PESARO 2019

The Ninth International Conference on Performance, Safety and Robustness in Complex Systems and Applications

ISBN: 978-1-61208-698-9

March 24 - 28, 2019

Valencia, Spain

PESARO 2019 Editors

Mohammad Rajabali Nejad, UTWente, Netherlands

PESARO 2019

Forward

The Ninth International Conference on Performance, Safety and Robustness in Complex Systems and Applications (PESARO 2019), held between March 24, 2019 and March 28, 2019 in Valencia, Spain, continued a series of events dedicated to fundamentals, techniques and experiments to specify, design, and deploy systems and applications under given constraints on performance, safety and robustness.

There is a relation between organizational, design and operational complexity of organization and systems and the degree of robustness and safety under given performance metrics. More complex systems and applications might not be necessarily more profitable, but are less robust. There are trade-offs involved in designing and deploying distributed systems. Some designing technologies have a positive influence on safety and robustness, even operational performance is not optimized. Under constantly changing system infrastructure and user behaviors and needs, there is a challenge in designing complex systems and applications with a required level of performance, safety and robustness.

We welcomed academic, research and industry contributions. The conference had the following tracks:

- Methodologies, techniques and algorithms
- Applications and services

We take here the opportunity to warmly thank all the members of the PESARO 2019 technical program committee, as well as all the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to PESARO 2019. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

We also thank the members of the PESARO 2019 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that PESARO 2019 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the areas related to performance, safety and robustness in complex systems. We also hope that Valencia, Spain provided a pleasant environment during the conference and everyone saved some time to enjoy the historic charm of the city.

PESARO 2019 Chairs

PESARO Steering Committee

Wolfgang Leister, Norsk Regnesentral (Norwegian Computing Center), Norway
Mohammad Rajabali Nejad, University of Twente, the Netherlands
Omar Smadi, Iowa State University, USA
Yulei Wu, University of Exeter, UK

PESARO Industry/Research Advisory Committee
John Favaro, INTECS, Italy
Jean-Pierre Seifert, TU Berlin & FhG SIT Darmstadt, Germany
Roger Rivett, Jaguar Land Rover, UK

PESARO 2019 Special Tracks Chair
Lorena Parra, IMIDRA, Spain // Universitat Politecnica de Valencia, Spain

PESARO 2019
Committee

PESARO Steering Committee
Wolfgang Leister, Norsk Regnesentral (Norwegian Computing Center), Norway
Mohammad Rajabali Nejad, University of Twente, the Netherlands
Omar Smadi, Iowa State University, USA
Yulei Wu, University of Exeter, UK

PESARO Industry/Research Advisory Committee
John Favaro, INTECS, Italy
Jean-Pierre Seifert, TU Berlin & FhG SIT Darmstadt, Germany
Roger Rivett, Jaguar Land Rover, UK

PESARO 2019 Special Tracks Chair
Lorena Parra, IMIDRA, Spain // Universitat Politecnica de Valencia, Spain

PESARO 2019 Technical Program Committee

Morteza Biglari-Abhari, University of Auckland, New Zealand
Quentin Bramas, University of Strasbourg, France
Hind Castel, Telecom SudParis | Institut Mines Telecom, France
Andrea Ceccarelli, University of Florence, Italy
Salimur Choudhury, Lakehead University, Canada
Dieter Claeys, Ghent University, Belgium
Frank Coolen, Durham University, UK
Daniele Di Pompeo, University of L'Aquila, Italy
Simon Eismann, University of Würzburg, Germany
Faten Fakhfakh, University of Sfax, Tunisia
John Favaro, INTECS, Italy
Francesco Flammini, UMUC Europe, Italy
Víctor Flores Fonseca, Universidad Católica del Norte, Chile
Simos Gerasimou, University of York, UK
Denis Gingras, Université de Sherbrooke, Canada
Teresa Gomes, University of Coimbra, Portugal
Lorena González Manzano, University Carlos III of Madrid, Spain
Denis Hatebur, University Duisburg-Essen, Germany
Mohamed Hibti, EDF R&D, France
C. Michael Holloway, Safety-Critical Avionics Systems Branch | NASA Langley Research Center, Hampton, Virginia, USA
Christoph-Alexander Holst, Institute Industrial IT / OWL University of Applied Sciences, Germany
Rémy Houssin, University of Strasbourg, France

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Link Prediction in Network by a Modified Mutual Information Model

Yuling Yang
College of Information Systems and Management
National University of Defense Technology
Changsha, China
yulingyoung@yeah.net

Guangquan Cheng
College of Information Systems and Management
National University of Defense Technology
Changsha, China
yyl9505@126.com

Kuihua Huang
College of Information Systems and Management
National University of Defense Technology
Changsha, China
yang_ma_cn@163.com

Zhong Liu
College of Information Systems and Management
National University of Defense Technology
Changsha, China
phillipliu@263.net

***Abstract*— Link prediction in a network refers to predicting the possibility of connection between two nodes. A traditional method, Local Baysian Method, based on nodes' common neighbors, achieves high prediction accuracy as well as has low computing complexity. However, the method ignores the Mutual Information between the common neighbors. So, we take mutual information model into consideration, while the algorithm has high computing complexity. In this paper, we will modify the model and make it more efficient.**

Keywords- link prediction; Mutual Information; baysian network.

I. INTRODUCTION

Real-world systems can be modeled by complex networks in most cases. A typical network is composed of nodes and links, where nodes represent different individuals in the system, and links represent relationships between individuals. If there is a connection between two nodes, edges are joined, and vice versa. Two nodes connected by an edge are considered neighbors in the network. The nervous system of nematode worms, for example, can be thought of as a network of neurons connected by synapses. The American aviation network can be seen as a network formed by airports connected with each other through existing direct flight routes. Similarly, there are computer networks, social networks, logistics networks and so on.

Link prediction in the network refers to predicting the possibility of connection between two nodes that have not yet generated edges or whose connection has not yet been discovered [1] through known network structure and other information, which is actually a process of data mining. For example, A is a friend of B's, B is a friend of C's, then there may be a connection between A and C. The traditional link prediction method is to use Markov chain or machine learning to predict nodes using nodes' attributes. The prediction accuracy of this method is high, but its computational complexity and non-universal parameters limit its uses. Another method is mainly based on similarity and likelihood analysis, which uses the network structure characteristics. Among various similarity-based indices, Common Neighbors (CN) is undoubtedly the precursor with low computing complexity. This paper mainly adopts this method.

II. PROBLEM DESCRIPTION

Considering an undirected network $G(V, E)$, where V is the set of nodes and E is the set of links. Multiple links and self-connections are not allowed. Denote by U the universal set containing all $|V| \cdot (|V| - 1)/2$ possible links, where $|V|$ denotes the number of elements in set V, and $|E|$ denotes the number of edges in set E. Then, the set of nonexistent links is $U - E$. We assume there are some missing links (or the links that will appear in the future) in the set $U - E$, and the task of link prediction is to find out these links. Generally, we do not know where the missing or future links are, otherwise we do not need to do prediction. Therefore, to test the algorithm's accuracy, the observed links, E, is randomly divided into two parts: the training set, E^T, which is treated as known information, while the probe set (i.e., validation subset), E^P, is used for testing and no information in this set is allowed to be used for prediction. Clearly, $E^T \cup E^P = E$ and $E^T \cap E^P = \emptyset$. Considering a simple undirected network denoted as $G(V, E)$, the given network can be represented by an $N \times N$ (N represents the number of the nodes) adjacency matrix A, where the element $A_{ij} = 1$, if nodes i and j are connected and $A_{ij} = 0$ otherwise.

III. MODIFIED MUTUAL INFORMATION (MI) APPROACH

In probability theory and information theory, the Mutual Information (MI) of two random variables is a measure of the mutual dependence between the two variables. More specifically, it quantifies the "amount of information" (in units such as shannons, commonly called bits) obtained about one random variable through observing the other

 ISBN: 978-1-61208-698-9

random variable. The concept of mutual information is intricately linked to that of entropy of a random variable, a fundamental notion in information theory that quantifies the expected "amount of information" held in a random variable.

A. Mutual Information (MI) Approach

Considering a random variable X related to the outcome x_k and probability $p(x_k)$, its self-information $I(x_k)$ can be denoted as

$$I(x_k) = log\frac{1}{p(x_k)} = -log\,p(x_k) \quad (1)$$

where the base of the logarithm is specified as (1), thus the unit of self-information is bit. This is applicable for the following if not otherwise specified. The self-information indicates the uncertainty of the outcome x_k. Obviously, the higher the self-information is, the less likely the outcome x_k occurs.

Consider two random variables X and Y with a joint probability mass function p(x, y) and marginal probability mass functions p(x) and p(y). The mutual information I(X; Y) can be denoted as follows:

$$\begin{aligned} I(X;Y) &= \sum_{x\in X}\sum_{y\in Y} p(x,y)\,log\frac{p(x,y)}{p(x)p(y)} \\ &= \sum_{x,y} p(x,y)\,log\frac{p(x|y)}{p(x)} \end{aligned} \quad (2)$$

Thus, in the network, the mutual information between x_i and x_j can be represented as:

$$\begin{aligned} I(x_i,y_j) &= log\frac{p(x_i|y_j)}{p(x_i)} \\ &= -log\,p(x_i) - \left(-log\,p(\chi_i|y_j)\right) \end{aligned} \quad (3)$$

Mutual information is a measure of the dependency between two variables. $I(x_i, y_j) = 0$ represents that x_i and y_j are independent to each other. Considering link prediction method, we want to use local structure information to improve the prediction. For this purpose, we use Γ(x) to represent the set of adjacent nodes of node x. For node pairs (x,y), the set of their common neighborhoods is denoted as $O_{\chi y} = \Gamma(x) \cap \Gamma(y)$Given an unconnected node pair (x,y), if the set of its common neighbor $O_{\chi y}$ is available, the likelihood score of node pair (x,y) is defined as

$$I(L_{xy}^1|O_{xy}) = I(L_{xy}^1) - I(L_{xy}^1; O_{xy}) \quad (4)$$

$I(L_{xy}^1)$ is the self-information of that node pair (x,y) is connected. $I(L_{xy}^1; O_{xy})$indicates the reduction in uncertainty of the connection between nodes x and y due to the information given by their common neighbors.

If the elements of Oxy are assumed to be independent of each other, then

$$I(L_{xy}^1; O_{xy}) = \sum_{z\in O_{xy}} I(L_{xy}^1; z) \quad (5)$$

$$I(L_{xy}^1; z) = \frac{1}{|\Gamma(z)|(|\Gamma(z)|-1)}\sum_{m,n\in\Gamma(z)} I(L_{mn}^1; z) \quad (6)$$

$$I(L_{mn}^1; z) = I(L_{mn}^1) - I(L_{mn}^1|z) \quad (7)$$

Here $I(L_{xy}^1; z)$ is defined as the average mutual information over all node pairs connected to node z. $I(L_{mn}^1|z)$ is the conditional self-information of that node pair (m,n) is connected when node z is one of their common neighbors, and $I(L_{mn}^1)$ denotes the self-information of that node pair (m,n) has one link.

B. A Modified Model

Since the computation of the Mutual Information of pair nodes costs much time, we want to simplify it. In formula (2), it is easy to relate the sum of possibility $p(x,y)\,log\frac{p(x|y)}{p(x)}$ to the expectation of $log\frac{p(x|y)}{p(x)}$, so we change the formula (2) into (8)

$$\begin{aligned} I(X;Y) &= \sum_{x\in X}\sum_{y\in Y} p(x,y)\,log\frac{p(x,y)}{p(x)p(y)} \\ &= \sum_{x,y} p(x,y)\,log\frac{p(x|y)}{p(x)} \\ &= E_{x,y\leftarrow p(x,y)}\left[log\frac{p(x|y)}{p(x)}\right] \end{aligned} \quad (8)$$

We can sample the network nodes, and calculate the $log\frac{p(x|y)}{p(x)}$ of them. When the sample is big enough,The expectation in formula (8) is close to the real $I(X;Y)$.

IV. CONCLUSION

By modifying the model, we repair the big bug in the traditional Baysian method in network link prediction. The method is simple and fast. It approximates the real value with simulation results, and saves a lot of computing time.

REFERENCES

[1] F. Tan, Y. Xia, and B. Zhu, "Link Prediction in Complex Networks: A Mutual Information Perspective" Plos One, 2014, 9(9):e107056.

[2] Z. Boyao, X. Yongxiang, and S. N. Irene , "Link Prediction in Weighted Networks: A Weighted Mutual Information Model" PLOS ONE, 2016, 11(2):e0148265-.

[3] H. Shakibian and N.M. Charkari. "Mutual information model for link prediction in heterogeneous complex networks" Scientific Reports, 2017, 7:44981.

[4] A.V.D. Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding", 2018.

[5] L. Lü et al. "Toward link predictability of complex networks", Proceedings of the National Academy of Sciences, 2015, 112(8):2325-2330

 ISBN: 978-1-61208-698-9

Modular-Based Maintenance for Load-Sharing System with Random Repair Time and Non-Identical Components

Nasser Fard
Department of Mechanical and Industrial Engineering
Northeastern University
Boston, USA
e-mail: n.fard@neu.edu

Yuanchen Fang
Department of Mechanical and Industrial Engineering
Northeastern University
Boston, USA
e-mail: fang.yua@husky.neu.edu

Huyang Xu
Department of Mechanical and Industrial Engineering
Northeastern University
Boston, USA
e-mail: xu.hu@husky.neu.edu

***Abstract*—In this paper, decision-making of repairable load-sharing k-out-of-n is discussed. Decision variables are related to system degradation and restoration. By exploiting these decision variables combinations, the optimal design solution is selected by utilizing weighted principal component analysis based multi-response optimization. The mathematical modeling of the decision-making process is based on the statistical flowgraph model. The statistical flowgraph model is used to describe degradation and restoration with the advantage of computation over the traditional Markovian model. Based on the statistical flowgraphs of different factorial decision variables combinations, the reliability-related measurements of load-sharing system can be evaluated, which correspond to the responses in the multi-response optimization problem.**

Keywords- system reliability; modular design; multi-response optimization.

I. INTRODUCTION

To improve system reliability, the well-known approaches are based on the determination of component reliabilities and their system configuration. That is, the system reliability can be improved by reducing the system complexity, using the highly reliable components and structural redundancy. Additionally, if the system is repairable, then a planned maintenance, repair schedule and repair policy can be used to increase the system availability.

If the computation of system reliability is based on the critical assumption of independent failures among components, the system reliability is determined by applying an appropriate reliability for each component of the system and the rules of probability according to the system configuration. However, when component failures are dependent, more powerful methods, such as Markov analysis, may be needed [1].

Concentrating on a parallel configuration, which is used for including redundant components in the system, if independence is assumed across the components in the system, a failure of any component does not affect the failure rates of surviving components in a parallel configuration. On the other hand, in any multicomponent system, the failure of one component can affect the performance of the remaining components [2]. That is, many systems are structured to share loads among components, which is known as load-sharing. For a load-sharing system, the assumption of independence is unreasonable. The system reliability can be estimated from dependencies among components, the knowledge of components and their system configuration.

With the purpose of analyzing the reliability of load-sharing systems, the relationship between the load and the failure behavior of a component, described by the failure rate of the component, is considered. For example, Tierney [3] proposed two load-sharing settings for fibers in the parallel arrangement. Assume that there is little or no cohesion between fibers. Once a fiber fails, the surviving fibers share the steady and tensile load equally and uniformly. In the second setting, the load of a failed fiber is transferred to an adjacent fiber based on the shape of the set of adjacent failed fibers. As a generalization of the two previous settings, a load on any individual component monotonically increases as other components fail.

For a repairable system, when components fail, the system can be restored. Due to the variety of failure causes, the repair times are random. Therefore, studies of probabilistic repair times and repair performance levels are necessary. Since both time-to-failure and repair time are stochastic processes, as mentioned earlier, Markov process is the most common mathematical methodology for the reliability design of the repairable system.

Once the system is repairable, the maintenance of the system can be classified into three categories. The first one is the corrective maintenance. In this case, the system is repaired based on the system failure only. Once the system failure is significant due to a series of losses, preventive maintenance should be used. There are two policies for preventive maintenance. Given the failure distributions of components, one can plan a repair treatment before the failure of components occurs. On the other hand, if the analysis of component failures concentrates on the physical evaluations, conditional maintenance can be performed based on continuous records of specific measurements, which have thresholds to indicate the component failure events. The third maintenance strategy is reliability-centered maintenance. It is a corporate level maintenance strategy based on analysis and testing of factors, which affect the reliability of components systematically.

