# Chapter 16
# Resilience and Fault Tolerance in Electrical Engineering

**Niels P. Zussblatt, Alexander A. Ganin, Sabrina Larkin, Lance Fiondella, and Igor Linkov**

**Abstract** As a result of the increased importance of engineered electrical systems to modern civilization, it is necessary to design systems that sustain ideal levels of performance despite the potential for internal faults and external attacks. Designing systems that exhibit resilience, also known as fault tolerance, is the primary method by which optimal performance is preserved despite adverse conditions. This paper is a review of a variety of computational and electromechanical fault tolerance techniques from the literature in order to evaluate the state of the art and identify potential areas for improvement. Our findings suggest that the existing literature has only focused on a limited number of resilience challenges, and that no single resilience-enhancing solution, either hardware- or software-based, is capable of addressing all of the major types of possible faults. Further, we classify the papers using the resilience matrix, which combines four resilience phases put forth by the National Academy of Sciences and four Network Centric Warfare domains. We identify the matrix components insufficiently addressed: particularly, we have found no relevant literature on the cognitive and social domains. Even within the parts of the resilience matrix that have received attention in the literature to date, we observe that there is relatively less emphasis placed on the adaptation of the computational



N.P. Zussblatt
Environmental Laboratory, US Army Engineer Research & Development Center, Concord, MA, USA

Department of Chemical Engineering, University of California, Santa Barbara, CA, USA

A.A. Ganin
Department of Systems and Information Engineering, University of Virginia, Charlottesville, VA, USA

S. Larkin • I. Linkov (✉)
Environmental Laboratory, US Army Engineer Research & Development Center, Concord, MA, USA
e-mail: Igor.Linkov@usace.army.mil

L. Fiondella
Department of Electrical & Computer Engineering, University of Massachusetts, Dartmouth, MA, USA



I. Linkov, J.M. Palma-Oliveira (eds.), *Resilience and Risk*, NATO Science for Peace and Security Series C: Environmental Security,
DOI 10.1007/978-94-024-1123-2_16

and electromechanical systems so that a repeated fault will not incur significant disruption in subsequent occurrences. Therefore, based on this review, we find that while significant and sustained attention has been dedicated to enhance the resilience of engineering electrical systems, substantial work remains to fully address resilience challenges that instill confidence in our ability to engineer resilient systems.

**Keywords** Resilience • Electrical engineering • Fault tolerance • Risk • Safety by design

## 16.1 Introduction

Computers and engineered electrical systems have become ubiquitous in the modern world. Moreover, society's increased dependence on computers in both critical and everyday applications necessitates that continuous operation of computational resources be preserved despite the presence of external threats. Of particular concern are the Critical Infrastructure Sectors identified by the United States Department of Homeland Security, which include chemical facilities, financial service institutions, and transportation systems, (Presidential Policy Directive n.d.; Department of Homeland Security n.d.) many of which are heavily reliant on computerized or electrical systems. Despite increased attention to threats directed against computerized systems, these critical systems are not fully protected from compromises that could result from accidents or deliberate malevolent acts. For example, in recent years, the aviation industry has suffered a series of debilitating incidents resulting from failure of computerized systems. In 2014, an air traffic control system in the southwestern United States suffered a failure when a single aircraft with a complex flight path overwhelmed the memory of the computers (Scott and Menn 2014). On another occasion, Delta Air Lines experienced a collapse of its flight management and passenger reservation systems when a piece of electrical equipment failed and the automatic backup systems failed to engage (Stelloh and Gutierrez 2016). As indicated by these examples, the design of computer systems that are capable of maintaining operation through faults is of the utmost importance.

Resilience is defined as the ability to anticipate and adapt to changing conditions as well as to withstand and recover rapidly from disruptions (Presidential Policy Directive n.d.). As a result of these varied requirements of resilience, four stages of resilience goals have been described: anticipate, withstand, recover, and evolve (Bodeau and Graubart 2011). Alternatively, these goals have been labeled by the National Academy of Science (NAS) as plan/prepare, absorb, recover, and adapt (Resilience 2012). A truly resilient system will have mechanisms in place to address each of these goals. Additionally, a complex system will have several distinct domains whose preparedness against adverse operation will need to be considered. The common method of dividing a system into domains, originally described by the

U.S. Department of Defense Network Centric Warfare (NCW) is to consider its physical, information, cognitive, and social components (Alberts 2002). Here, the physical domain refers to the physical components and capabilities of the system, the information domain refers to the information and data within the physical domain, the cognitive domain refers to the use of other domains for decision-making processes, and the social domain refers to the robustness of the organizational structure and the ability of the system to communicate information (Alberts 2002).

As a result of these distinct perspectives on the problem of evaluating resilience, a unified method to characterize the preparedness of a system was desired. To meet this need, Linkov, et al. introduced the concept of the "resilience matrix" where the four stages of resilience defined by the NAS and the four NCW resilience domains are placed on separate axes to generate a 4x4 matrix (Linkov et al. 2013). A system with robust safeguards where all elements of the resultant matrix have been addressed can be considered to be highly resilient. In contrast, a lack of attention to one or more elements in the resilience matrix would indicate a point of vulnerability, which may be used to direct attention to improve the security of the system as a whole. Recent publications have evaluated the state of resilience in a variety of fields, including cyber security (DiMase et al. 2015) and the energy sector (Roege et al. 2014). The concept of resilience is not foreign to computer scientists, who often know it by the term "fault tolerance." Regardless of the term used, resilience is an important attribute that must be implemented in electrical engineering systems and circuits so that they can provide stable service even in the face of errors. The importance of designing computers to exhibit resilience was first described by Avižienis in 1967, (Avižienis 1967) but to the best of our knowledge there does not exist a review of the field according to modern multi-criteria resilience principles. In this work, we examine the literature on resilience in electromechanical and computational systems according to the concept of the resilience matrix. In addition to identifying how the field of computer science has addressed resiliency (and failed to do so with respect to parts of the resilience matrix), the scope of robustness challenges examined within the field and the general methods employed are also examined.



## 16.2 Materials and Methods

In this work, we reviewed 61 papers that discussed resilience or fault tolerance within a computational system or from an electromechanical engineering standpoint and evaluated the degree to which strategies to assess and enhance resilience were articulated. Papers were identified by Google Scholar search in October 2014 for "electrical engineering", "resilience", "robustness", and "fault tolerance," and then sorted manually for relevance to the field of resilience in electrical engineering. Relevance was determined by interpretation of the abstract and the main text, with effort made to ensure selection of papers for review was performed without bias. The review was not strictly exhaustive and it is possible that certain relevant papers

were left out of its scope for reasons such as a different terminology, low ranking in the search results, and our misinterpretation of their abstracts. In addition, certain older works (such as Avižienis 1967) have been included as well if they were highly cited or identified as having made a lasting impact in later work.

