Robust Requirements Specifications for Safety-Critical Systems

A. Saeed, T. Anderson and R. de Lemos

TECHNICAL REPORT SERIES

No. 451 October, 1993
Robust Requirements Specifications for Safety-Critical Systems

A. Saeed, T. Anderson and R. de Lemos

Abstract

Experience in safety-critical systems has shown that deviations from assumed behaviour can and do cause accidents. This suggests that the development of requirements specifications for such systems should be supported with a risk analysis. In this paper we present an approach to the development of robust requirements specifications (i.e. specifications that are adequate for the risks involved), based on qualitative and quantitative analyses.
Bibliographical details

SAEED, Amer

Robust Requirements Specifications for Safety-Critical Systems
(By) A. Saeed, T. Anderson and R. de Lemos


(University of Newcastle upon Tyne, Computing Science, Technical Report Series, no. 451)

Added entries

UNIVERSITY OF NEWCASTLE UPON TYNE.
ANDERSON, Thomas
DE LEMOS, Rogerio Sergio Neves

Abstract

Experience in safety-critical systems has shown that deviations from assumed behaviour can and do cause accidents. This suggests that the development of requirements specifications for such systems should be supported with a risk analysis. In this paper we present an approach to the development of robust requirements specifications (i.e. specifications that are adequate for the risks involved), based on qualitative and quantitative analyses.

About the author

Dr. A. Saeed is a Research Associate in the DCSC, Department of Computing Science.

Mr. R. de Lemos is a Research Associate in the Departments of Computing Science and Chemical and Process Engineering.

Professor T. Anderson is a Professor in the Department of Computing Science.

Suggested keywords

REQUIREMENTS ANALYSIS
RISK ANALYSIS
SAFETY-CRITICAL

Suggested classmarks (primary classmark underlined)
Dewey (18th): 001.6425 658.47
U.D.C. 681.322.06 519.718
Robust Requirements Specifications for Safety–Critical Systems

Amer Saeed\textsuperscript{1}, Tom Anderson\textsuperscript{1,2} and Rogério de Lemos\textsuperscript{2,3}

\textsuperscript{1}BAe Dependable Computing Systems Centre
\textsuperscript{2}Department of Computing Science
\textsuperscript{3}Department of Chemical and Process Engineering
University of Newcastle upon Tyne, NE1 7RU, UK

Abstract

Experience in safety–critical systems has shown that deviations from assumed behaviour can and do cause accidents. This suggests that the development of requirements specifications for such systems should be supported with a risk analysis. In this paper we present an approach to the development of robust requirements specifications (i.e. specifications that are adequate for the risks involved), based on qualitative and quantitative analyses.

1 Introduction

During software development, the phase of requirements analysis provides the system context in which the software requirements must be considered. This is a fundamental issue for safety–critical systems because “safety” is essentially an attribute of the system rather than just software. The work in this paper enhances a methodology for the requirements analysis of safety–critical process control systems [1] by incorporating techniques for the production of robust requirements specifications, and by providing means to evaluate these specifications against the system risks. A robust requirements specification is constructed by modifying a specification to take into account violations in the assumptions upon which the specification is based, and the possibility of specifications being violated due to faults that might be introduced during later stages of software development. System risk is related to the likelihood of a system entering into a hazard state, the likelihood that the hazard will lead to an accident, and the expected potential loss associated with such an accident [2].

Robust requirements specifications are obtained by conducting qualitative and quantitative analysis of the requirements. Analysis aims to provide confidence that the level of risk is acceptable. The qualitative analysis seeks to identify those circumstances that can lead to violations of a specification, and subsequently take the system into a hazard state. The quantitative analysis attaches probabilities to the occurrence of the identified circumstances, in order to estimate the risk associated with a specification. The risk estimates provide the basis for conducting risk assessments, that compare alternative specifications and judge if the risk is acceptable. For the process of requirements analysis we adopt the approach of analysing the system from different perspectives and using different techniques [3]. This approach enables the extraction of different (and complementary) information concerning the robustness of the requirements specifications.
In summary, the enhancements proposed, in this paper, for the basic methodology are as follows. For each level of abstraction at which the analysis is performed, the assumptions are identified and recorded, and the fault analysis of the specifications is conducted, with the aim of analysing the circumstances in which the specifications are unable to maintain safe behaviour from the system. In other words, apart from checking how good the specifications are, the aim is to identify their weakness, and modify the specifications, in to make them more robust.

