Measuring reuse in hazard analysis

Shamus P. Smith 1 Michael D. Harrison

The Dependability Interdisciplinary Research Collaboration, Department of Computer Science, University of York, York YO10 5DD, United Kingdom.

Abstract

Hazard analysis for safety-critical systems require sufficient coverage and rigour to instill confidence that the majority of hazardous consequences have been identified. These requirements are commonly met through the use of exhaustive hazard analysis techniques. However such techniques are time consuming and error-prone. As an attempt at exhaustive coverage, hazard analysts typically employ reuse mechanisms such as copy-and-paste. Unfortunately if reuse is applied inappropriately there is a risk that the reuse is at the cost of rigour in the analysis. This potential risk to the validity of the analysis is dependent on the nature and amount of reuse applied.

This paper investigates hazard analysis reuse over two case studies. Initially reuse in an existing safety argument is described. Argument structures within the hazard analysis are identified and the amount of verbatim reuse examined. A second study is concerned with how reuse changes as a result of tool support. In contrast to the first case, the defined arguments are more diverse - reuse has occurred but is less verbatim in nature. Although tool support has aided the customisation of the reused arguments, many are only trivially customised. An edit distance algorithm is utilised to identify and enumerate verbatim and trivial reuse in the arguments.

Key words: Safety arguments, reuse, hazard analysis, edit distance
1 Introduction

Descriptive dependability arguments\(^2\) have become a standard part of the process of determining the dependability of a system. At the centre of this demonstration process is the use of techniques for systematic hazard analysis. Hazard identification, classification and mitigation techniques establish that either hazards can be avoided or that they will not affect the dependability of the system. Descriptive arguments are commonly produced to mitigate the *perceived* severity of hazards.

In such a process there are two main requirements that need to be fulfilled, that the analysis has (i) sufficient rigour and (ii) sufficient coverage. Our confidence in the rigour of a safety case, of which a hazard analysis is a component, is directly linked to the confidence in the hazard analysis itself. This confidence will be reinforced by objective evidence of coverage and depth of the analysis - that there are no unexpected adverse consequences within a safety-critical system. In recognition of these issues a range of methods have been developed to support systematic hazard analysis, for example HAZOP (Hazard and Operability Studies) [11], FMEA (Failure Modes and Effect Analysis) [6] and THEA (Technique for Human Error Assessment) [15]. Methods such as these commonly involve significant personnel effort and time commitment. As a result, the reuse of analysis fragments is common.

---

\(^1\) Corresponding author. Tel.: +44-1904-434755; Fax +44-1904-432767.

*E-mail addresses*: shamus.smith@cs.york.ac.uk (S. Smith), michael.harrison@ncl.ac.uk (M. Harrison)

\(^2\) We consider descriptive arguments as informal arguments in contrast to more quantitative, numeric arguments.
Taylor [20, pg69] notes “one of the most effective techniques for getting through an analysis is to use analogy, or repetition, in order to say ‘this item is just like the last one’.” Unfortunately, analysts may be less than vigilant in their application of ad hoc reuse in order to complete analysis and may be inconsistent in the application of, potentially unjustified, reuse as might appear in verbatim cross-referencing of evidence components for example. Kelly and McDermid [10] observe that for safety cases an analyst may believe that certain elements of two projects are sufficiently similar to actually “cut-and-paste” parts of the original documentation and subject them only to minor review and modification. Bush and Finkelstein [4] note that “anecdotal evidence across a range of industries would seem to support the existence of this informal application of reuse.” Such reuse, as an attempt to obtain coverage over an analysis, can come at the cost of rigour [18]. However the potential risk is limited by the actual amount of reuse.

Measuring the amount of reuse in two hazard analysis case studies is exemplified in this paper. Although techniques to quantify the amount of reuse are demonstrated, no claims about the “goodness” of the reuse in terms of whether the reuse is appropriate and/or consistent are made. The first case study describes an industrial safety case and investigates reuse in practice. In contrast, the second case study is from an in-house hazard analysis using a prototype reuse support tool and indicates how tool support might change the nature of the reused components.