The final count of 61 papers published between 1967 and 2014 was primarily from within computer science journals and conference proceedings, although some were found in other disciplines such as aerospace engineering (Chen and Trachtenberg 1991; Alena et al. 2008, 2011), where robustness of computer systems is also considered a priority. While most of the papers focused on the robustness of computer architecture (e.g., logic gates) and internal memory or data to faults or corruption, some of the works extended their scope to tolerating faults of internal mechanical components or mechanical systems controlled by the computer systems as well (Pradeep et al. 1988; Maciejewski 1990). Hence, we consider this to be a review of fault tolerance in both computational and electromechanical systems.

Once identified, papers were divided according to the type of problem they sought to guard against and the general methodology of their proposed solution(s). In each paper, the type of failure that was to be guarded against was noted. To make this work accessible to a general audience, the types of problems were grouped into a small number of distinct categories, including manufacturing variation and external malicious attacks. Following this, the method of the solution proposed in each work was identified, whether it required hardware alterations, changes to software or internal coding, or a combination of both. Finally, each paper was evaluated according to the extent to which the solutions they proposed addressed the National Academy of Science and Network Centric Warfare components of resilience. The complete list of papers examined and their assignments according to the resilience problem(s) addressed, method of solution (hardware, software, or combination), and the resilience phases and domains that were considered can be found in Tables 16.1, 16.2, and 16.3, respectively, which are placed after the conclusion of this chapter.

## 16.3 Results and Discussion

To appreciate the kinds of resilience of concern to the computational and electromechanical engineering research communities, the papers examined for this work were organized according to the general types of problems they considered. The percentage of papers addressing each general type of problem is given in Fig. 16.1, and the specific assignment of each paper examined to problem types is provided in Table 16.1. Of the types of failure examined, single event upsets, which are the flipping of a single bit of computer memory, primarily induced by cosmic rays or other radiation (Ziegler and Lanford 1979), was the most common problem that resilience considerations sought to guard against. Following single event upsets, failure of circuits or mechanical components such as logic gates stuck in the “on” or “off” state and voltage decreases that could result in failure to convert a bit state were the next most common impediment to achieving a resilient computer system. Another

t1.1 **Table 16.1** Resilience problems and solution types in reviewed publications

| t1.2 Author/s (Year) | Identified problem | | | | | | |
|---|---|---|---|---|---|---|---|
| | Voltage errors or droops | Timing errors | Manufact-uring variation | Failure of circuits and mechanical components | Single event upsets | Human-machine inter-action faults | Malicious attacks |
| t1.3 Agarwal et al. (2007) | | x | | x | | | |
| t1.4 Alena et al. (2008) | | | | x | | x | |
| t1.5 Alena et al. (2011) | | | | x | x | | |
| t1.6 Avižienis (1967) | x | | | x | x | | |
| t1.7 Avižienis (1997) | | | x | x | x | x | x |
| t1.8 Banerjee et al. (2007) | x | | x | | | | |
| t1.9 Bartlett and Spainhower (2004) | | | | x | | x | |
| t1.10 Bau et al. (2009) | | | x | x | x | | |
| t1.11 Bowman et al. (2009a) | x | x | | | | | |
| t1.12 Bowman et al. (2009b) | x | x | | | | | |
| t1.13 Bowman et al. (2011) | x | x | | | | | |
| t1.14 Breuer (2005) | | x | x | | | | |
| t1.15 Brunina et al. (2012) | | | x | x | x | | |
| t1.16 Chakrapani et al. (2006) | | | x | | | | |
| t1.17 Chen and Trachtenberg (1991) | | | | x | x | | |
| t1.18 Chippa et al. (2010) | x | | | x | | | |
| t1.19 Dolev and Haviv (2006) | x | | | | x | | |
| t1.20 Fang et al. (2014) | | | | | x | | |
| t1.21 Galster et al. (1998) | | | | | x | | |
| t1.22 Gaubatz et al. (2008) | | | | | | | x |
| t1.23 Hayes and Polian (2007) | | | | | x | | |
| t1.24 Hazucha et al. (2003) | | | | | x | | |
| t1.25 Hsieh et al. (2008) | | | x | | | | |

(continued)

**Table 16.1** (continued)

| Author/s (Year) | Identified problem | | | | | | |
|---|---|---|---|---|---|---|---|
| | Voltage errors or droops | Timing errors | Manufact-uring variation | Failure of circuits and mechanical components | Single event upsets | Human-machine inter-action faults | Malicious attacks |
| Huang et al. (2000) | | | | x | | | |
| Kang and Kim (2007) | x | x | x | | | | |
| Leem et al. (2010) | x | | x | x | x | | |
| Li and Yeung (2006) | | | | | | x | |
| Lima et al. (2001) | | | | | x | | |
| Liu and Whitaker (1992) | | | | | x | | |
| Maciejewski (1990) | | | | x | | | |
| Merlin et al. (2014) | x | | | | | | |
| Meshram and Belorkar (2011) | | | x | x | x | | |
| Mitra et al. (2005) | | | | | x | | |
| Mitra et al. (2007) | | | | | x | | |
| Mukherjee et al. (2002) | x | | | | x | | |
| Nassif et al. (2010) | | | x | x | x | | |
| Nickel (2001) | x | x | | | x | | |
| Nicolaidis (1999) | x | | | | x | | |
| Normand (1996) | | | | | x | | |
| Oh et al. (2002a) | x | x | | | | | |
| Oh et al. (2002b) | | | | | x | | |
| Pradeep et al. (1988) | | | | x | | | |
| Reddi et al. (2012) | x | | x | x | x | | |
| Rennels (1978) | | | x | x | | | |
| Richardeau et al. (2002) | x | | | | | | |
| Roche and Gasiot (2005) | | | | | x | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| t1.49 | Rockett (1992) | | | | | x | | |
| t1.50 | Rotenberg (1999) | x | x | | | | | |
| t1.51 | Sanda et al. (2008) | | | | | x | | |
| t1.52 | Saxena et al. (2000) | x | | x | x | x | | |
| t1.53 | Seshia et al. (2007) | x | | x | x | x | | |
| t1.54 | Touba and McCluskey (1997) | | | | x | | | |
| t1.55 | Tschanz et al. (2009) | x | x | | x | | | |
| t1.56 | Ullah and Sterpone (2014) | | | | x | x | | |
| t1.57 | Vishwanath and Nagappan (2010) | | | | x | | | |
| t1.58 | Visinsky et al. (1994) | | | | x | | x | |
| t1.59 | Walters et al. (2011) | | | | | x | | |
| t1.60 | Wong (2006) | | | | | x | | |
| t1.61 | Yoshimoto et al. (2012) | | | | | x | | |
| t1.62 | Yu et al. (2000) | | | | x | | | |
| t1.63 | Zhang et al. (2006) | | | | | x | | |