The methodology and its enhancements will be presented as follows. The next section describes a methodology for requirements analysis. Section 3 describes how quality can be attained, in terms of risk, by performing qualitative and quantitative analysis using viewpoints. Finally, section 4 contributes some concluding remarks.

2 A Methodology for Requirements Analysis

In this section we overview a methodology for requirements analysis; a more detailed discussion is given elsewhere [1]. The methodology consists of a framework with distinct phases of analysis, a graph that depicts the relationship between the specifications produced during the analysis, and a set of formal techniques appropriate for the issues to be analysed at each phase.

2.1 Framework for Requirements Analysis

The framework adopts the approach of separating the mission from the safety requirements during an initial phase, and then partitioning the analysis of the safety requirements into distinct phases. Each phase of analysis is focused onto a specific domain, where the identification of the relevant domains follows directly from the components (i.e. operator, plant and controller) of a general structure for safety-critical systems, and the relationship between the phases is dictated by the interactions between these components. The analysis of the phases will take into account non-standard behaviours of the entities of a domain; a basis for the analysis is provided by establishing the standard, exceptional and failure behaviours of the entities [4].

- **Conceptual Analysis.** The objective of this phase is to produce an initial statement of the aim and purpose of the system and determine those failure behaviours of the system which constitute accidents. As a product of this phase we obtain the Safety Requirements, enumerating the accidents. The accidents are the basis for separating mission from safety issues. Another activity to be performed during this phase is the identification of the modes of operation of the system; these are classes of states that group together related operational functions.

- **Safety Plant Analysis.** During this phase the plant properties relevant to the Safety Requirements, such as the physical laws and rules of operation
that govern plant behaviour and potential hazards, are identified. The outcome is the Safety Plant Specification which contains safety constraints (conditions over the physical process that are the negations of hazards modified to incorporate safety margins) and safety strategies (schemes to maintain safety constraints defined as a set of conditions, in terms of controllable factors, over the physical process).

- **Safety Interface Analysis.** The objective of this phase is to delineate the plant interface, and specify the behaviour that must be exhibited at that interface. This phase leads to the production of the Safety Interface Specification, containing the interface safety strategies (refinements of safety strategies, incorporating properties of sensors and actuators).

- **Safety Control System Analysis.** During this phase we establish a top level organization for the control system in terms of the properties of its components, and their interactions. This phase leads to the production of the Safety Control System Specification, containing the control system safety strategies (refinements of interface safety strategies incorporating the components of the control system).

### 2.2 Safety Specification Graph

The specifications produced at the different phases of the requirements analysis, are organized into a Safety Specification Graph (SSG). The structure embodied in modes of operations can be reflected in the organization of the requirements specifications by constructing a separate SSG for each mode. An SSG is a directed acyclic graph, in which the vertices represent the safety specifications (requirements specifications for safety) and the edges denotes relationships between the specifications. For a system with \( p \) accidents, the SSG consists of \( p \) component graphs. Each component graph is an evolutionary graph [5]; the evolution is related to the phases of the framework. At each phase a set of new specifications is added to the graph of the previous phases, by connecting the specifications to the terminal vertices (representing the specifications of the previous phase) of the graph.

On completion (see figure 1) the top element of each component graph is an accident (denoted by \( AC_i \)) and is related to a set of hazards (\( HZ_{ij} \)) that can lead to it. Each hazard is related to the safety constraint (\( SC_{ij} \)) that negates the hazard, and each safety constraint related to the safety strategies (\( SS_{ij,k} \)) that maintain the constraint. Then the safety strategies are related to their refinements into interface safety strategies (\( ISS_{ij,k,l} \)), and a similar relation is depicted for control system strategies (\( CSS_{ij,k,l,m} \)). When more than one strategy is related to a specification of a previous level either the strategies are exclusive and a choice has to be made in later stages of development to implement a single strategy, or the
strategies complement each other and all are needed to attain the confidence required for the risk involved.

