series, vol. 8. Springer (2013) 13
22. Haley, C.B., Laney, R., Moffett, J.D., Nuseibeh, B.: Security requirements engineering: A framework for representation and analysis. IEEE Transactions on Software Engineering **34** (2008) 133–153 13