t2.1 **Table 16.2** Reviewed publications sorted by use of "fault tolerance" or "resilience" and proposed
t2.2 solution type(s)

| t2.3 t2.4 t2.5 Author/s (Year) | Fault tolerance (FT) or Resilience (R) | Proposed solution | |
|---|---|---|---|
| | | Hardware | Software/Calculations |
| t2.6 Agarwal et al. (2007) | FT | x | |
| t2.7 Alena et al. (2008) | FT | x | x |
| t2.8 Alena et al. (2011) | FT | x | |
| t2.9 Avižienis (1967) | FT | x | x |
| t2.10 Avižienis (1997)) | FT | x | x |
| t2.11 Banerjee et al. (2007) | R | x | x |
| t2.12 Bartlett and Spainhower (2004) | FT | x | x |
| t2.13 Bau et al. (2009) | R | x | |
| t2.14 Bowman et al. (2009a) | R | x | |
| t2.15 Bowman et al. (2009b) | R | x | |
| t2.16 Bowman et al. (2011) | R | x | |
| t2.17 Breuer (2005) | R | | x |
| t2.18 Brunina et al. (2012) | R | x | x |
| t2.19 Chakrapani et al. (2006) | R | x | |
| t2.20 Chen and Trachtenberg (1991) | FT | | x |
| t2.21 Chippa et al. (2010) | R | x | |
| t2.22 Dolev and Haviv (2006) | FT | | x |
| t2.23 Fang et al. (2014) | R | | x |
| t2.24 Galster et al. (1998) | R | x | |
| t2.25 Gaubatz et al. (2008) | R | | x |
| t2.26 Hayes and Polian (2007) | R | | x |
| t2.27 Hazucha et al. (2003) | R | x | |
| t2.28 Hsieh et al. (2008) | R | | x |
| t2.29 Huang et al. (2000) | FT | x | x |
| t2.30 Kang and Kim (2007) | R | x | |
| t2.31 Leem et al. (2010) | R | x | x |
| t2.32 Li and Yeung (2006) | FT | | x |
| t2.33 Lima et al. (2001) | R | | x |
| t2.34 Liu and Whitaker (1992) | R | x | |
| t2.35 Maciejewski (1990) | FT | x | |
| t2.36 Merlin et al. (2014) | FT | x | |
| t2.37 Meshram and Belorkar (2011) | FT | x | |
| t2.38 Mitra et al. (2005) | R | x | x |
| t2.39 Mitra et al. (2007) | R | x | |
| t2.40 Mukherjee et al. (2002) | FT | | x |
| t2.41 Nassif et al. (2010) | R | x | x |
| t2.42 Nickel (2001) | FT | x | x |
| t2.43 Nicolaidis (1999) | FT | x | x |
| t2.44 Normand (1996) | R | x | x |
| t2.45 Oh et al. (2002a) | R | | x |

(continued)

**Table 16.2** (continued)

| Author/s (Year) | Fault tolerance (FT) or Resilience (R) | Proposed solution | | |
|---|---|---|---|---|
| | | Hardware | Software/Calculations | |
| Oh et al. (2002b) | FT | | x | t2.46 |
| Pradeep et al. (1988) | FT | x | | t2.47 |
| Reddi et al. (2012) | R | x | x | t2.48 |
| Rennels (1978) | FT | x | x | t2.49 |
| Richardeau et al. (2002) | FT | x | | t2.50 |
| Roche and Gasiot (2005) | R | x | | t2.51 |
| Rockett (1992) | R | x | | t2.52 |
| Rotenberg (1999) | FT | | x | t2.53 |
| Sanda et al. (2008) | R | x | x | t2.54 |
| Saxena et al. (2000) | FT | x | x | t2.55 |
| Seshia et al. (2007) | R | | x | t2.56 |
| Touba and McCluskey (1997) | FT | | x | t2.57 |
| Tschanz et al. (2009) | R | x | | t2.58 |
| Ullah and Sterpone (2014) | FT | x | | t2.59 |
| Vishwanath and Nagappan (2010) | R | x | | t2.60 t2.61 |
| Visinsky et al. (1994) | FT | | x | t2.62 |
| Walters et al. (2011) | FT | | x | t2.63 |
| Wong (2006) | R | | x | t2.64 |
| Yoshimoto et al. (2012) | R | x | x | t2.65 |
| Yu et al. (2000) | FT | x | x | t2.66 |
| Zhang et al. (2006) | R | x | x | t2.67 |



commonly identified problem is variation in manufacturing that causes components to behave differently, including failures at different rates which has become a more serious problem as the physical size of circuits have become smaller and manufacturing tolerances harder to meet (Borkar 2005). A variety of additional types of failures such as timing errors, human-machine interaction faults, and malicious attacks were discussed in a smaller number of papers. In many cases, the solutions proposed by a paper are applicable to a variety of problems, as is often indicated by the authors of each work. For instance, an error-correcting technique based on dynamic bit steering can be applied to address errors which are due to radiation-induced bit flips, and permanent hardware defects which may result from initial manufacturing variation or failure at a later time (Brunina et al. 2012). This broader applicability of many resilience techniques is indicated by the summation of the percentages in Fig. 16.1 being greater than 100%. Specifically, it was found that the 61 papers proposed solutions to 112 problems, indicating that the average proposed method to improve resilience was determined to be applicable to approximately two major types of fault-inducing errors.