The SSG of a system provides support in conducting a qualitative analysis and provides the basis for a systematic approach to the modification of the specifications. For the qualitative analysis, support is provided by establishing the conditions (which follow from the edges of the component graphs) that must be confirmed to ensure that the specifications maintain safe behaviour. A key concern in modification is traceability, that is the ability to trace back from a strategy to its origins and to trace forward to the strategies which are derived from the strategy. Support for traceability is provided by constructing reachability and adjacency matrices for the SSG. These matrices enable the localization of the side-effects of a modification and identification of the relationships that must be reconfirmed, thereby increasing the assurance that when changes are necessary they will be complete and consistent.

2.3 Techniques for the Framework

For the application of formal notations and techniques, the approach adopted is to employ notations in accordance with the characteristics of the system to be analysed during the different domains of analysis. Within the context of the framework, the relevant formalisms are grouped into two classes: descriptive and operational. A descriptive formalism specifies the behaviour of a domain in terms of axioms (representing system properties) over a model of the domain, whereas an operational formalism is used to model the activities and interactions between the entities of a domain. Real Time Temporal Logic and Timed History Logic are examples of descriptive formalisms, and Statecharts and Predicate—Transition Nets are examples of operational formalisms. The extent to which each class of formalism is applied in a specific domain depends on the level of abstraction at
which the domain resides. At higher levels, descriptive formalisms should be more prominent, however at the lower levels operational formalisms become increasingly relevant.

In order to describe the behaviour of systems at different levels of abstraction, we adopt an event/action model (E/A model) [1]. The main features of the E/A model are that its primitive concepts (events, actions, states and the concept of a time line) can be expressed in both descriptive and operational formalisms, and it supports both discrete and dense time structures. When employed in the framework, the models of system behaviour, constructed at the different phases, are built on top of a common foundation providing support for verification between the different levels of abstraction.

3 Quality Analysis of Safety Specifications

One important factor in determining the quality of the specifications for safety-critical systems is the risk analysis of the safety specifications; this aims to determine if the contribution of the software to the overall system risk is acceptable. In order to achieve this aim, a bridge has to be established between the risk analysis of the system and the software. Within the context of the methodology, this bridge is established through the SSG by relating the system requirements to the software requirements. To perform the risk analysis, those circumstances which can violate a specification, and cause the system to enter into a hazard state, have to be identified and their probability of occurring calculated. Once the risk is quantified we are able to judge whether the risk associated with a specification is acceptable or not (risk assessment). If not, the specification has to be modified or combined with other specifications in order to reduce the risk. As a result, we obtain a robust safety specification which is a specification that can be violated only within an acceptable risk. It should be noted that the risk analysis presented in this section does not take into account the consequences of an accident.

During the operation of the system, the occurrence of an initiating event (an event which can lead the system into a hazard state) of an accident sequence [6] distinguishes two kinds of system state: safe and unsafe state. An unsafe state is a state which could lead the system into a hazard state in the absence of corrective action and in the absence of subsequent initiating events. If a state is not an unsafe state then it is said to be safe. These definitions ensure that a hazard cannot occur subsequent to a safe state if no initiating event occurs. In terms of the requirements specifications, the concept of initiating event refers to those circumstances which can lead to the violation of a safety specification.

The quality analysis of the requirements, in each domain of analysis, is performed from two different perspectives: qualitative and quantitative. The purpose of the qualitative analysis is to identify those circumstances which can violate a specification, and analyse the impact of these violations upon the safety of the system. These circumstances are related to the
violation of assumptions upon which a specification is based and to the violation of certain conditions of a specification. The quantitative analysis complements the qualitative analysis by attaching occurrence probabilities to these circumstances. In order to ensure that essential system behaviour is not precluded, the restrictions that a safety specification will impose on the mission must also be considered.

3.1. Qualitative Risk Analysis

At each level of abstraction, analysis is conducted over safety specifications (descriptions of safe behaviour at the level) and assumptions (properties assumed at the level). In the proposed approach the qualitative analysis is conducted in two stages; firstly we perform the preliminary analysis and secondly the vulnerability analysis of the safety specifications.