In the design phase, maintenance is a functional design problem. System modular design is beneficial to the competition since the system is reconfigurable based on the functional combination of modules in the system. By exploring the relationship between system configuration and

 ISBN: 978-1-61208-698-9

separation of functional requirements, the functional combination is implemented. Proposed by [4], the modularity of a system depends on two characteristics of the design: 1) similarity between the physical and functional architectures of the design and 2) minimization of incidental interactions among physical components. In this paper, modular design concentrates on maintenance. The design decision includes the number of components in each maintenance module and the selection of component type for each module. It is assumed that each module has the same type of components.

For the load-sharing k-out-of-n systems, the repair time of each component is arbitrary distributed. This assumption is more valuable than constant repair time assumption in realistic applications. On the other hand, for the repairable load-sharing k-out-of-N configuration, Markov chain in which the states represent the number of failed components in the system has the strictest assumption. Comparing with the Markovian model, the flowgraph model is a graphical representation of a stochastic system in which possible outcomes are connected by directed line segments. This model provides a new computational way to the reliability evaluation of load-sharing k-out-of-N system based on Moment Generating Functions (MGFs). The use of MGFs simplifies the computational multiplication of different distribution functions. By linking covariates into branch transition, the MGFs of the system failure are evaluated under different covariates levels so that presence of external events can be described in quantitative way. In the proposed model, during the repair process of failed components, the operating component can fail.

The Weighted Principal Component Analysis (WPCA) based multi-response optimization [5] is used for determining components, system configuration and maintenance policy. This methodology is beneficial to the optimization problem, in which both network parameters and network structures (nodes and edges) of potential designs vary.

The paper provides a new computational framework based on statistical flowgraph model for repairable load-sharing k-out-of-n problem. Integrating the maintenance-based modular design concept into maintenance task, WPCA-based multi-response optimization is applied to determine the optimal design factorial combination. The remaining sections are organized as follows. Section 2 reviews the development of models for analyzing repairable load-sharing system. Section 3 proposes the flowgraph model and the methodology for computing system failure time MGF with different combinations of covariate levels. Section 4 presents the multi-response optimization for the module system and repair policy design, and Section 5 shows the detailed procedures of the proposed framework by a numerical example.

II. STATE OF THE ART

Most of the load-sharing k-out-of-n system models assume constant failure rate for every component, which can be either analytically solved [6] or represented by the Markov transition diagram [7]. On the other hand, the assumption of time-varying failure rates is proposed as well [8]. Another attempt was proposed by Liu [9], who modeled the component failure time distribution by proportional hazards model and the load changes by piecewise constant function. However, the generalizations of the models in these studies are limited by their computation complexity. Similarly, Liu and Mohammad et al. [10] presented a model in which the load-dependent time-varying failure rate of each component is expressed by Cox's proportional hazards model and provided a closed form expression for the system reliability when all components are identical. To reduce the computation complexity induced by multiple integrations for failure dependency, Suprasad et al. [11] proposed a series of models which can solve large systems in a short time. They considered two classes of models accounting for the effects of load history: tampered failure rate (TFR) model and cumulative exposure (CE) model. They converted the TFR load-shared model with general failure distributions and used the concept of supplementary variables in semi-Markov processes to model the effects of load history on system life for CE model [12]. A slightly different perspective of modeling load-sharing k-out-of-n system is based on task allocation and queueing, in which the load is considered as the tasks assignment on each component. Huang and Xu [13] studied such models and introduced the concept of queueing system and cumulative time in each state to generate a closed-form expression for reliability of load-sharing k-out-of-n system with arbitrary failure distributions.

Despite a wide range of applications for load-sharing redundant systems, the methods for lifetime-related performance evaluation and design of repairable load-sharing k-out-of-n systems are limited. The failure dependency as well as maintenance process complicate the states and transitions between states, so that it is of great challenge to model the lifetime reliability of such system. Shao and Lamberson [14] studied a Markov model for analyzing a shared-load repairable k-out-of-n system with imperfect switching. It assumed that all the components are identical with constant failure rates and constant repair rates. Although the repair rule declared in their paper considered more than one component at a time in each repair, the model used considered only one repair transition from each state to its one-step backward state, which means that only one component can be repaired at a time. Hasset et al. [8] extended Shao's model by considering time-varying failure rates and time-varying repair rates within states transitions and solved a 2- component failure-dependent parallel system. As stated by the authors, the computation of non-constant failure rate or repair rate models is rather challenging because the general solution for such model is intractable. A response to such intractability is to assume identical Weibull failure time distributions and identical constant repair rates. Even with these simple assumptions, the expression for system reliability and availability is extremely complicated and tedious to evaluate. Therefore, Amari et al. [15] proposed an efficient algorithm based on symmetric switching functions and iterative implementation to approximate the reliability, availability and failure distribution of a repairable k-out-of-n system with identical/non-identical components, which has O(kn) computational complexity. However, the cases with non-identical components still assume constant failure rates, and load-sharing was not considered in the paper. Different from the Markov models proposed in the previous papers, where

 ISBN: 978-1-61208-698-9

the states represent the number of failed components in the system, Mandziy et al. [16] modeled a detailed Markov chain where the states representing the failure of components at specific locations caused by specific fault mode for a simple 3-component system. In this study, the component failure time was distributed by Weibull, and load sharing effect was set by the scale functions depending on the status of all survival components in the system. A failed component could be repaired as long as it does not cause system failure, and the repair time was assumed exponentially distributed and identical for all components.

As a generalization of Markov process, semi-Markov process creates flexibility in modeling system degradation and recovery. In semi-Markov model, the states being successively visited are governed by a Markov chain, and the transition time distribution can be arbitrarily specified. Hellmich [17] modeled the repairable load-sharing k-out-of-n: G system with identical components by nonhomogeneous semi-Markov process, in which the failure time distribution of each component was arbitrary and repairable. But the repair time distribution was restricted to be exponential, because when the system was in the state that q components fail and another component failed, the process transited to the next state that q + 1 components fail and the repair of the previous failed component had to be forgotten which forced the repair process to be memoryless.

Although a semi-Markov multistate model provides a way allowing the transition time to a future state to depend on the duration of time spent in the current state, it is quite difficult to analyze data for semi-Markov models in practice. Flowgraphs model semi-Markov processes and allow a variety of distributions used within the multistate model. The "states" in the flowgraphs are all the possible outcomes of a stochastic system. The waiting time distributions of the change of states are formulated by MGFs. Moreover, flowgraphs can easily handle reversibility [18]. These give flowgraphs the natural advantage in analyzing time-to-event data and modeling system reliability performance. Jenab and Dhillon [19] used flowgraph to model the k-out-of-n system in which every component in the system had a failure detection – isolation – repair loop and all components were assumed to be identical and operating independently. Jenab and Dhillon [20] then extended this model to adapt the reversible multi-state case, where each unit in the system could transit from better states to worse states due to aging effect, or from worse states to better states due to repair, and the degradation and recovery process was a semi-Markov model and was represented by the flowgraph model. The load sharing was simply represented by changing the states from the level of degradation to the level of load carried by that unit in the system function, assuming all the units in the system are identical. In this paper, the flowgraph model is embedded in a novel framework for modular based design and extended for more general system in which the components are not necessarily identical and the failure and repair time distributions are not necessarily exponential.

From the previous reviews, it can be seen that Markov process is restricted to memoryless property. That is, the transition time from one state to another is exponential distributed. Although the semi-Markov process relaxes this restriction, the corresponding computational task is a challenge. Based on the property of MGFs, statistical flowgraph model is advantageous in regards to the challenge. Based on statistical flowgraph model, the repairable load-sharing k-out-of-n system is studied. By introducing the modular design concept, the intermediate layer between the top level and bottom level (component-level) is introduced, and the decision variables can be discussed for optimizing the corresponding repairable system. Specially, in the loading-sharing k-out-of-n system, the number of components in the system becomes a decision variable, which is denoted by N. Therefore, in this paper, the computational process encompasses the system degradation (step-by-step failures) and restoration (maintenance tasks) into a decision-making perspective. By using multi-response optimization methodology, the optimal factorial combination is obtained at last.

III. PROBLEM DESCRIPTION

A parallel model consists of n components in active redundancy, of which k ($1 \le k < N$) are necessary to perform the required function [21]. For many practical applications, load sharing is a suitable design to explain the dependence among components. Consider a k-out-of-N: G system with independent components, the system is put into operation at time zero, all components are functioning, and they are equally sharing a constant load that the system is supposed to carry [22]. When the system experiences component failures, the surviving components must carry the same load on the system. Considering the target performance levels prescribed in the design phase, the redundancy level is a decision variable, denoted by N, should be determined so that there is a suitable redundancy level in the system.

System design with prefabricated modules encompasses the production and use of preplanned modules as a solution to build with higher quality and more efficiency [23]. In order to manufacture systems in a manageable and economic way, prefabricated modules and adaptable module frames are selected, customized, and assembled [24]. For a k-out-of-N: G system, prefabricated modules are configured in the parallel structure to build redundancy of the system. Take a redundancy system with five components as example, which is illustrated in Figure 1. There are two types of modules: module 1 and module 2, in a shared-load k-out-of-5: G system, k = 1, 2, 3, 4, 5. Without loss of generality, the failure distributions of these components are not necessary to be identical.

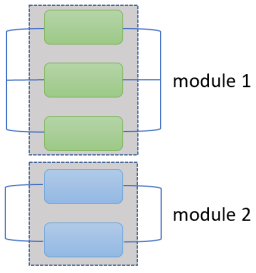


Figure 1. Shared-load k-out-of-5: G system with two types of module

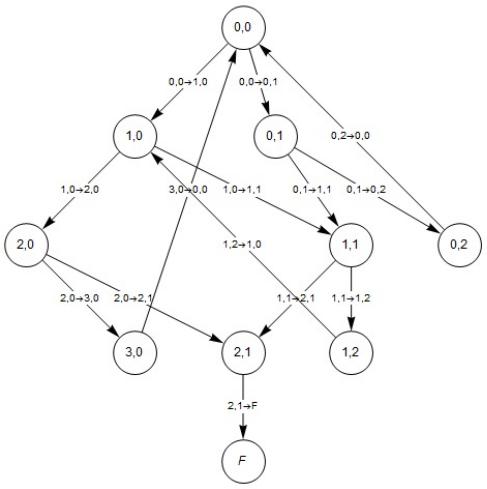
 ISBN: 978-1-61208-698-9

On the other hand, considering the stability of manufacturing processes, the types of prefabricated modules should be limited at an affordable level. Suppose that the manufacturer produces M types of prefabricated modules, and M is a small integer. The numbers of components in each module are denoted by $\{m_1, \ldots, m_M\}$. Denote the numbers of each module in the system by $\{n_1, \ldots, n_M\}$. Then the total number of components in the system is $\sum_{j=1}^{M} m_j n_j = N$ which is equal to the redundancy level of the system.

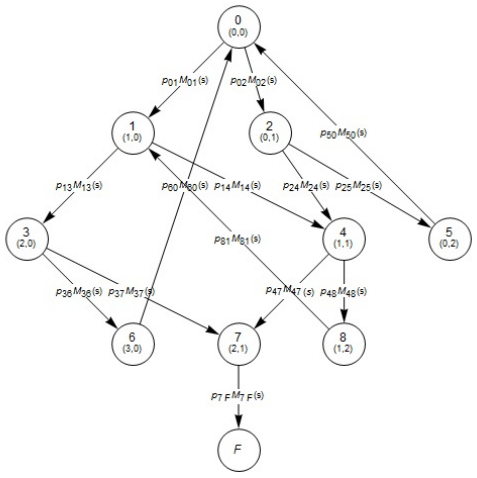
For a k-out-of-N: G system, two types of cost, interface cost and encapsulate cost, are assumed. The interface cost depends on the number of modules in the system, and the encapsulate cost depends on the number of components in the module. Therefore, the total design cost of the system is $c_1 \sum_{j=1}^{M} n_j + c_2 N$, where c_1 and c_2 are cost coefficients of the interface cost and the encapsulate cost respectively, $\sum_{j=1}^{M} n_j$ is the total number of modules in the system.

Additionally, for a k-out-of-N: G system, a maintenance policy determines how and when the maintenance should be performed in order to avoid the system failures. Basic maintenance policies, such as age repair, periodic repair, and block repair polices are usually suggested for non-modularization systems. In particular, a maintenance policy, described by [25], is that the failed components are replaced if and only if the failed components are contained within the critical component set. Inspired by this policy, in this paper, we assume that a module can be replaced if and only if all components in the module are failed based on continuous monitoring.

The analysis of system performance presented in this paper is based on the system reliability analysis upon the flow-graph concept. The reliability of modularized shared-load k-out-of-N: G system is evaluated by using the concept of the flowgraph and MGF. A flowgraph is a graphical representation of a stochastic system in which possible outcomes are connected by directed line segments. Possible outcomes for the system reliability analysis are determined by the components failures in each module. Define a M-tuple, $\mathbf{O} = (O_1, \ldots, O_j, \ldots, O_M)$ in order to describe the possible outcomes, where O_j, j = 1, …, M, is a variable representing the number of failed components in the module j. If all components in the system are operating, then $\mathbf{O} = (0, \ldots, 0, \ldots, 0)$ and the system is in state 0. When one component in the system fails, assume that the failed component belongs to the module j, $j \in \{1, \ldots, M\}$, then $\mathbf{O} = (0, \ldots, 1, \ldots, 0)$. The number of states used to represent one component failure case is equal to the number of modules in the system. When the second failure occurs, it is assumed that the second failed component belongs to module j, for $j \in \{1, \ldots, M\}$. It is possible that i = j, that the two failures occur in the same module. If i = j, $\mathbf{O} = (0, \ldots, 2, \ldots, 0)$. If $i \neq j$, $\mathbf{O} = (0, \ldots, 1, \ldots, 1, \ldots, 0)$. Similarly, all the possible outcomes (states) are determined. Once the total number of failed components is greater than k, $\sum_{j=1}^{M} O_j > k$, the system fails and the system state enters to failure state, called F. For maintenance policy, once all components of a module fail, $O_j = m_j$, $j \in \{1, \ldots, M\}$, the module j is replaced immediately with measurable probabilistic repair time. For the shared-load k-out-of-5: G system in the Figure 1, Figure 2 (a) gives the flowgraph for k = 2 where the transitions of states are indicated on the branch.