This paper is organised as follows. Firstly, an investigation into reuse in a hazard analysis used as part of an existing safety argument is described. Ver-
verbatim reuse\(^3\) is used as a measure to determine the frequency of actual reuse in practice. Secondly, tool supported reuse is demonstrated and the nature of the resulting reuse examined in a hazard analysis carried out by a team including the authors on a proposed system. Here the nature of the reuse has changed and a new reuse measure is needed. A measure of trivial reuse\(^4\) and how to measure it will then be discussed. Section 5 describes the results of using an edit distance algorithm to measure trivial reuse over the two cases. Section 6 considers the nature of the trivial reuse and argument clusters that are generated via the edit distance algorithm. Finally the conclusions will be presented.

2 Reuse in practice: DUST-EXPERT

For understandable reasons it is rare to find complete examples of hazard analysis in the open literature. It is therefore difficult to verify reuse practices within real world cases. However informal discussions with experts in safety-critical systems seem to indicate that reuse is common within industry based hazard analysis. These views appear to be consistent with the results of the following analysis.

\(^3\) Verbatim reuse is reuse without modifications [9, pg7].

\(^4\) Trivial reuse is a specialised form of leveraged reuse [9, pg7], reuse with modifications.
2.1 The domain

DUST-EXPERT is an application that advises on the safe design and operation of manufacturing plants subject to dust explosions. Dust explosion reduction strategies are suggested by the tool which employs a user-extensible database that captures properties of dust and construction materials [5]. Because of concerns about the consequences of wrong advice a safety case argument was developed as part of a rigorous analysis performed by a team of experts [5]. Part of this argument involves a hazard analysis utilising the HAZOP technique.

HAZOP is described as a technique of imaginative anticipation of hazards and operation problems [16, pg43]. It is a systematic technique that attempts to consider events in a system or process exhaustively. A full description of the method is not relevant to the argument of this paper and the reader is directed to [11]. Suffice to say that a key feature is the way that implicit descriptive arguments are defined, how these arguments can be structured and the extent of their reuse, particularly verbatim reuse. Figure 1 shows a fragment of the software HAZOP for DUST-EXPERT. Examples of verbatim reuse can be seen at references h 16 and h 17.

The HAZOP argument leg of the DUST-EXPERT safety case involves the identification and mitigation of hazards. This part of the analysis contains 334 individual HAZOP rows. In order to analyse the HAZOP, descriptive arguments for the HAZOP rows were transformed into a XML \(^5\) structure.

\(^5\) There is a vast array of texts on XML (eXtensible Markup Language) including [13].
that faithfully preserves the meaning of the original analysis. An example argument corresponding to the HAZOP reference h 15 in Figure 1 is shown in Figure 2.

For the descriptive arguments described in this paper the consequence elements are elicited from the Consequence/Implication column of the HAZOP and the claim elements are elicited from the Indication/Protection and Question/Recommendation columns of the HAZOP (see Figures 1 and 2). The structure of the arguments in this form is that the claims support the mitigation of the consequence. Arguments of this type are used to reduce, or mitigate, the perceived severity of hazardous consequences.
2.2 Analysis

Given this example of HAZOP in practice, it is possible to investigate verbatim reuse in the HAZOP data via the associated descriptive arguments. Arguments consist of two types: consequence mitigation arguments describe how an undesirable consequence can be mitigated by some claim(s) over an environment, for example a claim that appropriate test cases will show that a consequence will not happen; no meaning arguments arise when items in an environment cannot be considered meaningfully with HAZOP deviation keywords, for example more action, less action and no action. In this case study, there were 256 consequence mitigation arguments and 69 no meaning arguments. For this analysis only the consequence mitigation arguments have been considered relevant.

The arguments were transformed into a XML structure. Several filtering algorithms were developed to search the XML structure for interesting features and patterns over the arguments. Arguments in this case study are tree structures with nodes for consequences and support claims. The frequency of each argument in the XML structure was calculated to identify the amount of verbatim reuse. The arguments that occurred only once in the XML structure were classified as unique arguments and enumerated. Subtracting this result from the total number of arguments generates the number of non-unique arguments, i.e. those constructed with verbatim reuse.

