INTRODUCTION AND OBJECTIVES

As mentioned in Chapter I, a service’s dependability must be justified in a quantitative way and proved through extensive on-field testing and fault injection, verification and validation techniques, simulation, source-code instrumentation, monitoring, and debugging. An exhaustive treatment of all these techniques falls outside the scope of this book, nevertheless the author feels important to include in this text an analysis of the effect on dependability of some of the methods that have been introduced in previous chapters.

RELIABILITY ANALYSIS OF THE TIRAN DISTRIBUTED VOTING MECHANISM

As mentioned in Chapter IX, a number of applications are structured in such a way as to be straightforwardly embedded in a fault-tolerance architecture based on redundancy and consensus. Applications belonging to this class are, for instance, parallel airborne and spaceborne applications. The TIRAN Distributed Voting
mechanism provides application-level support to these applications. This section 
analyses the effect on reliability of the enhancements to the TIRAN Distributed 
Voting mechanism described in the mentioned chapter, that is, management of 
spares, dealt with in Sect. 2.1, and fault-specific recovery strategies supported by 
the \( \alpha \)-count feature, analyzed in Sect. 2.2.

**Using ARIEL to Manage Spares**

This section analyses the influence of one of the features offered by ARIEL—its 
ability to manage spare modules in \( N \)-modular redundant systems—that has been 
introduced and discussed in Chapter VI and Chapter IX.

Reliability can be greatly improved by this technique. Let us first consider the 
following equation:

\[
R^{(0)}(t) = 3R(t)^2 - 2R(t)^3,
\]

i.e., the equation expressing the reliability of a TMR system with no spares, \( R(t) \) 
being the reliability of a single, non-replicated (simplex) component. Equation (1) 
can be derived for instance via Markovian reliability modeling under the assump-
tion of independence between the occurrence of faults (Johnson, 1989).

With the same technique and under the same hypothesis it is possible to show 
that, even in the case of non-perfect error detection coverage, this equation can be 
considerably improved by adding one spare. This is the equation resulting from 
the Markov model in Fig. 1, expressed as a function of error recovery coverage 
(\( C \), defined as the probability associated with the process of identifying the failed 
module out of those available and being able to switch in the spare (Johnson, 1989)) 
and time (\( t \)):

\[
R^{(1)}(C, t) = (-3C^2 + 6C) \times [R(t)(1-R(t))]^2 + R^{(0)}(t).
\]

Appendix A gives some mathematical details on Eq. (2).

Adding more spares obviously implies further improving reliability. In general, 
for any \( N \geq 3 \), it is possible to consider a class of monotonically increasing reliability 
functions,

\[
(R^{(M)}(C, t))_{M \geq 0}.
\]
Related Content

Extracting Ontology Properties from the Web-Tables
Song-il Cha and Z. M. Ma (2012). *International Journal of Systems and Service-Oriented Engineering* (pp. 64-77).
[www.igi-global.com/article/extracting-ontology-properties-from-the-web-tables/79239?camid=4v1a](www.igi-global.com/article/extracting-ontology-properties-from-the-web-tables/79239?camid=4v1a)

Fault Tree Analysis (FTA) via Binary Diagram Decision (BDD) for Information Systems Design
[www.igi-global.com/chapter/fault-tree-analysis-fta-via/75753?camid=4v1a](www.igi-global.com/chapter/fault-tree-analysis-fta-via/75753?camid=4v1a)
Eliciting Security Requirements for an Information System using Asset Flows and Processor Deployment
www.igi-global.com/article/elicitingsecurity-requirements-for-an-information-system-using-asset-flows-and-processor-deployment/83634?camid=4v1a

Development of Automated Systems using Proved B Patterns
www.igi-global.com/chapter/development-automated-systems-using-proved/76954?camid=4v1a