(a)



(b)

Figure 2. Flow graph for the example in Figure 1: (a) branches are labeled with transition between states; (b) branches are labeled with transmittances

In the flowgraph, each branch has a transition probability, p_{xy}, and a waiting time distribution associated with the transition from its beginning and ending nodes, $M_{xy}(s)$, where x and y denote the nodes in the flowgraph. Figure 2 (b) shows the flowgraph of the example in Figure 1 with k = 2, where the branches are labeled with transition probability and MGF of waiting time. p_{xy} is determined by the number of survival components in each module and the module in which the next failure occurs. Let $\mathbf{O_x} = (O_{x1}, \ldots, O_{xj}, \ldots, O_{xM})$ denote the component failure state of node x, x = 0, 1, …, F, where O_{xj} is the number of failed components in module j when the system is in state x. The transition probability from state x to state y, is

$$p_{xy} = \frac{m_h - O_{xh}}{N - \sum_{j=1}^{M} O_{xj}}, \text{for } \sum_{j=1}^{M} O_{xj} + 1 = \sum_{j=1}^{M} O_{yj},$$

where N is the total number of components in the system, and h represents the module in which the new failure occurs, h ∈ {1, …, M}. For example, in Figure 2, state 4, (1, 1), represents that there is one failed component in module 1 and one failed component in module 2. Correspondingly, there are 5 – (1 + 1) = 3 survival components, in which two of them are in module 1 and the other one is in module 2. State 7, (2, 1), represents that the next failure component is in module 1. Thus, the probability of transformation from node 4 to 7 is

$$p_{47} = 1/[5 - (1 + 1)] \times 2 = 2/3$$

It is assumed that a module will be replaced when all components in that module fail. Thus, when the system reaches node Ox = (Ox1, …, Ox(j-1), mj, Ox(j+1), …, OxM), j = 1, …, M, module j is replaced with a new one, and the system will be transferred to Oz = (Ox1, …, Ox(j-1), 0, Ox(j+1), …, OxM), because the number of failed component in module j is restored to 0. In this case, the transition probability is

$p_{xz} = 1$, from $\mathbf{O_x} = (O_{x1}, \ldots, O_{x(j-1)}, m_j, O_{x(j+1)}, \ldots, O_{xM})$ to $\mathbf{O_z} = (O_{x1}, \ldots, O_{x(j-1)}, 0, O_{x(j+1)}, \ldots, O_{xM})$

Let T_{xy} be the random waiting time in state x until the transition to y occurs, x, y = 0, 1, …, F, x ≠ y, and $M_{xy}(s)$ be the MGF of T_{xy}

$$M_{xy}(s) = E(e^{sT_{xy}})$$

provided that the expectation exists for s in an open neighborhood of 0 [18]. T_{xy} on branches from state x to y, x < y, are the component failure times, and T_{xy} on branches from state x to y, x > y are the times to repair the failed component. T_{xy} can follow any arbitrary distributions. This paper assumes that the failure time of each component follows exponential distribution,

$$M_{xy}(s) = \frac{\lambda_{xy}}{\lambda_{xy} - s}, \text{for } \sum_{j=1}^{M} O_{xj} + 1 = \sum_{j=1}^{M} O_{yj},$$

where λ_{xy} is the failure rate of the failed component causing transition from node x to y and a function of various covariates and the shared load on that component. We assume that the repair time is normally distributed with mean μ and standard variation σ for all types of modules,

$$M_{xz}(s) = e^{\mu s + \frac{\sigma^2 s^2}{2}}, \text{for } \boldsymbol{O_x} = (O_{x1}, \cdots, m_j, \cdots, O_{xM}) \text{ and } \boldsymbol{O_z} = (O_{x1}, \cdots, 0, \cdots, O_{xM}),$$

The first step in this problem is to compute the overall transmittance of the entire flowgraph from the initial state 0 to the end state F, M(s). After identifying all paths, loops, and loops not connecting the path between nodes of state 0 and state F, M(s) is computed by Mason's rule. For details of computing M(s), refer to [18]. M(s) determines the distribution of the system life time. The flowgraph model concerns modeling the probabilities of the outcomes, the failure/repair time distributions of the outcomes, and manipulating the flowgraph to access overall failure time distribution. Once the system failure time MGF M(s) is computed, the system reliability measurements, such as mean time to failure, and average number of repairs at specified covariate levels can be determined. Therefore, for each combination of covariate levels, system life time, total design cost and performance deviation are considered, to obtain a criterion for design's quality.

IV. MODULE DESIGN USING MULTI-RESPONSE OPTIMIZATION

Suppose there are S different operating conditions. For the i^{th} operating condition, a system $\mathbb{S}^{(i)}$ consists of modules needs to be designed, i = 1, 2, …, S. Suppose there are M types of modules to be allocated to each system. Let m_j denote the number of components in the j^{th} type of module, j = 1, 2, …, M. Let $n_j^{(i)}$ denote the number of type j modules in system $\mathbb{S}^{(i)}$ designed for operating condition i, i = 1, 2, …, S, j = 1, 2, …, M. Therefore, the factors to be determined are m_j and $n_j^{(i)}$, i = 1, 2, …, S, j = 1, 2, …, M, and the possible values for them are the factor levels.

In this paper, system mean time to failure (MTTF), system failure time standard deviation (SD), average number of module repair before system failure (MRep), and total design cost (Cost) are selected as four response variables. The proposed method aims to obtain optimal values of m_j and n_j, while minimizing a function of four response variables through WPCA. Once the system MGF, $M_{\mathbb{S}^{(i)}}(s)$ is calculated following the method stated in Section 3, MTTF and SD can be obtained by

$$\text{MTTF}^{(i)} = \left.\frac{dM_{\mathbb{S}^{(i)}}(s)}{ds}\right|_{s=0} \quad (1)$$

$$\text{SD}^{(i)} = \left.\frac{d^2 M_{\mathbb{S}^{(i)}}(s)}{ds^2}\right|_{s=0} - \text{MTTF}^2 \quad (2)$$

To compute the distribution of repair occurrence, an auxiliary constant 1 is created and its MGF, e^u, is attached to the branch of repair. For example, in Figure 2, branch 5 → 0, 6 → 0, and 8 → 1 are the repair transitions, and the transmittance about these branches are changed to $p_{50}e^{\mu}M_{50}(s)$, $p_{60}e^{\mu}M_{60}(s)$, and $p_{81}e^{\mu}M_{81}(s)$. The overall system life MGF is computed as described in Section 3. The joint MGF of the distribution of system life time and number of repairment is $M_{\mathbb{S}^{(i)}}(s,u)$. Then, MRep can be calculated by taking the first derivative of $M_{\mathbb{S}^{(i)}}(s,u)\big|_{s=0}$ over u and letting u = 0,

$$\text{MRep}^{(i)} = \left.\frac{dM_{\mathbb{S}^{(i)}}(s,u)\big|_{s=0}}{du}\right|_{u=0} \quad (3)$$

The total design cost for system $\mathbb{S}^{(i)}$ is determined by the number of components and number of modules in the system. Let pe_j denote the cost of individual component in the j^{th} type of module, j = 1, 2, …, M, and pi is the cost of interface of each module. Therefore, for a system $\mathbb{S}^{(i)}$ consisting of $n_j^{(i)}$ type j modules, the total design cost is

$$\text{Cost}^{(i)} = \sum_{j=1}^{M} pe_j \times n_j^{(i)} \times m_j + pi \times \sum_{j=1}^{M} n_j^{(i)} \quad (4)$$

The objective for this optimization is to find the best combination of m_j and $n_j^{(i)}$, i = 1, 2, …, S, j = 1, 2, …, M, such that $\text{MTTF}^{(i)}$ is as close as possible to a target value $\text{MTTF}_0^{(i)}$, and $\text{SD}^{(i)}$, $\text{MRep}^{(i)}$, $\text{Cost}^{(i)}$ are minimized simultaneously among all operating conditions.

 ISBN: 978-1-61208-698-9

WPCA based Multi-Response Optimization method is applied to the experimental design, and the unique optimal combination of m_j and $n_j^{(i)}$, i = 1, 2, …, S, j = 1, 2, …, M is determined. WPCA based multi-response optimization utilizes PCA to map the original data to a new vector of component scores and transforms the original response variables into uncorrelated principal components. Each component is multiplied by a weight to emphasize the contribution of components based on their corresponding variation. All the weighted components are combined into one multi-response performance index (MPI), and the optimal result is the factor level combination with the largest MPI. The detailed procedure of WPCA multi-response optimization with unique solution can be found in [5]. The optimal value of m_j, j = 1, 2, …, M gives the modular design which has high manufacturing performance and accommodates to various demands under different operating conditions.

V. EXAMPLE

Suppose a manufacturer is producing three redundancy systems of electric motors as in Table I, which are designed to supply power under three operating conditions:

TABLE I. OPERATING CONDITIONS

Operating condition	Type of service application	Operating temperature, °F	Operating altitude, ft
1	Heavy shock load	5	500
2	Light shock load	75	3600
3	Uniform and steady load	140	40

Two types of electric motor are considered in the design, which are shown in Table II:

TABLE II. TYPES OF ELECTRIC MOTORS

Electric motor type	Shaft Material	Shaft surface manufactured finish	Viscosity of lubricant used in	
			bearing system	gear system
A	Alloy steel	Polished	1.0	1.0
B	Cast aluminum	Ground	0.8	1.2

The objective is to design the two types of module $\mathcal{M}^{(A)}$ and $\mathcal{M}^{(B)}$, where $\mathcal{M}^{(A)}$ is a parallel structure of type A electric motors, and $\mathcal{M}^{(B)}$ is a parallel structure of type B electric motors, such that $\mathcal{M}^{(A)}$ and $\mathcal{M}^{(B)}$ have high resilience to accommodate the redundancy product design for three different operating conditions $\mathbb{N}^{(1)}$, $\mathbb{N}^{(2)}$, and $\mathbb{N}^{(3)}$. $\mathbb{N}^{(1)}$, $\mathbb{N}^{(2)}$, and $\mathbb{N}^{(3)}$ consist of different allocations of $\mathcal{M}^{(A)}$ and $\mathcal{M}^{(B)}$. Therefore, the objective is to determine the optimal number of components in the two modules $\mathcal{M}^{(A)}$ and $\mathcal{M}^{(B)}$,

$$m_A, m_B$$

and the number of modules $\mathcal{M}^{(A)}$ and $\mathcal{M}^{(B)}$ in the three redundancy systems $\mathbb{N}^{(1)}$, $\mathbb{N}^{(2)}$ and $\mathbb{N}^{(3)}$, respectively,

$$n_A^{(1)}, n_B^{(1)}, n_A^{(2)}, n_B^{(2)}, n_A^{(3)}, n_B^{(3)}$$

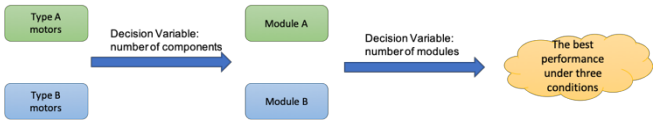


Figure 3. Illustrated Optimal Decision Procedure

for simultaneously meeting the requirements of mean time to failure (MTTF) for each operating condition, while minimizing the standard deviation of time to failure, average number of replaced modules before system failure, and the interface and encapsulate cost. Let $N^{(i)}$ denote the total number of electric motors in redundancy system i,

$$N^{(i)} = n_A^{(i)} m_A + n_B^{(i)} m_B, i = 1,2,3.$$

The failure time of an individual electric motor is exponentially distributed. Each module is considered as a tampered failure rate (TFR) load-sharing k-out-of-n: G system with identical electric motors where all surviving motors equally share the load. The system is consisted of different modules, which have non-identical types of electric motors. In this example, it is assumed that k = 3, and the module is replaced with a new one when all motors in that module fail, and the repair time is normally distributed with mean of 0.00048 million hour and standard deviation of 0.00024 million hour.

The failure rate model of an electric motor is based upon the failure rate of its parts, which includes windings, stator housing, armature shaft, bearings, and gears [26]. Failure mechanisms resulting in part degradation and failure rate distributions are independent in each failure mode. The total electric motor failure rate is the sum of the failure rates of each part in the motor, which are functions of covariates.

The target mean time to failure for each operating condition is assumed to be 8.76 thousand hours (1 year) for all operating conditions. The cost of individual electric motor is \$120 per type A motor and \$ 100 per type B motor, and the cost of module interface is \$100 per module. The possible choices of m_A and m_B are 1, 2, 3, or 4. The possible choices of $n_A^{(1)}, n_B^{(1)}, n_A^{(2)}, n_B^{(2)}, n_A^{(3)}, n_B^{(3)}$ are 0, 1, 2, or 3.

By introducing the covariate and load-sharing failure rate model to the system, we calculate the MGF of failure time for each combination of decision variables, $m_A, m_B, n_A^{(1)}, n_B^{(1)}, n_A^{(2)}, n_B^{(2)}, n_A^{(3)}$ and $n_B^{(3)}$. From the failure time MGF, MTTF and standard deviation of failure time are calculated. Factors $(m_A, m_B), (n_A^{(1)}, n_B^{(1)}), (n_A^{(2)}, n_B^{(2)}), (n_A^{(3)}, n_B^{(3)})$ are considered as four pairs of factors, and each factor pair consists of $4 \times 4 = 16$ levels. The 16^4 full experimental design is used. With cost of interface and encapsulate, the multi-response optimization experimental design is shown as Table III. Based on the Figure 3, we want to select the best system performance, described by the four responses. The objective is to determine the optimal number of components in the two module types and the number of modules in the redundancy systems. Therefore, in the Table III, the first eight columns of each row indicate sets of candidate designs, which are combinations of the number of motors in each module

 ISBN: 978-1-61208-698-9

associated with the three operating conditions. Based on these controllable design factors, we can evaluate the four responses (MTTF, SD, MRep, Cost) to describe the system performance. These four responses are computed based on the statistical flowgraph model described in Section 3.

For each operating condition, all these four responses are monitored so that responses in this example can be modeled in a three-level hierarchical structure. The top layer is about the system performance, the intermediate layer is about the three operating conditions, and the bottom layer is about the four responses. PCA-based multi-response optimization, as described earlier, can relax the response correlation problem, particularly for this hierarchical structure. On the other hand, there are two ways to solve the multi-response optimization problem: feature selection and dimensional reduction. For this hierarchical structure, feature selection is challenged since the intermediate levels (operating condition in this example) are equally important. This is another advantage of PCA-based multi-response optimization. In the experimental conduction perspective, the experiments were done by testing all the possible combinations of $(m_A, m_B), (n_A^{(1)}, n_B^{(1)}), (n_A^{(2)}, n_B^{(2)}), (n_A^{(3)}, n_B^{(3)})$. For easy visualization, the experiments with MTTF significantly deviating from the target value were eliminated from the table III.

Following the procedures proposed in [5] and [26], the unique solution WPCA based multi-response optimization method is applied on the data in Table III. Table IV summarizes the resulting MPIs. It can be seen that the optimal design for module $\mathcal{M}^{(A)}$ and $\mathcal{M}^{(B)}$, accommodating the three operating conditions, is to allocate two type A motors in module $\mathcal{M}^{(A)}$ and allocate three type B motors in module $\mathcal{M}^{(B)}$. The MTTF of system under operation condition 1, 2, and 3 are 8.2171, 8.1227, and 9.6936 thousand hours, respectively.

One of the most commonly used strategies of system design and repair rule is to consider every single component as an individual module which is subject to repair upon failure. To compare it with the proposed framework, the flowgraph is modified where the repair branch is added to every state with component failure. Let the system operates for 8.2171, 8.1227, and 9.6936 thousand hours under three operation conditions respectively. Following the procedures stated in Section 4 and the formulation of (3) and (4), the expected number of repairs and system configuration cost for different combinations of components are computed and summarized in Table V.

Compared the design options obtained in Table III and V, it is obvious that the design that immediately repairs every failed component leads to much greater times of repair action. For example, the expected number of repairs for the optimal solution under operating condition 1 (two modules with three type B motors in each module) is 0.1111. However, without considering modular repair rule, the system with six individual type B motors leads to an average of 1.6579 times of repair for operating the same length of time. Moreover, the system configuration cost is much higher under non-modular design (e.g., \$800 for the system with two modules with three type B motors in each module and \$1200 for the system with six individual type B motors), since the increased number of interfaces increases the total cost of the system.

VI. CONCLUSION AND FUTURE WORK

In this paper, incorporating the reliability concept into design for repairable systems is discussed. In the operating stage, for a disruptive event, the proposed maintenance strategy based on modular design provides a way to recover the system in the most appropriate way. In order to quantify the reliability-related performance in the design phase, a flowgraph model is introduced. The usage of flowgraph relaxes exponentially distributed assumptions for the state transition time, so that the proposed framework can model the problem with arbitrary distributions of failure time and repair time. Meanwhile, by linking the flowgraph with covariates, the model can be used when considering various external variates, such as different environmental conditions in which the system is operating. By applying the WPCA based multi-response optimization, the best design of modules and system can be obtained. The application of flowgraph is restricted by the complexity of the graph, because all the computations are based on functions [27]. The function-based operation limits the computation speed, and the higher graph complexity increases the number of functions involved, thus reducing the computation speed. Therefore, a novel algorithm for high efficiency flowgraph computation is needed and will extend the application of the proposed methodology in the future.

REFERENCES

[1] C. E. Ebeling, An Introduction to Reliability and Maintainability Engineering, 2nd ed.. Long Grove, IL: Waveland Press, Inc., 2010.

[2] I. Dewan and U. V. Naik-Nimbalkar, Load-Sharing Systems. Wiley Encyclopedia of Operations Research and Managment Science, 2010.

[3] L. Tierney, “Asymptotic bounds on the time to fatigue failure of bundles of fibers under local load sharing,” Advances in Applied Probability, vol. 14, no. 1, pp. 95-121, March 1982.

[4] K. Ulrich, “Fundamentals of product modularity,” in Management of Design. Dordrecht: Springer, pp. 219-231, 1994.

[5] N. Fard, H. Xu, and Y. Fang, “A unique solution for principal component analysis-based multi-response optimization problems,” The International Journal of Advanced Manufacturing Technology, vol. 82, no. 1-4, pp. 697-709, January 2016.