Over the 256 consequence mitigations in this case, 203 are unique arguments while the remaining 53 occurrences are examples of verbatim reuse. Therefore, approximately 21% of the arguments have been reused in a verbatim fashion.
The reuse described in this section was applied without any explicit tool support. Hence the reuse was entirely determined by the skill of the analysis team who generated the documentation. Relying on the craft skill of the analyst may open the analysis to bias [12, pg 311]. Appropriate tool support can provide a structuring mechanism to the reuse process [18].

A prototype of such a tool has been developed [18]. Although a full description of the prototype, its application and evaluation, is outside the scope of this paper, the underlying reuse method supported by the tool will be described in the context of the second case study. The primary focus here is in the nature of the arguments that are generated. In particular, there is a concern that any tool support, especially one that encourages reuse, may bias the reuse process by implicitly supporting verbatim reuse. Hence the following descriptions will avoid in-depth analysis of the tool and focus on the nature of the resulting arguments.

3 Supported reuse: Mammography

The analysis in Section 2 and informal discussions indicate that reuse within hazard analysis is common but there is a risk that ad hoc application may render an argument unsafe. The application of tool support within the systematic process of hazard analysis may alleviate this risk. Tool support may give the analyst the ability to reflect efficiently on particular examples of reuse. In [17] a mechanism for systematic argument reuse was proposed. A prototype tool to support this mechanism has been developed by the authors and applied to the following case study. The tool provides a platform for documenting a HAZOP style hazard analysis and enables the construction and reuse of conse-
quence mitigation arguments. The motivation for the tool has been to enable the authors to investigate the application of reuse within a constructed case.

As with the study in Section 2 the reuse is applied to the arguments line-by-line with the prototype tool prompting the user with reuse candidates. A specific case is presented to illustrate and explore the approach, namely the hazard analysis of a computer-aided detection tool (CADT) for mammography.

3.1 The domain

The UK Breast Screening Program is a national service that involves a number of screening clinics, each with two or more radiologists. Initial screening tests are by mammography, where one or more X-ray films (mammograms) are taken by a radiographer. Each mammogram is then examined for evidence of abnormality by two experienced radiologists [8]. A decision is then made on whether to recall a patient for further tests because there is suspicion of cancer [19]. In the screening process it is desirable to achieve the minimum number of false positives (FPs), so that fewer women are recalled for further tests unnecessarily, and the maximum true positive (TP) rate, so that few cancers will be missed [8]. Unfortunately the radiologists’ task is a difficult one because the small number of cancers is hidden among a large number of normal cases. Also the use of two experienced radiologists, for double readings, makes this process labour intensive.

A solution that is being explored is the use of computer-based image analysis techniques to enable a single radiologist to achieve performance that is equivalent or similar to that achieved by double readings [2,8]. Computer-aided
detection systems can provide radiologists with a useful “second opinion” [24]. The case study in this section involves the introduction of a CADT as an aid in screening mammograms. When a CADT is used the radiologist initially views the mammogram and records a recall decision. The CADT then marks a digitised version of the X-ray film with “prompts” that the radiologist should examine. A final decision on a patient’s recall is then taken by the human radiologist based on the original decision and the examination of the marked-up X-ray. A summary of this process can be seen in Figure 3 (from [19]).

A system based on the model shown in Figure 3 has been investigated to identify the undesirable consequences, for example an incorrect recall decision, that may arise. The general argument for safe use involves a number of argument legs covering three main activities namely (i) human analysis of the X-ray, (ii) CADT analysis of the X-ray and (iii) the recall decision by the human based on a review of their original analysis and the CADT analysis. A hazard/consequence analysis for the system was completed by a team including the authors. This was supported by a prototype tool and reuse method.
When investigating the introduction of new technology the construction of a safety case is common. For this domain a safety case would consist of several elements including reliability analysis for the marking of the digital mammogram, the CADT performance and the consequences of human-error. However, for this paper, one element of the safety case analysis will be considered, namely hazards and consequences in the diagnosis process as defined in the overall system model (see Figure 3).

A method, with tool support, has been developed and includes steps for the identification of hazardous consequences, the definition of selection criteria to search for possible reusable arguments and the selection of reuse candidates or the definition of a new argument form. The new argument, either from a reuse candidate or a new argument template, must then be adapted to meet the specifics of the current analysis row. Finally a judgement on the nature of the hazard or consequence, i.e. whether it has been completely mitigated or not, is produced. An overview of the method can be seen in Figure 4 where the major tasks, both user and system, are identified.