3.1.1 Preliminary Analysis

In this paper, we consider the preliminary analysis to be the analysis that must be conducted prior to the risk analysis. This analysis will involve confirming that the specifications at a particular layer of the SSG comply with those of the layer that precede it, and that the specifications in a layer are consistent. The relationships that must be confirmed, to ensure compliance between the layers, follow from the edges of the SSG. Demonstrating compliance between the layers involves employing both verification (formal analysis) and validation (informal analysis) techniques. The hazards are validated against the accidents and the safety constraints are verified against the negation of the hazards. Subsequently the strategies are verified against the specifications of the previous layer. At each layer, any assumptions required to confirm the relationships, depicted by the edges of the SSG, are recorded. As an example of the relations that must be verified, we examine the edge (from the SSG in figure 1) that connects the safety constraint $SC_{i,1}$ to the safety strategy $SS_{i,1,1}$. Let us suppose that the strategy is based upon assumption $A$ (which represents a property of the physical process); the relationship to be confirmed is then:

$$ A \land SS_{i,1,1} \Rightarrow SC_{i,1} $$

A result of the preliminary analysis is that the circumstances under which safe behaviour is maintained, are clearly scoped and organised in accordance with their contribution to each phase of the analysis. This activity ensures that the knowledge gained during the development and validation/verification of the safety specifications can be applied effectively during the risk analysis.

3.1.2 Vulnerability Analysis

After performing the preliminary analysis of the safety specifications, the qualitative risk analysis consists of performing the vulnerability analysis of the specifications which probes the safety specification, and associated assumptions, to identify the circumstances under which the specification is unable to maintain safe behaviour, i.e. the violation of a
specification. Once these circumstances are identified, the safety specifications can be modified to become more robust against possible violations. An initial step in the vulnerability analysis is to negate the relationships obtained during the preliminary analysis and to identify the system states which can lead to the violation of the specification when the above circumstances occur.

For the relationship $f_1$, the logical assertion is negated and those plant states (PS) which can lead to the violation of SC$_{i,1}$ are identified:

$$\neg SC_{i,1} \Rightarrow (\neg SS_{i,1,1} \land PS) \lor (\neg A \land PS) \quad f_2$$

For this relationship, which is associated with the plant level, the subsequent vulnerability analysis of the safety strategy SS$_{i,1,1}$ will identify those conditions that can lead to the violation of SC$_{i,1}$.

Although logical formulae are useful in obtaining a high-level view of the relationship between the specifications and assumptions, such formulae provide limited support for a failure analysis. A suitable representation, for such analysis, is one which supports the identification of possible failure behaviours that can lead to the identified hazardous states. In this paper, to perform the vulnerability analysis of the safety specifications, we employ fault tree analysis (FTA) [7] which has been used extensively in the analysis of system safety and more recently in the analysis of software safety [8]. A key feature of fault tree analysis that makes it suitable for the analysis to be conducted here, is that the analysis is restricted to the identification of system components and conditions that lead to one particular undesired system state.

To construct a fault tree for the relationship $f_2$, the initial step is to identify the undesired state, in this case the negation of the safety constraint SC$_{i,1}$, and then to determine the set of possible causes which can lead to the undesired state (refer to figure 2). For the logical formula $f_2$, we identify the violation of the assumption and the violation of the safety strategy SS$_{i,1,1}$. The latter has to be further refined in order to identify its primary events.

Qualitative risk analysis provides a basis for obtaining more robust safety specifications which will lead to a risk abatement of the overall system. In the approach adopted, the analysis is performed by employing both formal analysis and fault tree analysis in order to determine the weaknesses of the safety specifications. Once these weaknesses are identified the safety specifications can be modified to incorporate mechanisms which aim to reduce their vulnerability.

### 3.2 Quantitative Risk Analysis

In this section we discuss how a quantitative analysis complements the qualitative analysis by introducing a measurement of confidence in the quality of the safety specifications.
While the latter identifies circumstances which can lead to the violation of the specifications, the former associates probabilities to these circumstances.