[6] E. M. Scheuer, “Reliability of an m-out-of-n system when component failure induces higher failure rates in survivors,” IEEE Transactions on Reliability, vol. 37, no. 1, pp. 73-74, 1988.

[7] H. Lin, K. Chen and R. Wang, "A multivariate exponential shared-load model," IEEE Transactions on Reliability, vol. 41, no. 1, pp. 165-171, 1993.

[8] T. F. Hassett, D. L. Dietrich, and F. Szidarovszky, "Time-varying failure rates in the availability & reliability analysis of repairable systems," IEEE Transactions on Reliability, vol. 44, no. 1, pp. 155-160, 1995.

 ISBN: 978-1-61208-698-9

[9] H. Liu, "Reliability of a load-sharing k-out-of-n: G system: non-iid components with arbitrary distributions," IEEE Transactions on Reliability, vol. 47, no. 3, pp. 279-284, 1998.

[10] R. Mohammad, A. Kalam, and S. V. Amari, "Reliability of load-sharing systems subject to proportional hazards model," Proceedings Annual Reliability and Maintainability Symposium (RAMS), pp. 1-5, 2013.

[11] A. V. Suprasad, M. B. Krishna, and P. Hoang, "Tampered failure rate load-sharing systems: status and perspectives," in Handbook of performability engineering, pp. 291-308, 2008.

[12] S. V. Amari and R. Bergman, "Reliability analysis of k-out-of-n load-sharing systems," Reliability and Maintainability Symposium (RAMS), pp. 440-445, 2008.

[13] L. Huang and Q. Xu, "Lifetime reliability for load-sharing redundant systems with arbitrary failure distributions," IEEE Transactions on Reliability, vol. 59, no. 22, pp. 319-330, 2010.

[14] J. Shao and L. R. Lamberson, "Modeling a shared-load k-out-of-n: G system," IEEE Transactions on Reliability, vol. 40, no. 2, pp. 205-209, 1991.

[15] S. V. Amari, M. J. Zuo and G. Dill, "O (kn) algorithms for analyzing repairable and non- repairable k-out-of-n: G systems," in Handbook of performability engineering, pp. 309-320, 2008.

[16] B. Mandziy, O. Lozynsky, and S. Shcherbovskykh, "Mathematical model for failure cause analysis of electrical systems with load-sharing redundancy of component," Przeglad Elektrotechniczny, vol. 89, no. 11, pp. 244-247, 2013.

[17] M. Hellmich, "Semi-Markov Embeddable Reliability Structures and Applications to Load- Sharing k-Out-of-n Systems," International Journal of Reliability, Quality and Safety Engineering, vol. 20, no. 2, 1350007, 2013.

[18] A. V. Huzurbazar, Flowgraph Models for Multistate Time-to-Event Data. Hoboken, NJ: John Wiley & Sons, Inc., 2005.

[19] K. Jenab and B. S. Dhillon, "K-out-of-n system with self-loop units. International Journal of Reliability," Quality and Safety Engineering, vol. 12, no. 1, pp. 61-73, 2005.

[20] K. Jenab and B. S. Dhillon, "Assessment of reversible multi-state k-out-of-n: G/F/Load- Sharing systems with flow-graph models," Reliability engineering & System safety, vol. 91, no. 7, pp. 765-771, 2006.

[21] A. Birolini, Reliability Engineering: Theory and Practice, 8th ed.. Springer, 2017.

[22] W. Kuo and M. J. Zuo, Optimal Reliability Modeling: Principles and Applications. Hoboken, NJ: John Wiley & Sons, Inc., 2003.

[23] U. Knaack, S. Chung-Klatte, and R. Kasselbach, Prefabricated Systems: Principles of Construction. Birkhäuser Architectur, 2010.

[24] H. Mayr, "GEM—A generic engineering framework for mechanical engineering based upon meta models," International Conference on Computer Aided Systems Theory (EUROCAST 1997), Springer Berlin Heidelberg, 1997, pp. 83-91, doi: 10.1007/BFb0025036

[25] M. Ram and J. P. Davim, Advances in Reliability and System Engineering. Springer, 2017.

[26] T. L. Jones, Handbook of Reliability Prediction Procedures for Mechanical Equipment. Naval Surface Warfare Center, Carderock Division, West Bethesda, Maryland, 2011.

[27] N. Fard, H. Xu, and Y. Fang, "Reliability assessment of load-sharing systems by flowgraph for non-identical components with time-varying repair rates," Proceedings of International Conference on Computers and Industrial Engineering (CIE), pp. 1-8, 2017.

TABLE III. MULTI-RESPONSE OPTIMIZATION EXPERIMENTAL DESIGN LAYOUT

Factor								**Response**											
Module Type		**Operating Condition**						**Operating Condition**											
A	B	1		2		3		1				2				3			
m_A	m_B	$n_A^{(1)}$	$n_B^{(1)}$	$n_A^{(2)}$	$n_B^{(2)}$	$n_A^{(3)}$	$n_B^{(3)}$	$MTTF^{(1)}$	$SD^{(1)}$	$MRep^{(1)}$	$Cost^{(1)}$	$MTTF^{(2)}$	$SD^{(2)}$	$MRep^{(2)}$	$Cost^{(2)}$	$MTTF^{(3)}$	$SD^{(3)}$	$MRep^{(3)}$	$Cost^{(3)}$
1	2	0	3	0	3	0	3	9.159	9.0651	0.7395	900	9.0652	8.9654	0.7395	900	6.5112	6.3642	0.7395	900
1	3	0	2	0	2	1	2	8.2171	5.8736	0.1111	800	8.1227	5.7918	0.1111	800	12.1395	9.8169	1.3793	1020
1	3	0	2	0	2	2	1	8.2171	5.8736	0.1111	800	8.1227	5.7918	0.1111	800	5.0004	3.6295	1.6667	840
3	1	2	0	2	0	1	2	9.0182	6.4383	0.1111	920	8.9048	6.3401	0.1111	920	5.0332	3.5329	1.6667	860
3	1	2	0	2	0	2	1	9.0182	6.4383	0.1111	920	8.9048	6.3401	0.1111	920	12.5481	10.0771	1.3793	1120
2	2	0	3	0	3	3	0	9.159	9.0651	0.7395	900	9.0652	8.9654	0.7395	900	6.9375	6.7645	0.7395	1020
2	2	0	3	1	2	3	0	9.159	9.0651	0.7395	900	9.3472	9.2459	0.7395	940	6.9375	6.7645	0.7395	1020
2	2	1	2	0	3	3	0	9.4473	9.3526	0.7395	940	9.0652	8.9654	0.7395	900	6.9375	6.7645	0.7395	1020
2	2	1	2	1	2	3	0	9.4473	9.3526	0.7395	940	9.3472	9.2459	0.7395	940	6.9375	6.7645	0.7395	1020
2	3	0	2	0	2	1	2	8.2171	5.8736	0.1111	800	8.1227	5.7918	0.1111	800	9.6936	11.4477	0.3668	1140
2	3	0	2	0	2	2	1	8.2171	5.8736	0.1111	800	8.1227	5.7918	0.1111	800	8.5759	9.1844	0.5541	1080
3	2	2	0	2	0	1	2	9.0182	6.4383	0.1111	920	8.9048	6.3401	0.1111	920	8.5049	9.0895	0.5541	1060
3	2	2	0	2	0	2	1	9.0182	6.4383	0.1111	920	8.9048	6.3401	0.1111	920	9.9104	11.6823	0.3668	1220
3	2	2	0	0	3	1	2	9.0182	6.4383	0.1111	920	9.0652	8.9654	0.7395	900	8.5049	9.0895	0.5541	1060
3	2	2	0	0	3	2	1	9.0182	6.4383	0.1111	920	9.0652	8.9654	0.7395	900	9.9104	11.6823	0.3668	1220
3	2	0	3	2	0	1	2	9.159	9.0651	0.7395	900	8.9048	6.3401	0.1111	920	8.5049	9.0895	0.5541	1060
3	2	0	3	2	0	2	1	9.159	9.0651	0.7395	900	8.9048	6.3401	0.1111	920	9.9104	11.6823	0.3668	1220
3	2	0	3	0	3	1	2	9.159	9.0651	0.7395	900	9.0652	8.9654	0.7395	900	8.5049	9.0895	0.5541	1060
3	2	0	3	0	3	2	1	9.159	9.0651	0.7395	900	9.0652	8.9654	0.7395	900	9.9104	11.6823	0.3668	1220
3	3	1	1	1	1	2	0	8.6172	6.1652	0.1111	860	8.5133	6.0747	0.1111	860	6.0477	4.0822	0.1111	920
1	4	2	1	2	1	2	1	13.1177	8.5674	2.1667	940	12.98	8.4503	2.1667	940	9.3589	5.7286	2.1667	940
2	4	1	1	1	1	1	1	8.0723	5.9972	0.1738	840	7.9792	5.9132	0.1738	840	5.5548	3.9151	0.1738	840
2	4	1	1	1	1	2	1	8.0723	5.9972	0.1738	840	7.9792	5.9132	0.1738	840	10.1503	11.9017	0.4611	1180
4	2	1	1	1	1	1	1	8.3358	6.1035	0.1738	880	8.2359	6.0151	0.1738	880	5.6737	3.9467	0.1738	880
4	2	1	1	1	1	1	2	8.3358	6.1035	0.1738	880	8.2359	6.0151	0.1738	880	10.141	11.863	0.4611	1180

TABLE IV. MULTI-RESPONSE OPTIMIZATION RESUTLS

Factor								MPI
Module Type		Operating Condition						
A	B	1		2		3		
m_A	m_B	$n_A^{(1)}$	$n_B^{(1)}$	$n_A^{(2)}$	$n_B^{(2)}$	$n_A^{(3)}$	$n_B^{(3)}$	
1	2	0	3	0	3	0	3	0.2467
1	3	0	2	0	2	1	2	1.1930
1	3	0	2	0	2	2	1	1.0590
3	1	2	0	2	0	1	2	0.8483
3	1	2	0	2	0	2	1	1.0065
2	2	0	3	0	3	3	0	0.2813
2	2	0	3	1	2	3	0	0.2171
2	2	1	2	0	3	3	0	0.2248
2	2	1	2	1	2	3	0	0.1606
2	3	0	2	0	2	1	2	1.2180
2	3	0	2	0	2	2	1	1.1631
3	2	2	0	2	0	1	2	0.9476
3	2	2	0	2	0	2	1	1.0257
3	2	2	0	0	3	1	2	0.5617
3	2	2	0	0	3	2	1	0.6398
3	2	0	3	2	0	1	2	0.6761
3	2	0	3	2	0	2	1	0.7542
3	2	0	3	0	3	1	2	0.2902
3	2	0	3	0	3	2	1	0.3684
3	3	1	1	1	1	2	0	1.0041
1	4	2	1	2	1	2	1	-0.0604
2	4	1	1	1	1	1	1	0.9000
2	4	1	1	1	1	2	1	1.0713
4	2	1	1	1	1	1	1	0.8570
4	2	1	1	1	1	1	2	1.0158

TABLE V. NUMBER OF REPAIRS AND COST FOR IMMEDIATE REPAIR RULE

Number of Motor A	Number of Motor B	Operating Condition			Cost
		1 (Operating 8217.1 Hours)	2 (Operating 8122.7 Hours)	3 (Operating 9693.6 Hours)	
		$MRep^{(1)}$	$MRep^{(2)}$	$MRep^{(3)}$	
0	5	3.3315	3.3048	2.4736	1000
1	4	3.2827	3.2567	2.4469	1020
2	3	3.2336	3.2082	2.4202	1040
3	2	3.1841	3.1594	2.3932	1060
4	1	3.1343	3.1102	2.3660	1080
5	0	3.0841	3.0607	2.3387	1100
0	6	1.6579	1.6647	1.5747	1200
1	5	1.6359	1.6430	1.5629	1220
2	4	1.6138	1.6212	1.5510	1240
3	3	1.5916	1.5993	1.5391	1260
4	2	1.5694	1.5774	1.5271	1280
5	1	1.5471	1.5555	1.5152	1300
6	0	1.5247	1.5335	1.5032	1320

 ISBN: 978-1-61208-698-9

FITness Assessment
Hardware Algorithm Safety Validation

Andreas Strasser, Philipp Stelzer, Christian Steger
Graz University of Technology
Graz, Austria
Email: {strasser, stelzer, steger}@tugraz.at

Norbert Druml
Infineon Technologies Austria AG
Graz, Austria
Email: norbert.druml@infineon.com

***Abstract*—Error Correction Codes (ECC) are important safety methods for digital data to gain control of Single Event Upsets (SEU) in integrated digital circuits. SEU are responsible for single bit flips inside a digital circuit caused by ionizing radiation. This effect does not affect the physical structure of the components but the correctness of data inside flip flops. Consequently, data gets corrupted and the correct program flow gets disturbed. This effect needs to be considered especially for safety-critical systems. In the new ISO 26262 2nd Edition, the automotive domain suggests controlling SEU effects by algorithms that correct Single Bit Errors and Detect Double Bit Errors (SEC-DED). This raises the question what kind of impact Double Bit Error Correction (DEC) will have on the overall safety level for LiDAR (Light Detection and Ranging) systems. In this publication, we determine the difference between two ECC algorithms from a safety point of view: Hamming's code (SEC-DED) and Bose–Chaudhuri–Hocquenghem-Code (DEC). For this purpose, we developed a novel method for algorithm safety validation and applied it to both algorithms.**

Keywords–Safety Validation FPGA, Failure-in-Time Analysis FPGA, Error Correction Codes, ISO 26262 2nd Edition, Algorithm Validation.

I. INTRODUCTION

Fully autonomous driving will change our society, as well as individuals's daily routines and will improve overall road safety. To achieve the goal of autonomous driving, novel Advanced Driver-Assistance Systems (ADAS) are necessary. The two best-known ADAS are the Electronic Stability Control and the Anti-Lock Braking System, especially for their positive effect on active safety. Moreover, in the last years, a new generation of ADAS such as the Adaptive Cruise Control (ACC) has been established in middle class cars to avoid collisions. The next big step is introducing a comprehensive system enabling the perception of urban environment, which is one of the main goals of the PRYSTINE project [1].

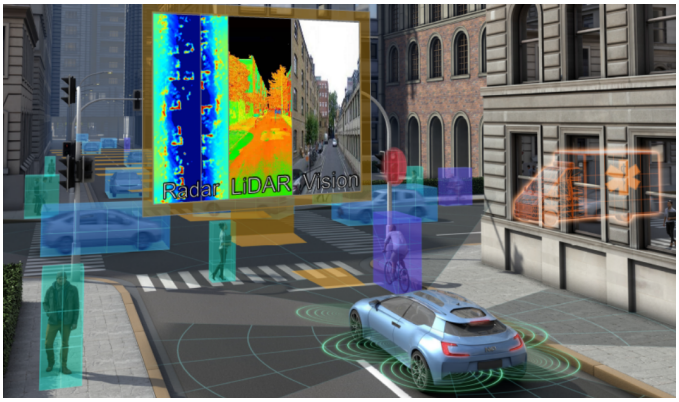


Figure 1. PRYSTINE's concept view of a fail-operational urban surround perception system [1].