Tool support aids both the gathering of hazard documentation (see Figure 5) and the selection and adaptation of reuse candidates. The tool automates the matching process between previously defined arguments to find suitable candidates for reuse either by keywords or via consequence and/or claim matching. The matching process compares arguments based on a notion of structural similarity [3,14] over argument structure and data elements.

Figure 6 shows a selection of arguments presented as candidates for possible
1. Identify hazardous consequence
2. Provide methods for selecting reuse candidates
3. Define selection criteria
4. Display reuse candidates
5. Select candidate argument or new argument
6. Display argument
7. Adapt argument
8. Classify hazard

Fig. 4. Argument reuse process for hazard mitigation and classification.

Fig. 5. Editor for collating hazard data.

reuse after a keyword search. Multiple reuse candidates are commonly identified for each query and the final selection for reuse and adaptation is left to the domain expert/tool user. As not all searches will provide an appropriate candidate for reuse the tool also allows arguments to be defined as new argument forms.

Having completed one analysis, the significant question is how tool support
affects the occurrence of reuse as identified in Section 2. There are a number of ways in which reuse may have been altered.

- The tool may produce a bias toward more verbatim reuse. Users may skip the argument adaptation step and leave the reused arguments in their initial form with the same argument structure and data.

- Increased artificial argument diversity may result when explicitly prompting the user to select and adapt arguments from previous examples. For instance more varied argument forms may be defined as users trivially adapt a reused argument. An example from the mammography case study can be seen in Figures 7 and 8. The Figure 7 argument has been reused in the Figure 8 argument by matching the consequence tag, in this case \textit{OUTPUT\_FAILURE}. The consequence data in the second argument (see Figure 8) has been adapted while the structure and claims of the argument itself are unchanged. Thus, although the structure remains the same, a unique, by data, argument has been defined.
Users may try to adapt every instance of reuse to form new argument forms. This may have the advantage of customising the fit of the arguments to the current situation but may not be cost effective due in part to time considerations. Also such extensive adaptation would result in large libraries of unique arguments thus increases the searching cost for identifying and selecting reuse candidates.

3.3 Analysis

The mammography case study contained 61 arguments where 56 arguments were unique making 5 occurrences of verbatim reuse. Thus in this case, 8% of the arguments have been reused in a verbatim fashion.

The proportion of verbatim reuse in this case is less than the amount identi-
fied in the earlier case. This may seem surprising since tool support provides easy access to copy-and-paste facilities, it may be envisioned that this would increase the amount of verbatim reuse. As an increase in verbatim reuse would signify a negative bias of the tool on the reuse process, it may be assumed that the structured reuse method is supporting good reuse habits within the hazard analysis. However, if the supported reuse process is investigated in more detail it is clear that the verbatim reuse measure is not providing an accurate figure of potentially problematic reuse.

When defining each new argument the tool provides a list of candidates for reuse that have been matched either on the basis of structural similarity or keyword matching. The user then adapts the new argument, on-the-fly, to one of these candidates. The reuse mechanism simplifies the adaptation/customisation process producing more unique argument forms and a smaller number of arguments with verbatim data. Although these adapted arguments are unique by data they may only contain trivial differences via the reuse mechanism. Unfortunately, verbatim reuse algorithms, as used in Section 2.2, are unable to identify the slight differences in the arguments. Through trivial customisation of the reused arguments the tool support may have hidden verbatim-like reuse.

4 Trivial reuse

The trivial in trivial reuse comes from the amount of customisation that has been applied to the reused case. With verbatim reuse no customisation has been applied. In trivial reuse examples only a superficial attempt has been made to customise the reused case. As with verbatim reuse, this is not prob-
Fig. 9. Trivial reuse in a consequence change example. If the reuse is appropriate and applied consistently. However, if the reuse is applied poorly then the reuse may make the argument unsafe.

Two variants of trivial reuse of interest include those (i) based on an argument consequence change and (ii) based on a single argument claim change. An example of a trivial consequence change, with verbatim claim structures, can be seen in Figure 9. In this example only the consequence description has been customised in the reused case.