t3.1 **Table 16.3** Types of resilience strategies in reviewed publications

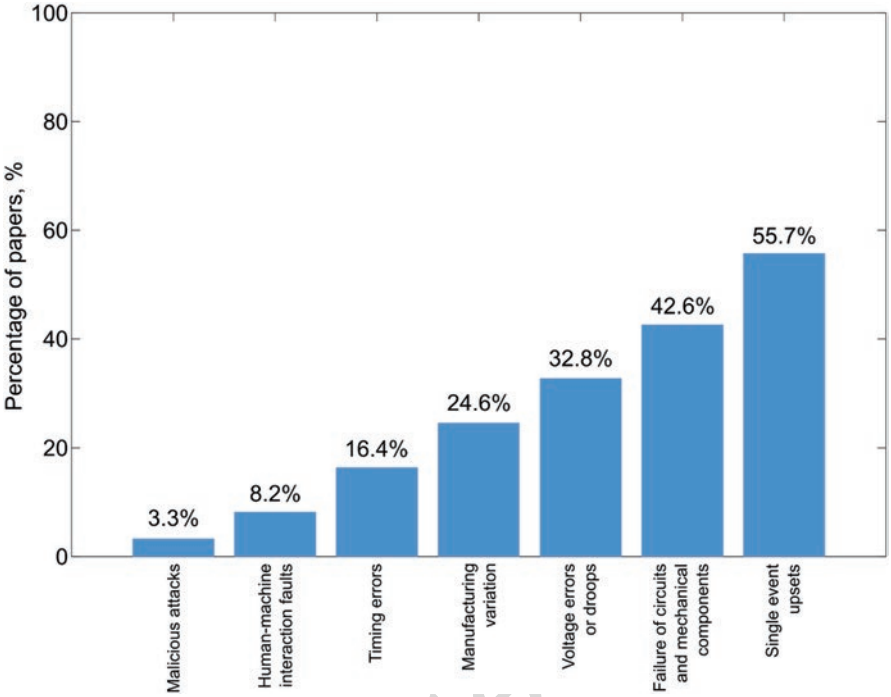
| | National Academy of Sciences (NAS) phases | | | | Network Centric Warfare (NCW) domains | | | |
|---|---|---|---|---|---|---|---|---|
| t3.2 Author/s (Year) | Plan | Adsorb | Recover | Adapt | Physical | Information | Cognitive | Social |
| t3.3 Agarwal et al. (2007) | x | | | | x | | | |
| t3.4 Alena et al. (2008) | x | | | x | x | x | | |
| t3.5 Alena et al. (2011) | x | | x | | | x | | |
| t3.6 Avižienis (1967) | x | x | | x | x | x | | |
| t3.7 Avižienis (1997) | x | x | x | | x | | | |
| t3.8 Banerjee et al. (2007) | x | x | | | x | x | | |
| t3.9 Bartlett and Spainhower (2004) | | x | x | x | x | x | | |
| t3.10 Bau et al. (2009) | x | | x | | x | | | |
| t3.11 Bowman et al. (2009a) | x | | x | | | x | | |
| t3.12 Bowman et al. (2009b) | x | x | x | | | x | | |
| t3.13 Bowman et al. 2011) | x | | x | | x | x | | |
| t3.14 Breuer (2005) | x | x | | | x | x | | |
| t3.15 Brunina et al. (2012) | x | | x | x | x | x | | |
| t3.16 Chakrapani et al. (2006) | | x | | | x | | | |
| t3.17 Chen and Trachtenberg (1991) | | x | | | x | x | | |
| t3.18 Chippa et al. (2010) | x | | | | x | x | | |
| t3.19 Dolev and Haviv (2006) | x | x | | | x | x | | |
| t3.20 Fang et al. (2014) | x | x | x | | x | | | |
| t3.21 Galster et al. (1998) | x | x | | | x | | | |
| t3.22 Gaubatz et al. (2008) | x | | x | x | | x | | |
| t3.23 Hayes and Polian (2007) | x | x | x | | x | x | | |
| t3.24 Hazucha et al. (2003) | x | x | | | x | | | |
| t3.25 Hsieh et al. (2008) | | x | | x | | x | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| t3.26 | Huang et al. (2000) | x | x | x | x | x | | | |
| t3.27 | Kang and Kim (2007) | x | x | | | x | | | |
| t3.28 | Leem et al. (2010) | x | x | | | x | x | | |
| t3.29 | Li and Yeung (2006) | x | x | | x | x | x | | |
| t3.30 | Lima et al. (2001) | | x | | x | x | | | |
| t3.31 | (Liu and Whitaker 1992 | | x | | | x | | | |
| t3.32 | Maciejewski (1990) | x | x | x | x | x | | | |
| t3.33 | Merlin et al. (2014) | x | | x | | x | | | |
| t3.34 | Meshram and Belorkar (2011) | x | x | | | x | | | |
| t3.35 | Mitra et al. (2005) | x | | x | | x | x | | |
| t3.36 | Mitra et al. (2007) | x | | x | | x | | | |
| t3.37 | Mukherjee et al. (2002) | | x | | | x | x | | |
| t3.38 | Nassif et al. (2010) | x | x | | | x | x | | |
| t3.39 | Nickel (2001) | | x | x | | x | | | |
| t3.40 | Nicolaidis (1999) | x | x | | | x | x | | |
| t3.41 | Normand (1996) | x | | | | x | | | |
| t3.42 | Oh et al. (2002a) | | x | | | x | x | | |
| t3.43 | Oh et al. (2002b) | x | x | | | | x | | |
| t3.44 | Pradeep et al. (1988) | x | | x | x | x | | | |
| t3.45 | Reddi et al. (2012) | x | x | x | x | | x | | |
| t3.46 | Rennels (1978) | x | x | x | x | x | | | |
| t3.47 | Richardeau et al. (2002) | x | x | x | x | x | | | |
| t3.48 | Roche and Gasiot (2005) | x | x | | | x | | | |
| t3.49 | Rockett (1992) | x | x | x | | x | x | | |
| t3.50 | Rotenberg (1999) | x | x | | | x | x | | |

(continued)

**Table 16.3** (continued)

| Author/s (Year) | National Academy of Sciences (NAS) phases | | | | Network Centric Warfare (NCW) domains | | | |
|---|---|---|---|---|---|---|---|---|
| | Plan | Adsorb | Recover | Adapt | Physical | Information | Cognitive | Social |
| Sanda et al. (2008) | x | x | x | | x | x | | |
| Saxena et al. (2000) | x | x | x | x | x | | | |
| Seshia et al. (2007) | x | | x | | | x | | |
| Touba and McCluskey (1997) | x | x | x | | x | | | |
| Tschanz et al. (2009) | x | x | x | | | x | | |
| Ullah and Sterpone (2014) | x | | x | | x | | | |
| Vishwanath and Nagappan (2010) | x | x | x | x | x | x | | |
| Visinsky et al. (1994) | x | | x | x | x | | | |
| Walters et al. (2011) | x | | x | x | x | x | | |
| Wong (2006) | x | | x | | x | x | | |
| Yoshimoto et al. (2012) | x | x | | | | x | | |
| Yu et al. (2000) | x | x | | | x | x | | |
| Zhang et al. (2006) | x | | x | | x | x | | |




this figure will be printed in b/w

**Fig. 16.1** Resilience problems identified in the literature, ranked according to the number of papers. The sum of percentages (184%) is greater than 100% because many papers address multiple resilience issues



There has been increasing interest in resilience of electromechanical systems in recent years. Figure 16.2 shows the number of papers in the review, sorted by year of publication. Further, the papers are also sorted by whether they use the terminology "fault tolerance" or "resilience," with the number of papers using each of these terms indicated by different colors. The complete listing of papers by year of publication and terminology used is available in Table 16.2. For many years, the number of publications included in the review was zero. There is however, an unmistakable trend toward increased numbers of papers with a maximum observed in 2007. Although a decline in interest may be suggested by the fewer papers in subsequent years, it is important to note that the number for 2014 only reflects those papers published during part of the year, and that the number of papers included from 2010–2014 (Alena et al. 2011) outpaced the number from 2000–2004 (Avižienis 1967), indicating a continued rise in attention to resilience concepts.