PRYSTINE stands for Programmable Systems for Intelligence in Automobiles and is based on robust Radar and LiDAR sensor fusion to enable safe automated driving in urban and rural environments, as seen in Figure 1. These devices must be reliable, safe and fail-operational to handle safety-critical situations independently [1]. In contrast to Radar, LiDAR has not been implemented in middle class cars yet but there are basic approaches in the automotive industry such as the 1D MEMS Micro-Scanning LiDAR system as seen in Figure 2 [2]. This modern LiDAR system consists of an emitter and receiver path. The emitter path contains the Microelectromechanical systems (MEMS) mirror and the MEMS Driver Application-specific integrated circuit (ASIC). Druml et al. [2] indicate that the MEMS Driver and its precision of sensing, actuation and control directly influence the complete LiDAR system's measurement accuracy. Consequently, the LiDAR system's control-related digital circuits need to be correct and fault-tolerant. Fault-tolerant digital circuits struggle mainly with random hardware faults like Single Event Upsets which are soft errors in semiconductor devices induced by ionizing radiation [3]. These events do not physically harm the semiconductor components but may alter the logical value of a flip flop [4]. These errors have been affecting digital integrated circuits for decades and therefore, Error Correction Codes (ECC) are used for safety-critical systems [5]. ECCs are self-repairing algorithms with the ability to correct certain bit errors and maintain data correction during runtime [6]. The effect of SEU exponentially increases with higher packaging density as less electrons are representing a logic value [4]. As the demand for semiconductor devices rises due to ADAS, packaging density needs to increase even faster to satisfy computation power for real-time video signal processing [7]. Nevertheless, this trend also introduces drawbacks, especially from a safety point of view, as the enhancement of packaging density also increases the sensitivity to SEU [4]. Consequently, the automotive industry needs regulations and standards for safety-related semiconductor devices. For safety-related electrical and electronic devices, the automotive industry considers the functional safety ISO 26262 standard. In nine normative parts, this standard

 ISBN: 978-1-61208-698-9

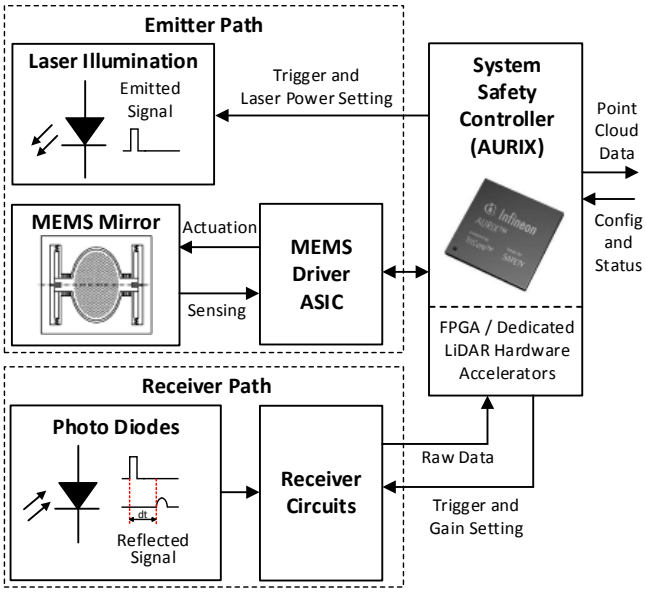


Figure 2. Overview of a LiDAR system for automonomous driving [2].

describes best practices to support engineers and managers in developing fail-safe automotive parts [8]. In the last years, this standard has been extended and the new version will be released end of 2018. The new version is called ISO 26262 2nd Edition and will include a part for semiconductors describing functional safety concepts for semiconductor devices [9]. For soft error mitigation, the standard suggests the use of Single Error Correction and Double Error Detection algorithms to protect digital circuits [9]. For semiconductor devices SEC-DED was already used in 1984 [5]. At that time, semiconductor devices were not that highly integrated and the packaging density was not as high as nowadays. Already in 1984, Chen et al. [5] described that in future semiconductor devices will use more complex ECC algorithms such as Double Error Correction and Triple Error Detection (DEC-TED). Contrary to the prediction of Chen et al. [5], the automotive industry still suggests using SEC-DED ECC algorithms 34 years later. This raises the question whether there are any disadvantages on DEC-TED algorithms or if the SEC-DED still fulfills the requirements for fail-safe automotive systems.

For this purpose, we will elaborate on the following two research questions:

- How can different ECC algorithms be validated from a safety point of view?
- Are Double Error Correction algorithms for LiDAR systems safer than SEC-DED algorithms?

II. Related Work

The need for error correction has always been vital for digital semiconductor devices due to possible alterations of flip flops caused by SEU. Already in 1984, Chen et al. described the application of these codes for semiconductor memory applications [5]. However, the history of ECC already began with punched card read errors in 1950. In this year, Hamming introduced his new approach for an automatic Error Correction Code during run-time to solve read errors [10]. Hamming's code is widely known and used for ECC. The algorithm corrects Single Bit Errors and is able to Detect Double Bit Errors (SEC-DED) by adding an additional parity bit [11]. For correcting more bits, other ECC algorithms are necessary. One of them is the concept of Bose-Chaudhuri-Hocquenghem-Codes (BCH-Codes). BCH-Codes can be used for multiple bit error corrections [12]. These two algorithms are the most important ECC concepts for digital integrated circuits and were already described by Chen et al. in 1984 [5]. Even modern and highly integrated complex systems still make use of Hamming's code and BCH-code [13] [14]. The novel ISO 26262 2nd Edition still refers to Hamming's ECC code to accomplish fail-safe digital circuits.

In the automotive industry, the ISO 26262 standard is used for functional safety. The new version ISO 26262 2nd Edition suggests ECC for diagnosing memory failures and rates the resulting diagnosis coverage as high. Therefore, this measure is often used for safety critical digital components [9] [13] [14]. For ECC, the standard still suggests the use of SEC-DED algorithms such as the Hamming code [9]. This raises the question whether SEC-DED has any advantages over DEC algorithms or vice versa. Still, novel safety critical automotive approaches, such as the fault-tolerant cache system for an automotive vision processor from Han et al. use SEC-DED [14].

The validation of algorithms is an important method for achieving certain requirements such as area, power dissipation or run time. Therefore, there are numerous articles about enhancing efficiency of fault-tolerant mechanisms through algorithm substitution [15] [16] [17]. Rossi et al. analyze the power consumption of fault-tolerant busses by comparing different Hamming code implementations with their novel Dual Rail coding scheme [15]. Also, Nayak et al. emphasize the low power dissipation of their novel Hamming code components [16]. Another example is the work of Shao et al. about power dissipation comparison between the novel adaptive pre-proccesing approach for convolutional codes of Viterbi decoders with conventional decoders [17]. Khezripour et al. provide another example for validating different fault-tolerant multi processor architectures by power dissipation [18]. Unfortunately, power dissipation is just one factor for reliability of safety-critical components and insufficient for safety validation. The most important indicator for safety at hardware level is the component reliability, which is measured in failure in time (FIT) rates [9]. Component reliability is the main indicator for safe hardware components and describes the quantity of failures in a specific time interval, mostly one billion hours [9]. These values can be calculated by specific standards for electronic component reliability such as the IEC TR 62380 [19] or statistically collected by field tests. Oftentimes, these field test have already been conducted by the manufacturers and are compiled in specific datasheets for component reliability [20]. For each component, the datasheets usually contain the specific FIT Rate for a certain temperature. To determine the FIT Rate for other temperatures, the Arrhenius equation as seen in (1) can be used.

$$DF = e^{\frac{E_a}{k} \cdot (\frac{1}{T_{use}} - \frac{1}{T_{stress}}))} \quad (1)$$

where:

DF	is Derating Factor
E_a	is Activation Energy in eV
k	is Boltzmann Constant (8.167303 x 10^{-5} ev/K)
T_{use}	is Use Junction Temperature in K
T_{stress}	is Stress Junction Temperature in K

The Arrhenius Equation requires the Junction Temperature instead of Temperature values. The Junction Temperature represents the highest operation temperature of the semiconductor and considers the Ambient Temperature, Thermal Resistance of the package as well as the Power Dissipation as seen in (2).

$$T_j = T_{amb} + P_{dis} \cdot \theta_{ja} \tag{2}$$

where:

T_{amb}	is Ambiant Temperature
P_{dis}	is Power Dissipation
θ_{ja}	is Package Thermal Resistance Value

The validation of ECC algorithms is crucial for designers to pick the optimal ECC. Rossi et al. analyzed SEC-DED and DEC codes on area overhead and cache memory access time but their work did not consider the impact of different ECC algorithms from a safety point of view [21]. For designers of safety-critical digital circuits, it would be helpful to be able to pick the most safe ECC with the advantage of lower FIT Rates. Especially for automotive Tier-1 companies lower FIT Rates imply higher component reliability which is crucial for the economic success or failure of the whole system as profit margins are that small that every defect matters. Therefore, to support designers of safety-critical digital circuits, this paper's contributions to existing research are:

1) Developing a novel method for safety validation of algorithms on Field Programmable Gate Array that is based on the approved ISO 26262 2nd Edition methods.
2) Applying the novel method to quantify the differences between SEC-DED and DEC from a safety point of view.
3) Recommendation of ECC algorithm for safety-critical automotive LiDAR systems, based on the novel method of this paper.

III. FITNESS ASSESSMENT

To validate different ECC algorithms, it is necessary to quantify the essential values. Based on the functional safety standard ISO 26262 2nd Edition's approved methods, the FIT Rate is the most important factor for safety-critical hardware components. As stated in the Related Work section II, the Derating Factor influences the FIT Rate and is expressed in the Arrhenius equation (1). Combined with the Temperature Junction equation it is obvious that the power dissipation is the most significant quantity that can be influenced by designers of digital circuits (see (3)).

$$DF = e^{\frac{E_a}{k} \cdot (\frac{1}{T_{use}} - \frac{1}{T_{amb} + P_{dis} \cdot \theta_{ja}}))} \tag{3}$$

Consequently, by decreasing Power Dissipation the designer increases component reliability. For Field Programmable Gate Array (FPGA), the power dissipation primarily depends on static and dynamic power consumption. Based on these physical principles, our novel method FITness Assessment for algorithm safety validation on FPGAs is segmented in the following parts, as seen in Figure 3:

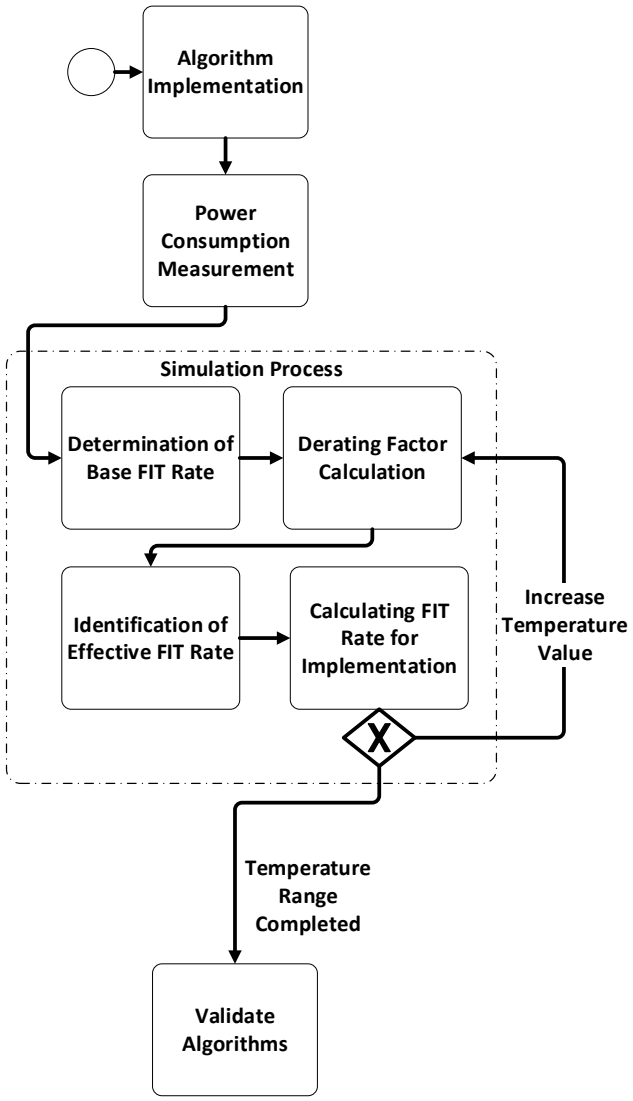


Figure 3. Workflow overview of our novel method FITness Assessment for algorithm validation from a safety point of view in Business Process Model and Notation.

1) **Algorithm Implementation**
To guarantee similar conditions for different algorithms, it is necessary to implement a generic framework that allows implementing algorithms without major changes.
2) **Power Consumption Measurement**
For each algorithm, a particular measurement is recorded. It is advisable to record the generic framework without any algorithm to be able to determine the algorithms' power consumption by subtraction.
3) **Determinination of Base FIT Rate**
The Base FIT Rate may be calculated by using the IEC TR 62380 [19] standard or analyzed statistically by field tests. Oftentimes, these field test have already been conducted by the manufacturers and are compiled in specific datasheets for component reliability.
4) **Derating Factor Calculation**
The Derating Factor can be calculated with the Arrhenius equation and the related Thermal Junction equation as seen in (1) and (2).

5) **Identification of Effective FIT Rate**
The Effective FIT Rate reflects the Base FIT Rate for a specific temperature and can be calculated with:

$$FIT_{ef} = FIT_{base} \cdot DF \quad (4)$$

where:
- FIT_{base} is Base FIT Rate from FPGA Reliability Datasheet
- DF is Derating Factor as seen in (1)

6) **Calculating FIT Rate of the Implementation**
The Effective FIT Rate as seen in (4) represents the component reliability for the whole FPGA. However, an FPGA is made up of many different logic elements. Consequently, the Effective FIT Rate can be broken down into the amount used by each logical element as seen in (5).

$$FIT_{imp} = \frac{FIT_{ef}}{N_{le}} \quad (5)$$

where:
- FIT_{ef} is Effective FIT Rate as seen in (4)
- N_{le} is Total Number of Logic Elements of the specific FPGA taken out from Datasheet

7) **Validate Algorithms**
The resulting FIT Rate of the implementation represents the FIT Rate of the specific algorithm and can be used for validation. It is adviseable to measure each algorithm once at room temperature conditions and simulate the rest of the temperature range by starting with the Derating Factor Calculation.

IV. Test Setup

In our research question, we analyze the differences between SEC-DED and DEC. For this purpose, we chose the Hamming code for SEC-DED as this code is recommended in the new ISO 26262 2nd Edition and the BCH-code for DEC, especially because other ECC algorithms are often based on this concept and both algorithms fulfil the following requirements:

- 32 Bit data size
- Combinatorical Logic
- Including Fault Injection Module
- SEC-DED or DEC Functionality

The generic algorithm framework contains a testbench with an automatic up-counter as well as a validator (see Figure 5). Both algorithms can be exchanged in the framework without any major changes. This enables a precise validation from a safety point of view.

In our test setup, we use the MAX1000 - IoT Maker Board by Trenz Electronic. This device is a small maker board for prototyping with sparse additional components. The main controller is the MAX10 10M08SAU169C8G, an FPGA device by Intel. For our research, the main advantages of using this board are:

- Small amount of additional hardware components
- Availability of Reliability Datasheet

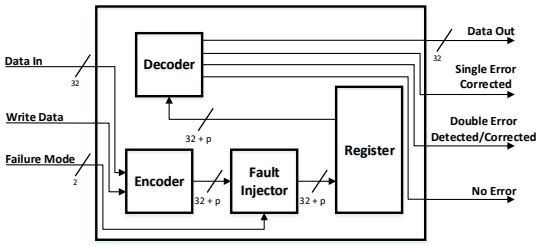


Figure 4. Pin configuration of both algorithms including an overview of functional blocks inside.

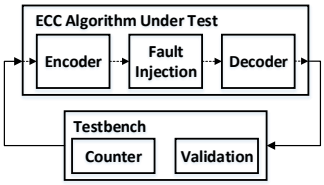


Figure 5. General framework for ECC algorithm validation including testbench and ECC algorithm.

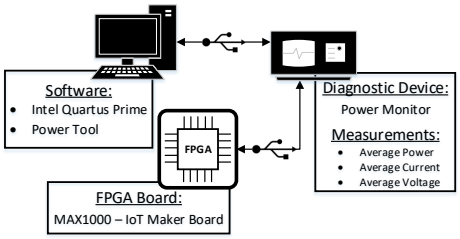


Figure 6. Overview of the entire measurement setup including software and hardware components.

This board also contains an FTDI chip that draws about 50 mA on average, which we will subtract out for our analysis. The power consumption measurement is performed by the Mobile Device Power Monitor of Monsoon Solutions. The big advantage of this power monitor is the direct measurement of USB devices. The entire measurement setup is shown in Figure 4 and 6 and contains the following software and hardware parts:

- Quartus Prime 18.0 (Intel)
- Power Tool 5.0.0.23 (Monsoon Solutions)
- Mobile Device Power Monitor (Monsoon Solutions)
- MAX1000 - IoT Maker Board (Trenz Electronic)

V. Results

This section summarizes our results of the comparison of SEC-DED and DEC ECC algorithm. The validation was performed with our novel FITness Assessment method for algorithm validation from a safety point of view as described in Section III.