The concern is that tool support may be biasing the reuse process by prompting the user to select and only trivially customise arguments from previous examples. Therefore a greater amount of artificial argument diversity may result which cannot be identified using verbatim reuse measures.

4.1 Measuring trivial reuse

As described in Section 2.2 identifying verbatim reuse is straightforward. Unfortunately, the identification of trivial reuse is not so simple. Each trivial reuse occurrence has the potential to be unique. For any sample argument it is necessary to determine the level of similarity to the other arguments in

---OUTPUT_FAILURE: Confusion in the decision making due to multiple markups
---CLAIM
---TRAINING_CLAIM: Large training set for examples fro NN
---CLAIM
---DIVERSITY_CLAIM: Use of diverse marking algorithm
---SUPPORT
---SUPPORT
---SUPPORT
---SUPPORT
---TESTING_CLAIM: Test cases meet export option and actual outcomes
---SUPPORT
---SUPPORT
---SUPPORT
---SUPPORT
---TESTING_CLAIM: Coverage by test cases

---OUTPUT_FAILURE: Additional markups of features. Possible FP
---CLAIM
---TRAINING_CLAIM: Large training set for examples fro NN
---CLAIM
---DIVERSITY_CLAIM: Use of diverse marking algorithm
---SUPPORT
---SUPPORT
---SUPPORT
---SUPPORT
---TESTING_CLAIM: Coverage by test cases

---OUTPUT_FAILURE: Confusion in the decision making due to multiple markups
---CLAIM
---TRAINING_CLAIM: Large training set for examples fro NN
---CLAIM
---DIVERSITY_CLAIM: Use of diverse marking algorithm
---SUPPORT
---SUPPORT
---SUPPORT
---SUPPORT
---TESTING_CLAIM: Coverage by test cases
the domain and in particular, to the arguments that are very similar. For complex systems this level of comparison between all the argument trees can be computationally expensive.

However, tree matching algorithms can be used to analyse the arguments as the arguments are represented in tree structures. An algorithm for approximate tree matching [21] for ordered trees has been applied. Treediff is a approximate tree matcher that runs in a high performance interpreted environment called K.

The Treediff algorithm generates the edit distance between trees from a set of input sets. Edit distance is defined as the minimum number of label modifications, node deletes and node inserts required to transform one tree to another. Edit distance is a common similarity measure, for example in plagiarism detection [23]. Sample edit distance output from the mammography case can be seen in Figure 10 showing tree pairs and their edit distance.

In this paper only edit distances 0, 1 and 2 are of interest. Edit distance 0 is a self comparison of an argument tree and provides the total number of arguments in the domain. All occurrences with edit distance 1 are examples of verbatim reuse, i.e. the only differences between the argument trees are their

7 See http://cs.nyu.edu/cs/faculty/shasha/papers/tree.html [last access 3/05/04].
8 K is list-based language integrating bulk operators, inter-process communication, and graphical user interface facilities (see http://www.kx.com/ [last access 3/05/04]). Whitney, Shasha and Apter [22] observe that “unlike most other list-based languages, K is extremely fast. For example, sorting three million records in memory takes two seconds on an IBM 990.” The speed of this environment is helpful for our larger case study (DUST-EXPERT) which contains over 296 argument trees and the tree matcher needs to perform over 60000 tree comparisons.
unique reference numbers. Edit distance 2 encompasses the unique reference number and one other change. The structure of each argument tree was artificially altered so that when edit distances were generated there were only two possible results for the extra edit distance. This was either a change in the consequence data or a change in the claim data. This is the measure of trivial reuse as only one change indicates limited argument customisation as the rest of the argument components are reused in a verbatim fashion.