During the course of the review, it was noted that the publications referred to their attention to the resiliency of computer systems as either "fault tolerance" or "resilience." When papers were sorted by which term used, it was evident that in early years "fault tolerance" was the normal descriptor, while the term "resilience" became more common after 2000. This shift in terminology may be a reflection of

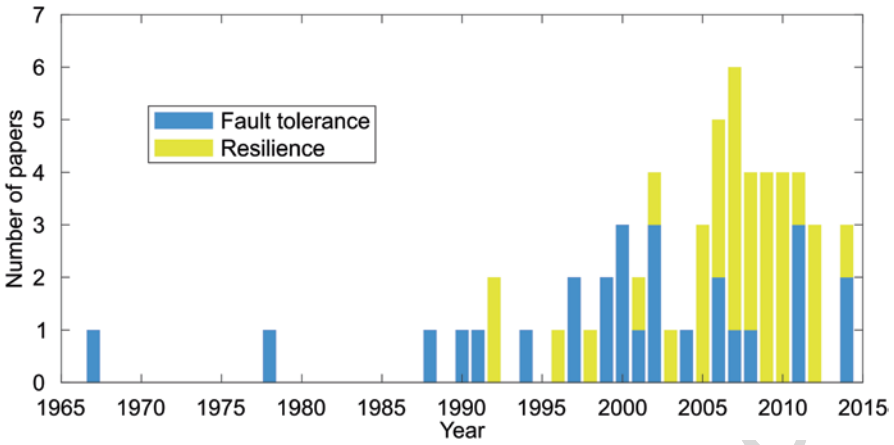


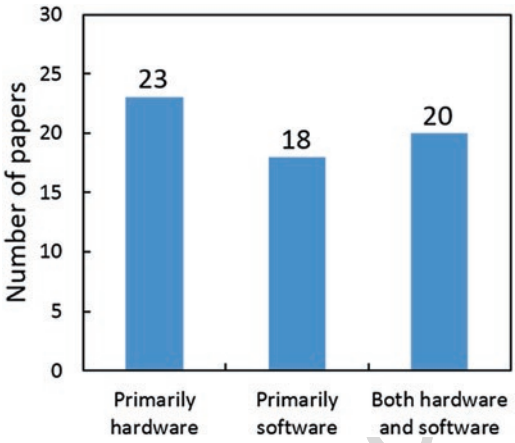

**Fig. 16.2** Grouping of the papers by the year of publication. Blue bars represent papers focusing on fault tolerance, while yellow bars show papers discussing resilience



increased consideration for system disruptions that are not purely mechanical failures. For example, both the earliest works in the review give attention to failure of circuits and mechanical components, (Avižienis 1967; Rennels 1978) whereas the majority of papers discussing human-machine interaction faults were published after 2000 (Alena et al. 2008; Bartlett and Spainhower 2004; Li and Yeung 2006). From this simple analysis, it is clear that as computer and electromechanical systems have increased in importance, there has been an increase in attention given to their resilient- and fault-free-operation. The transition to the term "resilience" suggests that the field is examining the robustness of these systems to failure modes not initially considered when discussing "fault tolerance."

The papers examined were also indexed according to the primary methodology they proposed as a solution for the particular resilience problem identified. For the sake of simplicity, methodologies were divided according to their focus on hardware, software, or a combination of both. The results summarized in Fig. 16.3 suggest that there is no preferred methodology within the computer and electrical engineering communities, with roughly equal attention dedicated to hardware-based methods (Merlin et al. 2014), software-based methods such as error-correcting codes or multi-threading of computational operations (Rotenberg 1999; Mukherjee et al. 2002), and mixed methods combining elements of both hardware and software (Brunina et al. 2012). A full summary of the high-level assignments of the resilience solution proposed by each paper to hardware, software, or a combined methodology is provided in Table 16.2.

Finally, papers were sorted by how they addressed the NAS Phases (Resilience 2012) and NCW Domains (Alberts 2002) of resilience. The resilience procedures that make up the four phases of resilience are called planning, absorbing, recovering, and adapting. For example, a resilience plan may focus attention on the prevention of operational failures. Alternatively, mechanisms could be incorporated to preserve




this figure will be printed in b/w

**Fig. 16.3** Number of papers by methodology of proposed resilience solution(s)

the function of circuits or other components despite a particular impairment. Still other options include focusing on correcting the problem at hand in hopes of returning to ideal state of function as soon as possible, or providing a mechanism whereby it was possible to learn more about the failure that had occurred in order to prevent similar occurrences in the future. These are the four resilience procedures of planning, absorbing, recovering, and adapting, respectively.

In addition to evaluating papers according to the NAS Phases of resilience, the solutions they presented were also indexed according to the four domains of resilience: physical, informational, cognitive, and social. Actions to improve a system's resilience can be categorized in terms of these domains. A physical solution could be a change to the design of the circuit; an informational solution could involve the way the circuit communicates information to those programming it; a cognitive solution could involve the engineers processing the physical and informational outputs to better inform future design decisions, and a social solution would be a way to share the learned cognitive conclusions.

Complete classification for all papers in the review into NAS Phases and NCW Domains is included in Table 16.3, while a summary of the results, presented in the form of a resilience matrix (Linkov et al. 2013), is shown in Fig. 16.4. In this figure, the percentage of papers in the review addressing each element of the matrix generated by permutations of the NAS Phases (columns) and NCW Domains (rows) is indicated. In the review, most papers either evaluated resilience strictly in the physical or informational domains; only 26 of the 61 articles considered both types of solutions in their resilience efforts. Additionally, no paper took either the cognitive or social domains into consideration. Ultimately, more solutions for faults in electrical engineering systems and circuits could potentially be determined if more of these resilience domains were taken into account.