The first algorithm we implemented was the Hamming code, which is a SEC-DED ECC algorithm. The implementation reserves 45 logic elements of the used FPGA and the whole board has an average power dissipation of 571.78 mW. With the second BCH-code DEC ECC algorithm, the board consumes an average of 599.05 mW and assignes 65 logic elements. The first result shows a difference between both algorithms in logic elements as well as in power dissipation resulting in a varying FIT Rate. The next step is the simulation

 ISBN: 978-1-61208-698-9

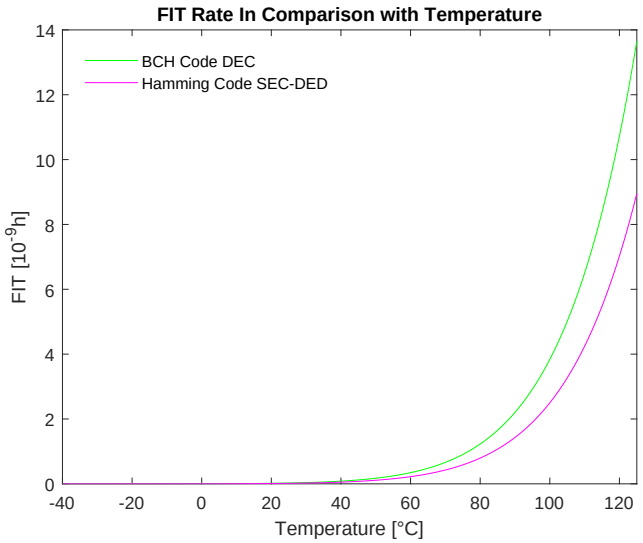


Figure 7. Simulation results of the resulted FIT Rates between -40°C and 125°C for both ECC implementations.

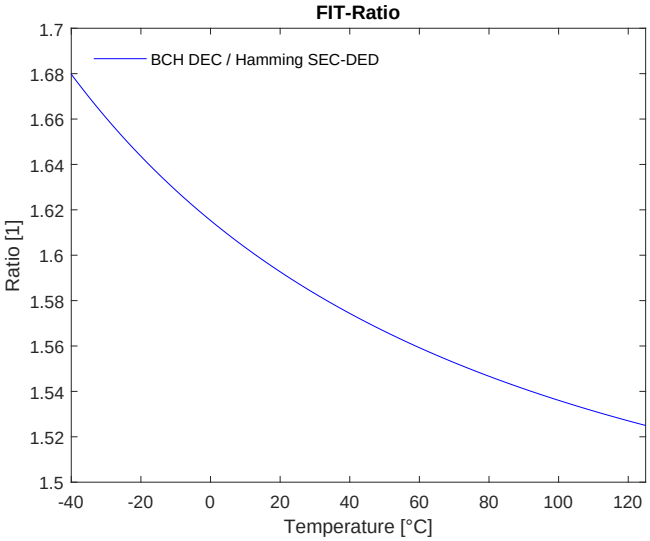


Figure 8. Overview of the FIT Rate overhead between SEC-DED and DEC ECC algorithm.

process over the whole temperature range. We selected a temperature range between -40°C and 125°C and the values of Table I were used for the simulation process. In our simulation we neglected the alteration of power dissipation through temperature because it would affect both ECC implementations evenly.

Figure 7 points out that both algorithms vary in their FIT Rate and rise exponentially with increasing temperature. The FIT Rate may be neglected for temperatures up to 40 °C. The Hamming code with SEC-DED shows a better FIT Rate indicating more reliability of the hardware components which results in a higher safety level. The reason for this difference is the greater number of logic elements used for the DEC ECC algorithm and the resulting increase of power dissipation. The higher power dissipation results in a higher Thermal Junction temperature as seen in (2) which leads to a higher FIT Rate.

Both algorithms were implemented without any safety measures. This means that any damage to the Logic Element of the FPGA leads to failure of the whole ECC algorithm and

TABLE I. RESULTS OF THE RESERVED LOGIC ELEMENTS AND AVERAGE TOTAL POWER DISSIPATION OF BOTH ECC IMPLEMENTATIONS.

	Hamming Code	BCH-Code
Used Logic Elements	45	65
Total Average Power Dissipation	571.78 mW	599.05 mW

the safe memory block. The ECC algorithm is the measure against SEU related altered flip flops inside the memory block which decreases the specific FIT Rate of the memory block. The results of Figure 7 do not represent the FIT Rates of the memory block but the FIT Rate of the pure ECC implementation. It is important to understand that the ability of more bit error correction is not considered for the algorithm validation because it only positively influences the FIT Rate of the memory block.

Moreover, it is important to understand that the absolute values of the FIT Rate always correlate to a specific FPGA. Consequently, it is advantageous to look at the ratio between the algorithms because this gives a better overview of the overhead. The SEC-DED/DEC ECC FIT Ratio is depicted in Figure 8. The FIT Ratio overhead of the DEC ECC algorithm is slighly decreasing with increasing temperature, which is negligible in practice.

We recommend using the Hamming code algorithm for SEC-DED error correction for 32 bit memory size registers in automotive LiDAR systems. The SEC-DED algorithm used in our experiment resulted in a FIT Rate that was at least 52% lower than the DEC ECC algorithm.

VI. Conclusion

In this paper we analyzed SEC-DED and DEC ECC algorithms from a safety perspective. In Section III, we introduced the FITness Assessment, a novel method for algorithm validation from a safety point of view. This method is based on approved methods of the novel automotive functional safety standard ISO 26262 2nd Edition. The result clearly shows that different algorithms lead to different FIT Rates. FITness Assessment allowed the measurement of each algorithm's specific FIT Rate, facilitating the selection of the most reliable ECC algorithm. Our case shows a DEC ECC algorithm that has a higher FIT Rate than the SEC-DED ECC algorithm. The FIT Rate reflects component reliability which is an important hardware indicator for safety.

The paper's findings demonstrate that algorithm validation from a safety point of view is possible and that different ECC algorithms also result in different FIT Rates. These differences should not be neglected from a safety as well as from a business point of view. The FIT Rate also statistically indicates the amount of defective components, which is an economically important indicator as lower FIT rates also result in less defect components. Our results also give an explanation why the automotive industry still suggests using SEC-DED ECC algorithms instead of DEC ECC algorithms as SEC-DED offers a lower FIT Rate than DEC. In our case, the difference in FIT Rate was at least 52% and consequently, we suggest using SEC-DED for LiDAR systems.

The automotive industry is disrupted by autonomous driving which is why fault-tolerance, safety and reliability will become increasingly important in the next years. Our novel method FITness Assessment enables the validation of different algorithms to be able to select the most reliable one, which

 ISBN: 978-1-61208-698-9

helps improve the overall safety level of the automotive vehicle by increasing component reliability.

VII. ACKNOWLEDGMENTS

The authors would like to thank all national funding authorities and the ECSEL Joint Undertaking, which funded the PRYSTINE project under the grant agreement number 783190.

PRYSTINE is funded by the Austrian Federal Ministry of Transport, Innovation and Technology (BMVIT) under the program "ICT of the Future" between May 2018 and April 2021 (grant number 865310). More information: https://iktderzukunft.at/en/.

REFERENCES

[1] N. Druml, G. Macher, M. Stolz, E. Armengaud, D. Watzenig, C. Steger, T. Herndl, A. Eckel, A. Ryabokon, A. Hoess, S. Kumar, G. Dimitrakopoulos, and H. Roedig, "Prystine - programmable systems for intelligence in automobiles," in 2018 21st Euromicro Conference on Digital System Design (DSD), Aug 2018, pp. 618–626.

[2] N. Druml, I. Maksymova, T. Thurner, D. Van Lierop, M. Hennecke, and A. Foroutan, "1D MEMS Micro-Scanning LiDAR," in Conference on Sensor Device Technologies and Applications (SENSORDEVICES), 09 2018.

[3] B. D. Sierawski, J. A. Pellish, R. A. Reed, R. D. Schrimpf, K. M. Warren, R. A. Weller, M. H. Mendenhall, J. D. Black, A. D. Tipton, M. A. Xapsos, R. C. Baumann, X. Deng, M. J. Campola, M. R. Friendlich, H. S. Kim, A. M. Phan, and C. M. Seidleck, "Impact of low-energy proton induced upsets on test methods and rate predictions," IEEE Transactions on Nuclear Science, vol. 56, no. 6, Dec 2009, pp. 3085–3092.

[4] R. Islam, "A highly reliable SEU hardened latch and high performance SEU hardened flip-flop," in Thirteenth International Symposium on Quality Electronic Design (ISQED), March 2012, pp. 347–352.

[5] C. L. Chen and M. Y. Hsiao, "Error-Correcting Codes for Semiconductor Memory Applications: A State-of-the-Art Review," IBM Journal of Research and Development, vol. 28, no. 2, March 1984, pp. 124–134.

[6] J. Singh and J. Singh, "A Comparative Study of Error Detection and Correction Coding Techniques," in 2012 Second International Conference on Advanced Computing Communication Technologies, Jan 2012, pp. 187–189.

[7] H. Shaheen, G. Boschi, G. Harutyunyan, and Y. Zorian, "Advanced ECC solution for automotive SoCs," in 2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS), July 2017, pp. 71–73.

[8] R. Mariani, "An overview of autonomous vehicles safety," in 2018 IEEE International Reliability Physics Symposium (IRPS), March 2018, pp. 6A.1–1–6A.1–6.

[9] I. n. E. ISO, "Draft 26262 2nd Edition: Road vehicles-Functional safety," International Standard ISO/FDIS, vol. 26262, 2018.

[10] R. W. Hamming, "Error detecting and error correcting codes," The Bell System Technical Journal, vol. 29, no. 2, April 1950, pp. 147–160.

[11] H. Liu, D. Kim, Y. Li, and A. Z. Jia, "On the separating redundancy of extended hamming codes," in 2015 IEEE International Symposium on Information Theory (ISIT), June 2015, pp. 2406–2410.

[12] Z. Xie, N. Li, and L. Li, "Design and Study on a New BCH Coding and Interleaving Techniques Based on ARM Chip," in 2008 4th IEEE International Conference on Circuits and Systems for Communications, May 2008, pp. 315–318.

[13] S. Sooraj, M. Manasy, and R. Bhakthavatchalu, "Fault tolerant FSM on FPGA using SEC-DED code algorithm," in 2017 International Conference on Technological Advancements in Power and Energy (TAP Energy), Dec 2017, pp. 1–6.

[14] J. Han, Y. Kwon, K. Byun, and H. Yoo, "A fault tolerant cache system of automotive vision processor complying with ISO26262," in 2016 IEEE International Symposium on Circuits and Systems (ISCAS), May 2016, pp. 2912–2912.

[15] D. Rossi, A. K. Nieuwland, S. V. E. S. van Dijk, R. P. Kleihorst, and C. Metra, "Power Consumption of Fault Tolerant Busses," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 16, no. 5, May 2008, pp. 542–553.

[16] V. S. P. Nayak, C. Madhulika, and U. Pravali, "Design of low power hamming code encoding, decoding and correcting circuits using reversible logic," in 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT), May 2017, pp. 778–781.

[17] W. Shao and L. Brackenbury, "Pre-processing of convolutional codes for reducing decoding power consumption," in 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, March 2008, pp. 2957–2960.

[18] H. Khezripour and S. Pourmozaffari, "Fault Tolerance and Power Consumption Analysis on Chip-Multi Processors Architectures," in 2012 Seventh International Conference on Availability, Reliability and Security, Aug 2012, pp. 301–306.

[19] T. IEC, "Iec 62380," Reliability data handbook–universal model for reliability prediction of electronics components, PCBs and equipment (emerged from UTEC 80-810 or RDF 2000), 2004.

[20] "Reliability Report," Jul 2018, [retrieved: 01, 2019]. [Online]. Available: https://www.intel.com/content/www/us/en/programmable/support/quality-and-reliability/reports-tools/reliability-report/rel-report.html

[21] D. Rossi, N. Timoncini, M. Spica, and C. Metra, "Error correcting code analysis for cache memory high reliability and performance," in 2011 Design, Automation Test in Europe, March 2011, pp. 1–6.

 ISBN: 978-1-61208-698-9

Does a Loss of Social Credibility Impact Robot Safety?

Balancing Social and Safety Behaviours of Assistive Robots

Catherine Menon

Adaptive Systems Research Group
School of Computer Science
University of Hertfordshire
Hatfield AL10 9AB, United Kingdom
Email: c.menon@herts.ac.uk

Patrick Holthaus

Adaptive Systems Research Group
School of Computer Science
University of Hertfordshire
Hatfield AL10 9AB, United Kingdom
Email: p.holthaus@herts.ac.uk

***Abstract*—This position paper discusses the safety-related functions performed by assistive robots and explores the relationship between trust and effective safety risk mitigation. We identify a measure of the robot's social effectiveness, termed social credibility, and present a discussion of how social credibility may be gained and lost. This paper's contribution is the identification of a link between social credibility and safety-related performance. Accordingly, we draw on analyses of existing systems to demonstrate how an assistive robot's safety-critical functionality can be impaired by a loss of social credibility. In addition, we present a discussion of some of the consequences of prioritising either safety-related functionality or social engagement. We propose the identification of a mixed-criticality scheduling algorithm in order to maximise both safety-related performance and social engagement.**

Keywords–Human-Robot interaction; Social credibility; Robot safety.

I. INTRODUCTION

Assistive robots offer significant benefits to an increasingly elderly population, both in terms of their social impact and their functionality [1][2]. Assistive robots support independent living by aiding humans to conduct basic activities, such as preparing food and bathing. Similarly, these robots may support the psychological health of elderly or isolated individuals via socially-important behaviours, providing companionship and encouraging these individuals to engage and interact.

There are safety implications to the use of assistive robots, both in terms of the physical hazards they present and in terms of the functionality they provide. An assistive robot will often act as mitigation for a safety risk, alerting the user to a hazardous situation and requesting that they take action.

In this paper we bring together concerns from the safety community and the robotics community. The social effects of autonomous systems are not typically factored into hazard analysis of these systems, and this paper aims to address that omission. Equally, from an Human-Robot Interaction (HRI) perspective the ways in which the social performance of an assistive robot are affected by safety features (e.g., automatic stops, avoidance of physical contact) is not always explicitly considered. Bringing these concerns together within a single domain provides the research community with a foundation for discussing how to assure the safety of an autonomous system which must also perform another (social) function. This is relevant not only to assistive robots but also to autonomous vehicles, medical devices and companion robots.

Figure 1. The assistive robot Care-O-Bot 4 configured with two arms and spherical hip and head joints.

To meet this aim we examine how both the safety-critical and socially important behaviours of an assitive robot rely on the user's engagement with the robot. User engagement, particularly in safety-critical situations, is partially determined by the *social credibility* of the robot, or how well it follows social norms relevant to its environment. In Section II we present a case study assistive robot, identifying some of its socially important behaviours. Section III looks at the functional and physical hazards associated with such a robot, while Section IV considers restrictions on the behaviours considered socially appropriate, as well as introducing and defining the concept of social credibility. In Section V we identify how a loss of social credibility impacts both safety-critical and socially-important types of behaviour, illustrate how such behaviours may be in conflict with each other and discuss a solution which allows both to be prioritised. Section VI contains a proposal for future work to validate these concepts and solution, summarizes our position and concludes our contribution.

 ISBN: 978-1-61208-698-9

II. THE CARE-O-BOT ASSISTIVE ROBOT

The Care-O-Bot [3] is an up-to-date example of a mobile assistant robot with the capacity for social interaction. Its most recent iteration can be adapted to various applications in care due to its modular design. When equipped with two 7-DoF arms and two spherical joints at its hip and head (as shown in Figure 1), it can manipulate objects within an exceptionally large workspace. When such a robot operates within a sensorized domestic environment [4], it is able to support humans in their daily activities. In conjunction with its interactive capabilities the robot therefore is well suited to execute a wide range of desirable tasks in elderly care [5].

In such a setting, the Care-O-Bot might be typically expected to perform a range of functions including:

- Accepting and handling a parcel at the front door
- Reminding a user to take their medication
- Assisting a user to carry food items from the kitchen

In addition, more complex temporal behaviours [6] can also be defined by a formal or informal care-giver. These behaviours may include requesting the robot to alert a care-giver if the user has remained in bed for longer than a specified time, or alerting a user if the oven has remained on after cooking a meal. Existing research has utilised formal verification [7] to ensure that user-defined behaviours do not conflict with each other, and has highlighted a need for human-intelligible output to help users define behaviours. In addition to these care-giving behaviours, the Care-O-Bot would typically be expected to encourage the user to engage and interact by offering entertainment and companionship.