5 Results

There are 61 argument trees in the mammography case and the edit distance algorithm generated 3721 comparisons. This is a large amount of data to manually process. Therefore a prototype tool was constructed to graph sections of the results so the nature of the reuse via the edit distance could be visually inspected. The tool functions by examining the results of a set edit distance. The tool plots the argument trees as nodes and draws transitions between the node if an edit distance relation is present. An example of several trees with edit distance 2 can be seen in Figure 11. All the relations and hence transitions

Fig. 10. Sample tree edit distance output from the Treediff algorithm.
Ideally all the results within an edit distance will form groups of symmetrically interlinked clusters. If all the nodes within a cluster are interlinked it implies that, apart from the difference causing the edit distance, the structure and data of all the arguments are identical. For example all the argument trees in Figure 11 share the same structure and support claims. It is only the consequence data that has changed between arguments (see Figure 12).

If all the clusters within an edit distance are interconnected it is possible to enumerate the amount of reuse as $\sum \text{nodes} - \sum \text{clusters}$. If each cluster is totally interconnected there is an assumption that one of the nodes is the original argument tree and that the others are the product of reuse and for edit distance 2 of trivial reuse. Although it is not possible to determine which
argument in particular is the original argument this is not necessary to allow
the amount of reuse to be enumerated.

5.1 Clustering results

The results of graphing edit distance 1, or the verbatim reuse, for the mam-
mosophy study can be seen in Figure 13. As noted in Section 3.3, there is
limited verbatim reuse in this case study with only 5 examples of verbatim
reuse within the 61 arguments. However this result validates the previous re-
sult of 8% verbatim reuse. Also each cluster is totally interconnected.

Figure 14 shows edit distance 2, the trivial reuse in the mammography study.
Although there are several interconnected clusters, there are four clusters that
are not totally interconnected. This is problematic in the attempt to enumerate
trivial reuse. Fortunately, on inspecting the raw data it was determined that
the trees in this example not connected at edit distance 2, within a cluster,
were examples of verbatim reuse. This was confirmed by comparing the results to those in Figure 13. As verbatim trees can be considered identical at edit distance 2, each of the problem clusters disappeared and all of the clusters become totally interconnected (see Figure 15).

Therefore the amount of trivial reuse in this case can be calculated as the sum of the nodes\(^9\) (39) minus the number of clusters (11) leaving 28 cases of trivial reuse. This makes up 46% of the arguments in this case. This is a considerable amount of potentially inappropriate reuse. As the motivation for investigating trivial reuse was the difference between the amount of verbatim reuse in the DUST-EXPERT and mammography cases, the edit distances of the DUST-EXPERT arguments were examined.

There are 256 argument trees in the DUST-EXPERT case and the edit dis-

\(^9\) Ignoring the verbatim reuse nodes.
Fig. 15. Final clustering of trivial reuse in the mammography case.

tance algorithm generated 65536 comparisons. The results of graphing the edit distance 1, the verbatim reuse, can be seen in Figure 16. In this case there are 53 examples of verbatim reuse or 21% of the total arguments. It should be noted that all the verbatim clusters are totally interconnected.

Within the DUST-EXPERT case there was a significant amount of trivial reuse identified. The edit distance algorithm identified 143 examples of trivial reuse out of the total 256 arguments. Therefore the trivial reuse is approximately 56%. On investigating the complete set of clusters, as with the mammography case, all the clusters were totally interconnected when nodes with verbatim reuse were taken into consideration.
6 Discussion

In the previous section the amount of verbatim and trivial reuse have been identified. In examples such as the two case studies it is possible that there is an illusion of coverage not borne out by reality. In this paper a numerical measure for verbatim and trivial reuse has been developed while investigating reuse within hazard analysis cases. This process can be summarised in the following six steps.

(1) Identify the explicit (descriptive) arguments (e.g. from a HAZOP).
(2) Structure the arguments (for example in XML).
(3) Generate edit distances between arguments.
(4) Identify the amount of verbatim reuse.
(5) Examine the clustering of arguments within edit distances.
(6) Identify the amount of trivial reuse.

The amount of reuse is the primary focus in the examples considered in this paper and can be completed in step 6 above. However, the clustering of the arguments within an edit distance may provide more information about the nature of the reuse in question.