There is a clear bias toward the planning and absorbing phases within the two resilience domains addressed in the articles reviewed. Further, a clear reduction in the percentage of papers addressing later stages of resilience (recover and adapt)

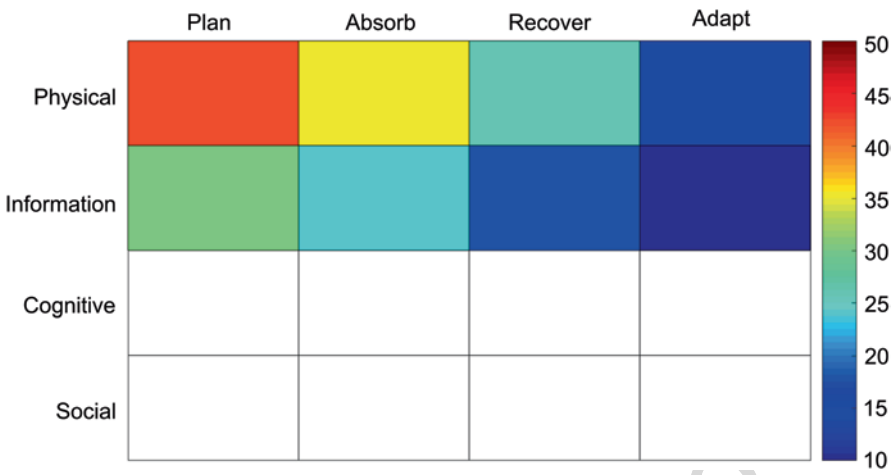



**Fig. 16.4** Number of papers reviewed by classification of the phases and domains of resilience addressed



234 can be observed. Specifically, every reviewed paper included fault prevention (i.e.
235 planning) or absorption as a stage in their resilience strategy, and fully 22 of the 61
236 reviewed papers included only these two phases. One particular exception, Bartlett
237 and Spainhower, focused strictly on what should happen after an error is detected:
238 damage should be minimized, and a quick recovery plan initiated (Bartlett and
239 Spainhower 2004). While this paper touched upon both the recovery and adaptation
240 phases, most of the remaining papers that addressed the latter two phases only
241 considered one or the other, with a greater number including recovery as an aspect
242 of their resilience solution. In total, less than one-third (specifically 18/61) papers
243 attempt to consider adaptation – how to improve system function and performance
244 to prevent similar errors from occurring in the future. One recent example of
245 adaptation as a resiliency phase is included in Reddi et al. (2012) where software is
246 used to control the frequency of operation of individual circuits. This control is
247 linked to measures of the operation of the circuits, and so it is possible for the system
248 to predict optimal settings for individual components and adjust these to maximize
249 error-free performance despite manufacturing differences, voltage droops, and tran-
250 sient single-event upsets. Ultimately, resilience efforts will be most effective if all
251 four tenets are considered. However, most electrical engineering and circuit-related
252 resilience studies only considered on average two of the four categories.
253 Resilience in electromechanical systems focused primarily on the plan and
254 absorb phases may be a consequence of how the field of how fault tolerance and
255 resilience in this field developed. Many of the earliest works addressed resilience
256 problems via hardware-only or hardware-and-software methods (Pradeep et al.
257 1988; Maciejewski 1990; Rennels 1978; Liu and Whitaker 1992; Rockett 1992).
258 The types of resilience issues that primarily hardware-based solutions may be suited
259 to address appear to be more suited to planning and absorbing problems, rather than

adapting to them. Examples of such issues and their proposed hardware-based solutions include introducing additional joints that allow a computer-controlled robot to retain function during partial mechanical failure (Pradeep et al. 1988), and improved diodes that better absorb radiation-induced upsets (Galster et al. 1998). By contrast, software-based resilience solutions, such as error-correcting codes (Nickel 2001) or software that can produce qualitatively correct outputs despite errors in inputs or calculations (Li and Yeung 2006; Wong 2006), are somewhat more modern and have the ability to address neglected resilience phases of recovery and adaptation. However, it is important to note that hardware-based approaches can be used to address recovery or adaptation. For instance, Tschanz, *et al.* describe "tunable replica circuits" that can dynamically respond to timing and voltage errors (Tschanz et al. 2009). Addressing all phases of resilience is an important goal for future work within the computer and electrical engineering fields, and it will be exciting to see new software-based solutions presented that are suited to recovery and adaptation. However, it will be necessary for the field to begin to address the cognitive and social domains of resilience, which they appear to not have done to date. Once these portions of the resilience matrix are explicitly considered, computer systems will be much more robust against faults, errors, and unexpected disruptions.

## 16.4 Conclusions

Creation of resilient electrical systems (circuits) is challenged by the fact that throughout the years engineers and scientists have focused on the processing efficiency of computing systems and their optimization, while resilience calls for adaptive and flexible structure of the system. Current practices lead to unnecessarily rigid streamlined algorithms of how the circuits process information. The general approach to combat errors occurring in these systems have been focused on error detection and correction. Recovery is accomplished by recalculation of the originally miscalculated values, while adaptation is difficult in many cases because the system is not flexible. This tendency is highlighted in how the phases of resilience rank by the number of the reviewed papers addressing them: plan (51/61 papers), absorb (41/61), recover (33/61), and adapt (18/61). Moreover, the inflexibility of the systems defined the predictable and probabilistic nature of the fault events considered.

Although significant attention was given to the physical (51/61 papers) and information domains (36/61), we found no papers on the cognitive and social domains of the resilience matrix. Understandably, this may be caused by the fact that electrical engineering as a field of research lies in the physical and, to a lesser extent, in the information domains. On the other hand, our review also shows the need for an integrated design and deployment practices which encompass all four domains.

The main contribution of this work is a comprehensive review of the electrical engineering research papers on resilience and fault tolerance published to date and

a new approach to classify resilience research papers with the resilience matrix. The next steps may include the development of guidelines and recommendations for resilient design of electrical circuitry and a methodology for resilience quantification in the field of electrical engineering.