III. SAFETY CRITICAL PERFORMANCE OF ASSISTIVE ROBOTS

Some of the functions performed by an assistive robot such as the Care-O-Bot have the potential to impact safety. The robot presents both physical hazards (e.g., its weight can contribute to crush injuries) as well as functional hazards. Functional hazards are those resulting from its behaviour: the robot may fail to perform a safety-critical function (e.g., reminding a user to take medication) or may perform this function incorrectly (e.g., reminding the user too frequently).

The Care-O-Bot has been designed with safety as a priority. All personal care and assistive robots are required to comply with safety standards [8], as well as broader UK safety legislation [9]. The Care-O-Bot accordingly contains a number of features to reduce or eliminate collisions with a user [10]. The robot's base is equipped with three laser range sensors with a safety Programmable Logic Controller (PLC) that allow for a 360 degree obstacle recognition at ankle height. Its joints are protected with two separate safe-torque-off (STO) switches at base and torso. The STOs are either triggered by the laser range sensors, one of two emergency buttons at the robot's front and back (Figure 1), or a wireless emergency stop. Furthermore, the robot's autonomous navigation software implements well established collision avoidance mechanisms [11] by default. Despite this, however, there is a lack of sensors at the arm joint and thus no mitigation against crush injuries received at this site. As a result, Care-o-bot requires constant monitoring while participating in interactions with humans that involve the robot's arms.

A. System failure and resultant hazards

System failure still remains an issue for the Care-O-Bot, as for all safety-critical systems. Should the proximity sensors fail, the Care-O-Bot could collide with a user and cause injury. Other potential hazards include hot surfaces from the engine, trip hazards from the wheels, potential corrosive substances and the presence of electrical items. Furthermore, collision hazards are not limited only to collision with the robot itself, but include collisions with any objects it is holding. In particular, a key characteristic of the Care-O-Bot is the presence of arms that can be used to carry hot liquids on a tray [6]. Should a system failure occur, the arm may be stuck in an unpredictable position, resulting in anything held being spilt on the floor or on a user. It is clear, therefore, that complete or partial system failure of the Care-O-Bot or similar assistive robot should be treated as a serious occurrence, both in terms of the risks presented by inherent characteristics of the robot and the risks presented by the environmental situation at the time of failure.

B. Functional hazards

Software failure is a primary cause of functional hazards in the Care-O-Bot, as it can result in behaviours being carried out incorrectly or not at all. Software failure has been extensively studied in complex systems [12], and methods for assessing the contribution of development techniques to safety [13] are common across multiple domains. In addition, existing research has examined the correlation between failure rate estimates and verification performed [14].

However, a significant complexity for assistive robots such as the Care-O-Bot is the ability for end-users to define their own desired robot behaviours. Because of this, it cannot be assumed that the safety-critical behaviours of an assistive robot are known at the time of deployment. Notwithstanding verification such as [7], there is the potential for an inexperienced end-user to define behaviours which impact safety, or which put the robot in a position which can violate assumptions about the constraints it will obey. For example, an inexperienced user may define a behaviour which causes the robot to remind them to take their medication at an incorrect period or frequency. Equally, a user may define a behaviour which causes the robot to remain in another room, thus compromising its availability to perform those safety-critical functions which rely on direct observation of the user.

As with all systems, there is a UK legal requirement that the risk posed by assistive robots should be reduced As Low As Reasonably Practicable (ALARP) [9]. This requires hazards to be identified, risks to be estimated and mitigation to be put into place where needed to reduce the system risk to a tolerable level. In the case of assistive robots, the robot itself is typically taking a monitoring role and acting as partial mitigation for a wider risk. For example, a robot programmed to remind the user to take medication is partially mitigating against the illness which will result from a lack of medication. Similarly, a robot programmed to notify the user if the oven has been left on is partially mitigating against the risk of fire.

In each of these cases the user is required to take action to complete the mitigation (take the medication, switch off the oven, or evacuate the home). This is an effect of the fundamental design principles of the robot, driven by the need to prioritise *reablement*[2]. Reablement is defined as the drive to "Support people to do rather than doing to / for people"

 ISBN: 978-1-61208-698-9

[15] and is an important characteristic for service and assistive robots. Designing with reablement in mind means that the assistive robot is not intended to carry out the tasks itself (e.g., administering medicine to a user), but is instead intended to encourage the user to complete the task themselves. A side-effect of this design principle is that an assistive robot will typically require human engagement in order to successfully mitigate safety risks by completing the necessary action. One of the most important aspects of safety-critical assistive robot performance is therefore determined by the extent to which end-users engage with the robot.

IV. Socially appropriate behaviours

Because assistive and service robots are used within a domestic environment, it is important that the behaviour they display is both empathic and socially interactive [5]. Specifically, the behaviour a robot exhibits must be appropriate to the social role that it is expected to fulfil [16]. The extent to which a robot exhibits socially appropriate or socially intelligent behaviour is characterised by a number of factors, including its ability to establish and maintain social relationships, use natural cues, and express and perceive emotions [17].

Much existing work has explored the viability of transfering models of human interaction to robots, including an examination of adequate interaction distances and orientations [16], [18], [19]. Pursuing a complementary approach, research into the Care-O-Bot [6] has also exploited many techniques from the "learning by following" model [20]. Under this model the robot learns desired behaviours from following, observing and interacting with the human. The robot also conveys its capabilities and intentions using social signals [21] that might involve using whole-body or arm movements.

The social norms relevant to the robot will vary with its environment and operational use. Some may be generalised to a certain extent, modulo cultural differences. It is likely that there are certain situations in which it would be inappropriate for the robot to follow the user or capture their attention. For example, the human may have expectations of privacy which would be violated by the robot following them into the bathroom or bedroom [22][23]. Similarly, the human may have the social expectation that when they're engaged in a particular task (e.g., conducting a conversation), that the robot will not interrupt. Other social norms relevant to a domestic environment include detecting and adapting to a user's personal space, involving the user in decisions about entertainment and companionship and respecting the user's autonomy. Social norms will vary depending on the level of care required by the user, the degree of autonomy they expect, their age and personal preferences for interaction, as well as existing wider cultural and social constraints.

A. Social credibility

In this paper we extend the notion of socially appropriate behaviour to encompass the concept of *social credibility*. The social credibility of a domestic robot is a measure of how well it obeys the social norms relevant to its environment.

Social credibility helps determine the extent to which a human considers the robot to be a functioning social being. Work in [24] demonstrates a link between social intelligence and consideration of the robot as a social being. Further experiments have reinforced this tendency of humans to treat a socially intelligent or emotionally empathic robot as a social being, even to the extent of exhibiting concern over "hurting its feelings" [25]. This is amplified in a domestic or home setting, with end-users asked to rank the utility of cleaning robots considering their emotional impact as well as their functionality [26].

Social credibility has both a static and dynamic element. The static element refers to design: Has this robot been designed to follow social norms? Are its behaviours consistent with its appearance so that both match a potential user's expectations [21]? Static social credibility is also achieved via constraints embedded within the robot's programming (e.g., "do not follow a human into the bathroom").

Dynamic social credibility refers to the ongoing adaptability of the robot's behaviour: is it capable of adjusting its own behaviours based on feedback and the observed environment? Dynamic social credibility allows for evolution of the social norms over time. For example, it may be within norms for a domestic robot to follow a child user into a bedroom, but not for it to similarly follow an adult user. As a child user ages, dynamic social credibility ensures that the robot's behaviour reflects the changing application of the norm.

Social credibility is an evolving measure, and dependent on the actions of the robot. Much as a system which does nothing is "perfectly safe", a robot which is turned off and hence never takes an action will not lose nor gain social credibility. Social credibility may be temporarily lost by an inappropriate action, and gained back by subsequent actions.

As discussed in Section IV, social norms will vary with the environment. For a domestic service or assistive robot, we consider the following contributors (both positive and negative) to social credibility:

- Frequency and urgency of interruptions
- Nature and intensity of interaction, engagement and interruption
- Responsiveness of the robot to verbal and non-verbal feedback
- Appropriate physical movement and distance maintained from end-user
- Trust inspired by the robot in the end-user
- Understanding communicated by the robot as to its capabilities

It is important to note that although trust is a significant aspect of social credibility for an assistive robot, it is not the only factor. Much work already exists on the questions of eliciting and maintaining trust (see [27] for an overview, additionally [28][29]), with considerations of factors such as reliability, predictability, physical presence and emotional response.

However, it is possible for a robot to inspire trust and emotionally engage a user without necessarily having a high degree of social credibility. For example, a pet-like robot [30] may emotionally engage a user because of its appearance and actions, but there are typically fewer social norms applicable to a pet. Similarly, an autonomous vehicle or an alarm system may be trusted by its end users without any imputation of sociability or social knowledge. By contrast, a robot which shares personal information about its user with a third party will typically be regarded as untrustworthy [31], but such sharing does not in

 ISBN: 978-1-61208-698-9

itself mean the robot is not seen as a social being (a malicious person may have also done the same).

Crucially, social credibility also requires that the user understand the robot's capabilities, much as they would understand the different capabilities of a human adult or a human child. A high degree of social credibility implies that a robot has communicated an understanding as to its capabilities and reduces the potential for over-trust [32]. From the perspective of safety, over-trust is considered a negative factor as it leads to excessive reliance on the automation even when there are indications of system failure.

V. Social credibility and safety-critical systems

A large part of the duties of an assistive robot involve reminding or prompting the end-user to take action. This involves some form of interruption to the user's current activity. Two important social norms for these robots are therefore around the frequency of interruptions and on the way these interruptions are made. In [33], users explicitly identify that reminders given by an assistive robot become irritating under the following circumstances:

- When repeated often
- When repeated in a "mechanical" voice
- When repeated at inopportune times, interrupting the user

Conversely, some behaviours and methods of interruption are viewed positively by users and considered to mimic human interruptions, as discussed in [19] and [34]. These include the use of direct, random and non-random gaze directions to signal the beginning of an interaction. Other studies [35][36] have examined users' preference for personal space from robots, identifying that users perception of their personal space diminishes for likeable robots, and similarly that robots which encroach on this space are regarded as unlikeable, threatening or irritating. Personal space preferences will vary with context; for example, users are typically reluctant to accept a robot following them into the bathroom [22].

Inappropriate interruptions therefore present a potential for a loss of social credibility. A robot whose interruptions take no account of social norms is more likely to be regarded as a simple mechanical system (e.g., an alarm or reminder application) instead of as another social entity. For example, an assistive robot which always sounds an alarm at a certain time to remind the user to take medication is performing a role no more complex than an alarm clock, and hence complying with no relevant social norms. As such, it does not build social credibility in the same way that an assistive robot would if its interruptions were sensitive to the users' environment, engagement and current activities ([19]). As social credibility is a dynamic concept (see Section IV-A), a robot which has already built social credibility by demonstrating such sensitivities is vulnerable to losing this credibility if its interruptions become inappropriate.

A loss of social credibility (from any cause) can lead to an end-user disengaging with the robot in a number of ways. Firstly, the user may simply switch the robot off. Studies have shown that users are reluctant to switch off robots they consider to be intelligent [37], or perceived social beings. However, once social credibility is lost, this "protective" aspect is lost with it. Users are much more willing to switch off a robot considered to be solely a *robotic device*, particularly when the mode of engagement with this robot becomes arduous. In [38], drivers concluded that they would prefer to be able to turn off a speed warning system that was judged "irritating", even where they agreed that use of the technology would be helpful.

Secondly, even where the user permits the robot to remain switched on, they may start to ignore the suggestions and prompts made by the robot. This then leads to a dilemma for those designing such robots: if repeated interruptions lower social credibility, then how should the robot deal with an urgent prompt that has been ignored?

A. Safety-critical systems

Any disengagement with an assistive robot (whether switching it off or ignoring its prompts) compromises its ability to perform its safety-critical functions. It is clear that switching a robot off renders it incapable of providing any alerts or reminders. Similarly, because assistive robots mitigate risk by prompting end-user action (see Section III), any user disengagement means that the risk mitigation is not carried out in full. For example, a robot reminding the user that the oven has been left on has no effect unless the user engages with the robot, and returns to switch the oven off.

Furthermore extrapolating from studies performed in other domains has enabled us to identify a unique user reaction that may result from loss of social credibility, and which affects only safety-critical actions of the robot (as opposed to routine actions). In more detail, safety-critical situations are the exception, not the rule, and hence any alert or reminder in such a situation will be perceived by the user as "not the expected behaviour". In the aviation domain, where autonomous cockpit systems are not considered to be social entities, pilots have been observed to attempt to debug the automation when its actions deviate from those they expected. In a study of cockpit automation [39] established the tendency in pilots to monitor automation status via the flight control unit (FCU), which shows *commanded* targets, paths and modes, rather than via the display showing *actual* targets, paths and modes being executed by the automation.

Given that this observation took place in a highly-trained cohort of pilots, it is reasonable to say that untrained end-users of an assistive robot may also display the same mode confusion. This would lead to a situation in which a user is alerted to a hazard and instead of taking mitigating action attempts to debug or force the assistive robot to return to the "expected" behaviour.

B. Prioritising safety-criticality

The performance of safety critical behaviours is a clear priority from a legal and regulatory viewpoint [9][8] (for other priorities, see V-C). Motivated by this, we have identified a number of potential methods to address loss of safety-critical functionality resulting from lowered social credibility. Each of these methods trades a slight decrease in the robot's overall capability in return for maintaining an adequate level of social credibility. Since social credibility is a requirement for effective safety critical performance, this corresponds to decreasing the robot's capabilities in order to gain confidence that safety-critical engagements will be performed effectively when needed.

 ISBN: 978-1-61208-698-9

The first method we describe is an attempt by the robot to alter its behaviour when the social credibility drops below a threshold value which we will term the *disengagement threshold*. The disengagement threshold is the level of social credibility at which engagement with the robot (including its future safety-critical behaviours) is jeopardised. When this threshold is being approached, the robot should choose to alter the nature of its alerts and reminders to stop social credibility loss.

Both [19] and [34] identify a number of methods whereby a robot may interrupt an end-user, based on non-verbal behavioural cues. The extent and urgency of the interruption can be tailored to its nature: a safety-critical behaviour may still merit an urgent (and socially inappropriate) interruption even when the robot's social credibility is at risk of dropping below the threshold. However, for less critical interruptions the robot may choose to utilise any of the following behaviours:

- Slow its physical movements when coming to interrupt a user
- Decrease the volume of any audible alerts
- Display visual alerts (e.g., on the attached screen, for the Care-O-Bot), instead of audible alerts
- Approach the user and wait for the user to initiate an interaction

In addition to altering the nature of its alerts and interruptions, the robot may also choose to alter the frequency of these when approaching the disengagement threshold. Interruptions are a cognitive challenge for a user, and existing work shows that in some situations user satisfaction is maximised by delaying an interruption at the cost of some awareness [40].

This proposal allows a robot to *delay* a routine behaviour (such as interrupting the user with the offer of food or drink) in order to retain sufficient social credibility to ensure that any safety-critical behaviour (such as notification the oven is on) will be engaged with by the user. Other routine behaviours a robot may choose to suspend or delay if its social credibility is low include: greeting the user, engaging in social interaction and conversation, reminding the user of appointments and offering the user entertainment.

C. Prioritising social credibility

However, safety-critical performance is not the only consideration for assistive robots. It is also imperative that these robots perform their social functionality adequately. There is the potential for prioritisation of functionality relating to safety (e.g., requiring the robot to follow the user through the house in case of a fall) to result in the neglect of other socially important behaviours such as greeting, user engagement and user interaction. In other words, a robot performing only safety-related behaviours may not be free to perform other roles which are critical to its reablement functionality.

Furthermore, the performance of the safety-critical behaviours can itself lead to a loss of social credibility. A robot alerting the user to a fire may out of necessity do so at an inopportune time or in an urgent or disruptive fashion. The nature of such (intense, potentially ill-timed) alerts means that they will result in a certain loss of social credibility. This has the potential to drive the social credibility of the robot below the disengagement threshold, and therefore result in reduced capability (both routine and safety-critical) due to lack of user engagement.