While developing the trivial reuse measure two issues concerning the edit distance results and argument clustering were identified that require further investigation. Firstly, there is the issue of cluster membership. Each cluster contains all the arguments that are similar in respect to the given edit distance. However, as seen in Figure 14, there are a varying number of arguments per cluster. Having multiple members within a cluster may indicate that some level of consistent reuse was being applied over reuse candidates. In contrast, having many limited member clusters, for example clusters only consisting of two members, may indicate inconsistent reuse, i.e. reuse is present but in a more specific context. Identifying the nature of the clusters would add another level of analysis to the data provided by the edit distance algorithm. In addition, cluster membership may provide insights into the nature of argument reuse by indicating any relations between arguments in the original hazard analysis documentation.

Secondly, the nature of the arguments may be biasing the edit distance calculations. This could result in unusually large numbers of results within an edit distance. It is possible that several arguments may seem to be the product of trivial reuse but in fact any similarities are just coincidental. This could easily be the case if the argument structures are small and the argument descriptions terse. For example in Figure 9 the claim data “Coverage by test cases”
could have easily been documented as “Test cases” and used any number of times within the analysis without necessarily being part of any explicit reuse mechanism. A superficial examination of the DUST-EXPERT arguments has shown some evidence of such examples. However to get verbatim and trivial reuse matches there must be identical components in the argument trees. Although this reuse may not be intentional it still indicates reuse via the analysis process whether it is explicit, e.g. via copy-and-paste, or implicit, e.g. via the analyst’s experience.

7 Conclusions

Descriptive arguments are a standard part of the process of determining the dependability of any system. Such arguments are typically at the core of hazard analysis techniques that contribute to the construction of safety cases. Unfortunately hazard analysis is a time consuming and labour intensive process and hence reuse of analysis components is common. Reuse of analysis also results in the reuse of the associated descriptive arguments. However, inappropriate reuse can lead to misleading levels of confidence in the final analysis. This potential risk to the validity of the analysis is dependent on the nature and amount of reuse applied.

This paper has presented methods for enumerating the amount of reuse within a hazard analysis. An analysis process is described that utilises an edit distance algorithm to highlight argument clusters. Verbatim and trivial reuse can then be enumerated. This provides an indication of the potential risk to the rigour that the reuse has injected into the analysis. There has been a considerable amount of reuse in the two case studies presented. Particularly if the totals
for verbatim and trivial reuse are combined. This abundance of reuse can lead to an illusion of rigorous coverage if the reuse is not noticed. This is obviously undesirable for analysis used to support the dependability of safety-critical systems.

However, the amounts of reuse indicated by the verbatim and trivial reuse measures are only problematic if the reuse has been inappropriately applied. Although a method for structuring reuse, with associated tool support, has been defined (see Section 3.2) this is no guarantee of the “goodness” of the argument reuse. To build a good argument, the analysis needs to determine whether there is a suitable candidate argument to reuse and if so to customise it to the current context. A third case study is under development to investigate methods of providing the analyst with the best candidate to reuse as an extension to the work presented in this paper. A selection of arguments will be processed via a reuse method and analysed by domain experts for quality. If appropriate reused arguments are constructed the risk associated with verbatim and trivial reuse will be reduced.

Another issue is the cost of the reuse process. There will be costs associated with both the organisation of the raw data into argument structures and the ease of the final reuse. Also there is the overhead of identifying appropriate reuse arguments. Such issues must be balanced against any proposed benefits. However, issues of cost and benefit typically require some form of measure to allow realistic predictions to be made. A notion of confidence (and confidence in the worth of an argument) is currently being investigated as a measure to demonstrate that argument reuse will lead to improved arguments and consequently improved confidence in the arguments. This is ongoing work.
Acknowledgements

This work was supported in part by the UK EPSRC DIRC project [7], Grant GR/N13999. The authors are grateful to Adelard [1] for providing the DUST-EXPERT safety case, Eugenio Alberdi and Andrey Povyakalo who attended a field test of the prototype tool in the mammography domain and the referees and attendees of SAFECOMP 2003 who provided helpful feedback on an early version of this paper.

References


[5] Tim Clement, Ian Cottam, Peter Froome, and Claire Jones. The development of a commercial “shrink-wrapped application” to safety integrity level 2: The DUST-EXPERT™ story. In Massimo Felici, Karama Kanoun, and


[21] Jason Tsong-Li Wang, Kaizhong Zhang, Karpjoo Jeong, and Dennis Shasha. A system for approximate tree matching. IEEE Transactions on Knowledge and