Although the qualitative approach strives to achieve total assurance for the safety specifications, there are three basic limitations which indicate that this aim may not be realised. A first limitation stems from the process of capturing user requirements: some faults introduced during the requirements stage may not be removed during the verification process, nor can it be guaranteed that all such faults will be removed during validation. A second limitation arises from observing past experience in the utilization of formal techniques, which shows that a formal verification may itself contain faults [9]. The third limitation is related to the confidence that can be placed on the assumptions upon which a specification is based. From these limitations we infer that even after performing the qualitative analysis we are still faced with uncertainties concerning the quality of the safety specifications, hence the necessity to quantify the uncertainties in order to obtain a level of confidence in the quality of the safety specifications. In other words, the aim is to obtain an early prediction of the contribution of the software to the risk of the system.

To associate occurrence probabilities to those circumstances which can lead to the violation of the specification, such as plant states and violation of assumptions, might not be a difficult task. On the other hand, associating occurrence probabilities to the violation of certain conditions that depend on a software implementation is more problematic because during the requirements phase of software development sufficient design and implementation information is not yet available. Instead of estimating the probability of a condition to be violated, target probabilities demanded from the higher level safety specifications such as hazards, can be used. However, once a specification is sufficiently detailed, currently
available techniques which attempt to make early predictions of the software reliability can be used, such as: metrics [10], product-in-a-process [11], and execution of the specifications [12].

After conducting the quantitative risk analysis, the last stage of the quality (risk) analysis is to perform the risk assessment of the safety specifications. This is a judgement based on the estimated risk which provides guidance for high level decisions, usually associated with the process of requirements analysis. The results obtained from the quantitative analysis should be considered as a relative measurement of how effective a given strategy is in reducing the risk of a hazard, compared to the results obtained for alternative strategies. Hence it is most useful in determining which strategy or combination of strategies is most suitable for the risks involved (the choice of a strategy might also be influenced by constraints imposed by the implementation, e.g. availability of sensors and actuators with the required properties). Also, if the utilization of more than one strategy is required, this preliminary risk analysis facilitates the search for a suitable combination of the available strategies in order to avoid common mode failures.

3.3 Mission and Safety Analysis

The primary aim of the quality analysis presented in this paper is to reduce the system risk. However, it is usually impossible to maintain a complete dichotomy between the mission and safety aspects, and it would be futile to impose safety requirements which were so stringent that the system could not satisfy its mission. To complement the risk analysis, the impact of the safety specifications on the mission of the system must also be considered. Such an analysis involves relating the different safety specifications to the mission requirements that can be affected by them. If analysis of the mission requirements follows the framework described in section 2.1, leading to the construction of a Mission Specification Graph (MSG), a comparison between the safety specifications and mission specifications (requirements specifications for the mission) is made possible. During the development of a robust safety specification it would be possible to identify the mission specifications that may be influenced by inspecting the variables that are restricted by the safety specifications and relating these to the mission specifications at the same level of abstraction. Once the relevant mission specifications have been identified an informal analysis of the restrictions that the safety specification imposes on the mission can be conducted. An example of such an analysis is presented elsewhere [13].

4 Conclusions

This paper describes a systematic approach for the quality analysis of the requirements specifications, in the context of a methodology for the requirements analysis of safety-critical systems. The approach is based on an analysis, at each level of abstraction, of the risks introduced by the various decisions (that are based on assumptions) made in
establishing the requirements. The quality analysis follows the structure of a traditional safety study and incorporates both qualitative and quantitative techniques.

The results of the risk analysis provide estimates of the risk associated with a specification and predictions of the software's contribution to the system risk. These results are used to guide the construction of robust requirements specifications, increase the confidence (assurance) that the level of risk is acceptable and provide the basis for a feasibility study. The approach to risk analysis brings the safety studies of the system and software closer together and delineates the contribution of the software to the overall system risk. Some aspects of the approach have been applied to a train set example [14].

Acknowledgements

The authors would like to acknowledge the financial support of BAe (DCSC) and the SERC (UK) SCHEMA project.

References