## Further Suggested Readings


Agarwal M, Paul BC, Zhang M, Mitra S (2007) Circuit failure prediction and its application to transistor aging. In 25th IEEE VLSI test symposium (VTS'07); pp 277–286

Alberts DS (2002) Information age transformation. getting to a 21st century military. Revised.; Washington, DC

Alena R, Ellis SR, Hieronymus J, Maclise D (2008) Wireless avionics and human interfaces for inflatable spacecraft. In IEEE aerospace conference proceedings

Alena R, Gilstrap R, Baldwin J, Stone T, Wilson P (2011) Fault tolerance in ZigBee wireless sensor networks. In IEEE aerospace conference proceedings; pp 1–15

Avižienis A (1967) Design of fault-tolerant computers. In Proceedings of the November 14–16, 1967, Fall joint computer conference; pp 733–743

Avižienis A (1997) Toward systematic design of fault-tolerant systems. *Computer (Long Beach Calif) 30*(4):51–58

Banerjee N, Karakonstantis G, Roy K (2007) Process variation tolerant low power DCT architecture. In Proceedings – Design, Automation and Test in Europe, DATE'07; Vol. 7, pp 630–635

Bartlett W, Spainhower L (2004) Commercial fault tolerance: a tale of two systems. IEEE Trans Dependable Secur Comput 1(1):87–96

Bau J, Hankins R, Jacobson Q, Mitra S, Saha B, Adl-Tabatabai A-R (2009) Error resilient system architecture (ERSA) for probabilistic applications. In Proceedings of the international symposium on low power electronics and design

Bodeau DJ, Graubart R (2011) MITRE cyber resiliency engineering framework, MTR110237; Bedford

Borkar S (2005) Designing reliable systems from unreliable components: the challenges of transistor variability and degradation. IEEE Micro 25(6):10–16

Bowman KA, Tschanz JW, Kim NS, Lee JC, Wilkerson CB, Lu S-LL, Karnik T, De VK (2009a) Energy-efficient and metastability-immune resilient circuits for dynamic variation tolerance. IEEE J Solid-State Circuits 44(1):49–63

Bowman K, Tschanz J, Wilkerson C, Lu S-L, Karnik T, De V, Borkar S (2009b) Circuit techniques for dynamic variation tolerance. In Design Automation Conference, 2009. DAC '09. 46th ACM/IEEE; pp 4–7

Bowman KA, Tschanz JW, Lu S-LL, Aseron PA, Khellah MM, Raychowdhury A, Geuskens BM, Tokunaga C, Wilkerson CB, Karnik T, De VK (2011) A 45 Nm resilient microprocessor core for dynamic variation tolerance. IEEE J Solid-State Circuits 46(1):194–208

Breuer MA (2005) Multi-media applications and imprecise computation. In Proceedings – DSD'2005: 8th euromicro conference on digital system design – architectures, methods and tools; pp 2–7

Brunina D, Lai CP, Liu D, Garg AS, Bergman K (2012) Resilient optically connected memory systems using dynamic bit-steering [Invited]. J Opt Commun Netw 4(11):B151

Chakrapani LN, Akgul BES, Cheemalavagu S, Korkmaz P, Palem KV, Seshasayee B (2006) Ultra-efficient (Embedded) SOC architectures based on probabilistic CMOS (PCMOS) technology. In Proceedings – design, automation and test in Europe, DATE'06; pp 1110–1115

Chen M, Trachtenberg EA (1991) Permutation codes for the state assignment of fault tolerant sequential machines. In Proceedings Of The 10th digital avionics systems conference; pp 85–90

Chippa VK, Mohapatra D, Raghunathan A, Roy K, Chakradhar ST (2010) Scalable effort hardware design: exploiting algorithmic resilience for energy efficiency. In Design Automation Conference (DAC), 2010 47th ACM/IEEE; pp 555–560

Department of Homeland Security. Critical Infrastructure Sectors (n.d.) https://www.dhs.gov/critical-infrastructure-sectors

DiMase D, Collier ZA, Heffner K, Linkov I (2015) Systems engineering framework for cyber physical security and resilience. Environ Syst Decis 35(2):291–300

Disaster Resilience: A National Imperative (2012) The National Academies Press: Washington, DC

Dolev S, Haviv YA (2006) Self-stabilizing microprocessor: analyzing and overcoming soft errors. IEEE Trans Comput 55(4):385–399

Fang L, Yamagata Y, Oiwa Y (2014) Evaluation of a resilience embedded system using probabilistic model-checking. In Electronic proceedings in theoretical computer science; Vol. 150, pp 35–49

Galster N, Frecker M, Carroll E, Vobecky J, Hazdra P (1998) Application-specific fast-recovery diodes: design and performance. In Power Conversion April 1998 Proceedings; pp 1–14

Gaubatz G, Savaş E, Sunar B (2008) Sequential circuit design for embedded cryptographic applications resilient to adversarial faults. IEEE Trans Comput 57(1):126–138

Hayes JP, Polian I, Becker B (2007) An analysis framework for transient-error tolerance. In Proceedings of the IEEE VLSI test symposium; pp 249–255

Hazucha, P.; Kamikl, T.; Walstra, S.; Bloechell, B.; Tschanzl, J.; Maiz, J.; Soumyanath, K.; Demer, G.; Narendra, S.; De, V.; Borkar, S. (2003) Measurements and analysis of SER tolerant latch in a 90 nm Dual-Vt CMOS Process. In IEEE 2003 custon integrated circuits conference; pp 617–620

Hsieh T-Y, Lee K-J, Breuer MA (2008) An error rate based test methodology to support error-tolerance. IEEE Trans Reliab 57(1):204–214

Huang W-J, Saxena N, McCluskey EJ (2000) Reliable LZ data compressor on reconfigurable coprocessors; pp 249–258

Kang K, Kim K, Roy K (2007) Variation resilient low-power circuit design methodology using on-chip phase locked loop. In ACM/IEEE Design Automation Conference; pp 934–939

Leem L, Cho H, Bau J, Jacobson QA, Mitra S (2010) ERSA: error resilient system architecture for probabilistic applications. In Design, Automation Test in Europe Conference Exhibition, 2010; pp 1560–1565

Li X, Yeung D (2006) Exploiting soft computing for increased fault tolerance. In Workshop on architectural support for Gigascale integration

Lima F, Rezgui S, Carro L, Velazco R, Reis R (2001) On the use of VHDL simulation and emulation to derive error rates. In 6th European Conference on Radiation and Its Effects on Components and Systems; pp 253–260

Linkov I, Eisenberg DA, Bates ME, Chang D, Convertino M, Allen JH, Flynn SE, Seager TP (2013) Measurable resilience for actionable policy. Environ Sci Technol 47:10108–10110

Liu N, Whitaker S (1992) Low power SEU immune CMOS memory circuits. IEEE Trans Nucl Sci 39(6):1679–1684

Maciejewski AA (1990). Fault tolerant properties of kinematically redundant manipulators. In IEEE Conference on Robotics and Automation; pp 638–642

Merlin MMC, Green TC, Mitcheson PD, Trainer DR, Critchley R, Crookes W, Hassan F (2014) The alternate arm converter: a new hybrid multilevel converter with DC-fault blocking capability. IEEE Trans Power Deliv 29(1):310–317

Meshram SS, Belorkar UA (2011) Design approach for fault tolerance in FPGA architecture. Int J VLSI Des Commun Syst 2(1):87–95

Mitra S, Seifert N, Zhang M, Shi Q, Kim KS (2005) Robust system design with built-in soft-error resilience. Computer (Long. Beach. Calif). No. February, 43–52

Mitra S, Zhang M, Seifert N, Mak TM, Kim KS (2007) Built-in soft error resilience for robust system design. In IEEE International Conference on Integrated Circuit Design and Technology; 2007; pp 1–6

Mukherjee SS, Kontz M, Reinhardt SK (2002) Detailed Design and Evaluation of Redundant Multithreading Alternatives. In 29th Annual International Symposium on Computer Architecture; pp 99–110

Nassif SR, Mehta N, Cao Y (2010) A resilience roadmap. In Design, Automation &Test in Europe Conference & Exhibition; pp 1011–1016

Nickel JB, Somani AK (2001) REESE: a method of soft error detection in microprocessors. In Proceedings of the International Conference on Dependable Systems and Networks; pp 401–410

Nicolaidis M (1999) Time redundancy based soft-error tolerance to rescue nanometer technologies. In Proceedings 17th IEEE VLSI Test Symposium

Normand E (1996) Single event upset at ground level. IEEE Trans Nucl Sci 43(6):2742–2750

Oh N, Mitra S, McCluskey EJ (2002a) ED4I: error detection by diverse data and duplicated instructions. IEEE Trans Comput 51(2):180–199

Oh N, Shirvani PP, McCluskey EJ (2002b) Error detection by duplicated instructions in super-scalar processors. IEEE Trans Reliab 51(1):63–75

Pradeep AK, Yoder PJ, Mukundan R, Schilling RJ (1988) Crippled motion in Robots. IEEE Trans Aerosp Electron Syst 24(1):2–13

Presidential Policy Directive – Critical Infrastructure Security and Resilience. https://www.whitehouse.gov/the-press-office/2013/02/12/presidential-policy-directive-critical-infrastructure--security-and-resil. n.d.

Reddi VJ, Pan DZ, Nassif SR, Bowman KA (2012) Robust and resilient designs from the bottom-up: technology, CAD, circuit, and system issues. In Proceedings of the Asia and South Pacific Design Automation Conference, ASP-DAC; pp 7–16

Rennels DA (1978) Architectures for fault-tolerant spacecraft computers. Proc IEEE 66(10):1255–1268

Richardeau F, Baudesson P, Meynard TA (2002) Failures-tolerance and remedial strategies of a PWM multicell inverter. IEEE Trans Power Electron 17(6):905–912

Roche P, Gasiot G (2005) Impacts of front-end and middle-end process modifications on terrestrial soft error rate. IEEE Trans Device Mater Reliab 5(3):382–395

Rockett LR (1992) Simulated SEU hardened scaled CMOS SRAM cell design using gated resistors. IEEE Trans Nucl Sci 39(5):1532–1541

Roege PE, Collier ZA, Mancillas J, McDonagh JA, Linkov I (2014) Metrics for energy resilience. Energy Policy 72:249–256

Rotenberg E (1999). AR-SMT: A microarchitectural approach to fault tolerance in microprocessors. In Twenty-Ninth Annual International Symposium on Fault-Tolerant Computing; pp 84–91

Sanda PN, Kellington JW, Kudva P, Kalla R, McBeth RB, Ackaret J, Lockwood R, Schumann J, Jones CR (2008) Soft-error resilience of the IBM POWER6 processor. IBM J Res Dev 52(3):275–284

Saxena N, Fernandez-Gomez S, Huang W, Mlra S, Ya S-Y, Mccluskey EJ (2000) Dependable computing and online testing in adaptive and configurable systems. IEEE Des Test Comput 17:29–41

Scott A, Menn J (2014) Exclusive: air traffic system failure caused by computer memory shortage. Reuters

Seshia SA, LiW, Mitra S (2007) Verification-guided soft error resilience. In Proceedings of the conference on design, automation and test in Europe; pp 1442–1447

Stelloh T, Gutierrez G (2016) Georgia power company disputes “Outage” behind delta’s system failure. NBC News

Touba NA, McCluskey EJ (1997) Logic synthesis of multilevel circuits with concurrent error detection. IEEE Trans Comput Des Integr Circuits Syst 16(7):783–789

Tschanz J, Bowman K, Wilkerson C, Lu S-L, Karnik T (2009) Resilient circuits – enabling energy-efficient performance and reliability. In Proceedings of the 2009 International Conference on Computer-Aided Design - ICCAD ‘09; pp 71–73

Ullah A, Sterpone L (2014) Recovery time and fault tolerance improvement for circuits mapped on SRAM-based FPGAs. J Electron Test Theory Appl 30(4):425–442
Vishwanath KV, Nagappan N (2010) Characterizing cloud computing hardware reliability. In Proceedings of the 1st ACM Symposium on Cloud Computing - SoCC '10; pp 193–203
Visinsky ML, Cavallaro JR, Walker ID (1994) Expert system framework for fault detection and fault tolerance in robotics. Comput Electr Eng 20(5):421–435
Walters JP, Kost R, Singh K, Suh J, Crago SP (2011) Software-based fault tolerance for the maestro many-core processor. In IEEE aerospace conference proceedings; pp 1–12
Wong V, Horowitz M (2006) Soft error resilience of probabilistic inference applications. In IEEE workshop on silicon errors in logic; pp 1–4
Yoshimoto S, Amashita T, Okumura S, Nii K, Yoshimoto M, Kawaguchi H (2012) Bit-error and soft-error resilient 7T/14T SRAM with 150-Nm FD-SOI Process. IEICE Trans Fundam Electron Commun Comput Sci E95–A (8), 1359–1365
Yu S-Y, Saxena N, McCluskey EJ (2000) An ACS Robotic control algorithm with fault tolerant capabilities. In IEEE Symposium on FPGAs for custom computing machines, Proceedings pp 175–184
Zhang M, Mitra S, Member S, Mak TM, Seifert N, Wang NJ, Shi Q, Kim KS, Shanbhag NR, Patel SJ (2006) Sequential element design with built-in soft error resilience. IEEE Trans Very Large Scale Integr Syst 14(12):1368–1378
Ziegler JF, Lanford WA (1979) Effect of cosmic rays on computer memories. *Science (80-. ). 206* (4420), 776–788