A loss of social credibility has significant impact on the socially important aspects of a robot's functionality. In more detail, the characteristics identified in Section IV-A as associated with social credibility are also important for user engagement. Trust, for example, means that a user is likely to extrapolate from observed characteristics of the robot to generalise about its wider capabilities [41]. While over-trust is in itself a problem [28], a lack of trust in the robot means that users are likely only to engage the robot in scenarios which they have directly observed to be satisfactorily carried out. That is, even where the overall social credibility has not been driven below the disengagement threshold, the overall capability of the robot may still be impaired. Similarly, a lack of trust in a robot may lead to negative associations with it, and a reluctance on the part of the user to engage [29].

It is therefore clear that a balance will need to be struck between performing safety-critical behaviours, and performing the social routines necessary to build user engagement (socially-important behaviours).

D. Schedulability of behaviours

We propose the identification of an optimum scheduling such that socially-important and safety-critical behaviours can both be performed to an acceptable level. This will correspond to maintaining social credibility above the disengagement threshold by delaying behaviours based on their priority, where priority considers both safety and social engagement.

Such a prioritisation system would correspond to trading off (safety) risks against (social) benefit, a concept described in [42]. Traditional scheduling algorithms could be used to ensure that the correct behaviours are selected to run, with a level of customisation also being provided.

This problem has been explored extensively when considering scheduling within mixed-criticality systems (see [43] for an overview). In the case of assistive robots, the following (non-exhaustive) criteria should be considered for schedulability:

- Estimated risk associated with not fulfilling the behaviour
- Estimated loss of social credibility associated with fulfilling the behaviour
- Current social credibility as considered against the disengagement threshold
- Functional importance of other behaviours

Such a prioritisation system could also be customised to allow users and care-givers to adjust the balance between safety and social behaviours. A user more comfortable and engaged with the robot may not need the same degree of social behaviours as a user who has not engaged with the robot before. Similarly, a user requiring a higher level of care may want to prioritize safety-critical behaviours.

VI. Conclusions and Future Work

In this paper we have identified a link between safety and *social credibility*, where this is defined as being a reflection of how well a robot follows social norms. The relevant social norms are dependent on the environment and purpose of the robot, and we have presented some examples that would apply

 ISBN: 978-1-61208-698-9

to an assistive robot. We have also drawn on analysis of existing systems to identify how user disengagement can affect both social credibility and the safety-critical functions of an assistive robot. In this process, we have shown how loss of social credibility can lead to effective loss of these safety functions.

We have built on this in order to discuss prioritisation of socially-important behaviours and safety-related behaviours, particularly where these may conflict. Over-prioritisation of safety-related behaviours can itself lead to a loss of social credibility, and to user disengagement. Correspondingly, over-prioritisation of routine behaviours can lead to poor performance in the robot's safety-related roles. We have proposed a solution to this that builds on existing concepts of mixed-criticality system scheduling. Such a scheduling would rely on a prioritisation system that takes both safety and social engagement into account.

As part of future work, we propose to develop this prioritisation further. We will evaluate in a user study how exactly social credibility could be affected by violations of social norms that are required from a safety point of view. Furthermore, we plan to investigate how safety-relevant routines might be neglected by the user when the robot is not perceived as socially credible. This data will be used in studies futher investigating the automatic scheduling of behaviours to ensure the robot maintains high levels of social credibility while being acceptably safe to operate. We also propose to expand this work to discussions of other autonomous systems, providing a generalised mechanism for assuring safety of a robot which must also perform another (social) function.

REFERENCES

[1] J. Broekens, M. Heerink, and H. Rosendal, "Assistive social robots in elderly care: A review," *Gerontechnology*, vol. 8, no. 2, pp. 94–103, 2009.

[2] F. Amirabdollahian *et al.*, "Assistive technology design and development for acceptable robotics companions for ageing years.," *Paladyn: Journal of Behavioral Robotics*, vol. 4, no. 2, pp. 94–112, 2013.

[3] R. Kittmann *et al.*, "Let me introduce myself: I am care-o-bot 4, a gentleman robot," in *Mensch und Computer 2015 – Proceedings*, S. Diefenbach, N. Henze, and M. Pielot, Eds., Berlin: De Gruyter Oldenbourg, 2015, pp. 223–232.

[4] J. Saunders, N. Burke, K. L. Koay, and K. Dautenhahn, "A User Friendly Robot Architecture for Re-ablement and Co-learning in A Sensorised Home," in *European AAATE (Associated for the Advancement of Assistive Technology in Europe) Conference*, Vilamoura, Portugal, 2013, pp. 49–58.

[5] S. Bedaf, P. Marti, F. Amirabdollahian, and L. de Witte, "A multi-perspective evaluation of a service robot for seniors: The voice of different stakeholders," *Disability and Rehabilitation: Assistive Technology*, pp. 1–8, 2017.

[6] J. Saunders, D. Syrdal, K. L. Koay, N. Burke, and K. Dautenhahn, "'Teach Me - Show Me' - End-user personalisation of a smart home and companion robot," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 27–40, 2016.

[7] M. Webster *et al.*, "Toward reliable autonomous robotic assistants through formal verification: A case study," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 2, pp. 186–196, 2016.

[8] International Standards Organization, "Robots and robotic devices - safety requirements for personal care robots," *ISO 13482*, 2014.

[9] UK Health and Safety Executive, *Reducing Risks, Protecting People*. HSE Books, London, UK, 2001.

[10] Fraunhofer IPA, *Care-o-bot data sheet*, 2018.

[11] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[12] J. C. Knight, "Safety critical systems: Challenges and directions," in *Proceedings of the 24th International Conference on Software Engineering*, ACM, 2002, pp. 547–550.

[13] International Electrotechnical Commission, "Functional safety of electrical / electronic / programmable electronic safety related systems," *IEC 61508*, 2010.

[14] J. McDermid and T. Kelly, "Software in safety critical systems-achievement & prediction," *Nuclear Future*, vol. 2, no. 3, p. 140, 2006.

[15] UK Department of Health, *Care services efficiency delivery programme 'homecare re-ablement workstream: Retrospective longitudinal study'*, 2007.

[16] K. L. Koay, D. S. Syrdal, M. Ashagari-Oskoei, M. L. Walters, and K. Dautenhahn, "Social Roles and Baseline Proxemic Preferences for a Domestic Service Robot," *International Journal of Social Robotics*, vol. 6, no. 4, pp. 469–488, 2014.

[17] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robotics and autonomous systems*, vol. 42, no. 3-4, pp. 143–166, 2003.

[18] D. S. Syrdal, K. Dautenhahn, M. L. Walters, and K. L. Koay, "Sharing Spaces with Robots in a Home Scenario – Anthropomorphic Attributions and their Effect on Proxemic Expectations and Evaluations in a Live HRI Trial," in *AAAI Fall Symposium "AI in Eldercare: New Solutions to Old Problems"*, Washington, DC, USA, 2008, pp. 116–123.

[19] P. Holthaus, K. Pitsch, and S. Wachsmuth, "How Can I Help? - Spatial Attention Strategies for a Receptionist Robot," *International Journal of Social Robotics*, vol. 3, no. 4, pp. 383–393, 2011.

[20] M. N. Nicolescu and M. J. Mataric, "Learning and interacting in human-robot domains," *IEEE Transactions on Systems, man, and Cybernetics-part A: Systems and Humans*, vol. 31, no. 5, pp. 419–430, 2001.

[21] F. Hegel, S. Gieselmann, A. Peters, P. Holthaus, and B. Wrede, "Towards a typology of meaningful signals and cues in social robotics," in *2011 RO-MAN*, 2011, pp. 72–78.

[22] D. Feil-Seifer and M. J. Matarić, "Socially assistive robotics," *IEEE Robotics & Automation Magazine*, vol. 18, no. 1, pp. 24–31, 2011.

[23] D. T. Anderson, J. M. Keller, M. Skubic, X. Chen, and Z. He, "Recognizing falls from silhouettes," *Electrical and Computer Engineering publications (MU)*, pp. 6388–6391, 2006.

[24] M. Heerink, B. Kröse, V. Evers, and B. Wielinga, "The influence of social presence on acceptance of a companion robot by older people," *Journal of Physical Agents*, vol. 2, no. 2, pp. 33–40, 2008.

[25] A. Hamacher, N. Bianchi-Berthouze, A. G. Pipe, and K. Eder, "Believing in bert: Using expressive communi-

 ISBN: 978-1-61208-698-9

cation to enhance trust and counteract operational error in physical human-robot interaction," in *Robot and Human Interactive Communication (RO-MAN), 2016 25th IEEE International Symposium on*, IEEE, 2016, pp. 493–500.

[26] J. Forlizzi, "How robotic products become social products: An ethnographic study of cleaning in the home," in *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, ACM, 2007, pp. 129–136.

[27] J. Lee and K. See, "Trust in automation: Designing for appropriate reliance," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 46, pp. 50–80, 2004.

[28] M. Salem, G. Lakatos, F. Amirabdollahian, and K. Dautenhahn, "Would You Trust a (Faulty) Robot?: Effects of Error, Task Type and Personality on Human-Robot Cooperation and Trust," in *International Conference on Human-Robot Interaction (HRI)*, Portland, Oregon, USA: ACM/IEEE, 2015, pp. 141–148.

[29] M. Desai, K. Stubbs, A. Steinfeld, and H. Yanco, "Creating trustworthy robots: Lessons and inspirations from automated systems," in *Proceedings of the AISB Convention: New Frontiers in Human-Robot Interaction*, 2009, pp. 49–56.

[30] C. D. Kidd, W. Taggart, and S. Turkle, "A sociable robot to encourage social interaction among the elderly," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, IEEE, 2006, pp. 3972–3976.

[31] A. Rossi, K. Dautenhahn, K. L. Koay, and M. L. Walters, "How the timing and magnitude of robot errors influence peoples' trust of robots in an emergency scenario," in *International Conference on Social Robotics*, Springer, 2017, pp. 42–52.

[32] M. Salem, G. Lakatos, F. Amirabdollahian, and K. Dautenhahn, "Towards Safe and Trustworthy Social Robots: Ethical Challenges and Practical Issues," in *International Conference on Social Robotics*, Cham: Springer International Publishing, 2015, pp. 584–593.

[33] S. Bedaf, H. Draper, G.-J. Gelderblom, T. Sorell, and L. de Witte, "Can a service robot which supports independent living of older people disobey a command? the views of older people, informal carers and professional caregivers on the acceptability of robots," *International Journal of Social Robotics*, vol. 8, no. 3, pp. 409–420, 2016.

[34] P. Saulnier, E. Sharlin, and S. Greenberg, "Exploring minimal nonverbal interruption in hri," in *RO-MAN, 2011 IEEE*, IEEE, 2011, pp. 79–86.

[35] A. Sardar, M. Joosse, A. Weiss, and V. Evers, "Don't stand so close to me: Users' attitudinal and behavioral responses to personal space invasion by robots," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, ACM, 2012, pp. 229–230.

[36] J. Mumm and B. Mutlu, "Human-robot proxemics: Physical and psychological distancing in human-robot interaction," in *Proceedings of the 6th international conference on Human-robot interaction*, ACM, 2011, pp. 331–338.

[37] C. Bartneck, T. Kanda, O. Mubin, and A. Al Mahmud, "Does the design of a robot influence its animacy and perceived intelligence?" *International Journal of Social Robotics*, vol. 1, no. 2, pp. 195–204, 2009.

[38] J. Wall, V. Cuenca, K. Creef, and B. Barnes, "Attitudes and opinions towards intelligent speed adaptation," in *Intelligent Vehicles Symposium Workshops (IV Workshops), 2013 IEEE*, IEEE, 2013, pp. 37–42.

[39] N. B. Sarter and D. D. Woods, "Team play with a powerful and independent agent: Operational experiences and automation surprises on the airbus a-20," *Human factors*, vol. 39, no. 4, pp. 553–569, 1997.

[40] E. Horvitz, J. Apacible, and M. Subramani, "Balancing awareness and interruption: Investigation of notification deferral policies," in *International Conference on User Modeling*, Springer, 2005, pp. 433–437.

[41] P. Robinette, W. Li, R. Allen, A. M. Howard, and A. R. Wagner, "Overtrust of robots in emergency evacuation scenarios," in *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, IEEE Press, 2016, pp. 101–108.

[42] C. Menon and R. Alexander, "Ethics and the safety of autonomous systems," in *Proceedings of the 26th Safety Critical Systems Symposium*, Safety Critical Systems Club, 2018, pp. 25–43.

[43] A. Burns and R. Davis, "Mixed criticality systems-a review," *Department of Computer Science, University of York, Tech. Rep*, pp. 1–69, 2013.

 ISBN: 978-1-61208-698-9

A GRU-based Meta-learning Model Based on Active Learning

Honglan Huang
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: huanghonglan17@nudt.edu.cn

Shixuan Liu
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: liushixuan19@nudt.edu.cn

Yanghe Feng
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: fengyanghe@yeah.net

Jincai Huang
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: huangjincai@nudt.edu.cn

Zhong Liu
College of Systems Engineering, National University of Defense Technology
Changsha, China
e-mail: liuzhong@nudt.edu.cn

***Abstract*—In the realities of machine learning, labeling a data set may be expensive, tedious, or extremely difficult and it is often not easy to choose the common criteria for active learning to select samples for different data sets. In order to solve these difficulties, this paper introduces a Gated Recurrent Unit (GRU)-based meta-learner model, which combines active learning with reinforcement learning and uses it in a stream-based one-shot learning task. Based on the uncertainty of the instances, the model learns an action strategy that determines when to predict or request the label of each instance. Through the experiments on Omniglot dataset, the model shows its ability to achieve a good prediction accuracy with few label requests.**

Keywords-active learning; meta learning; reinforcement learning; GRU.

I. INTRODUCTION

Active learning [1] uses unlabeled and labeled instances to train a highly accurate classifier to reduce the workload of human experts. The algorithm simulates the human learning process, selects part of instances to label and iteratively improves the generalization performance of the classifier. Therefore, it has been widely used in information retrieval and text, image and speech recognition in recent years.

Most of the traditional active learning methods are carefully formulating some criteria for selecting samples, such as uncertainty sampling [2], query-by-committee [3], margin [4] and representative and diversity-based sampling [5]. It's hard to pinpoint which approach is better, because each approach starts from a reasonable, meaningful and completely different motivation. However, for now, there is no universal criteria that performs well on all datasets. This paper introduces a learning-based approach, rather than a manually-designed sample-selection criterion, which integrates active learning algorithm with reinforcement learning. Our approach not only learns to use small supervisors to classify instances, but also learns about label-querying strategies. The model adopts a stream-based active learner that considers the online environment for active learning.

Our primary contribution in this work is using a GRU to improve the active one-shot learning model introduced by Woodward *et al.* [6]. We evaluate the model on Omniglot ("active" variants of existing one-shot learning tasks [7]), and our experiment results show that it can learn label-querying strategies efficiently with simpler structure.

The rest of this paper is structured as follows. Section II summarizes the existing approaches related to our work. Section III presents the task and the general framework of our proposed active learning model. Section IV presents the experiments and interprets the results. Finally, we conclude this paper in Section V.

II. RELATED WORK

Active learning has been well studied in the past few decades. The main idea of the active learning is that a learner should achieve higher accuracy with fewer labeled training instances, if it is able to choose the training instances from which it learns. Numerous algorithms have been proposed to design the criteria for the selection of which examples to label [2][5][8][9]. However, most of these traditional active learning methods are based on heuristics, which may be limited when the data distribution of the underlying learning problems vary (e.g. a new class appears). Instead, we used a meta-learning approach to train an active learner via reinforcement learning to solve a one-shot learning task. The idea of combing active learning and reinforcement learning was recently investigated by Woodward *et al.* [6]. In contrast to their work, we used a GRU instead of a Long Short-Term Memory (LSTM) network to approximate the action-value function in reinforcement learning. Compared with LSTM, GRU has fewer parameters, so it can effectively speed up the training process [10] and requires fewer samples, which is more suitable for our one-shot active learning task. Similar inspirations have also been studied by Bachman *et al.* [7] Pang *et al.* [11] and Puzanov *et al.* [12].

 ISBN: 978-1-61208-698-9

III. MODEL DESCRIPTION

A. Task Description

We mainly focus on the stream-based online active learning scenario [6], in which instances can be continually obtained from the data stream and presented in an exogenously-determined order. Thus, the input of the model is a stream of images, the label of which is queried or predicted by our model. The performance of our model was improved with a short training episode and a small number of examples per class to maximize the performance of test episodes, which consists of classes that are not encountered in training. The structure of our task can be seen in Figure 1. The classes and their labels and the specific samples are shuffled and randomly presented at each episode.

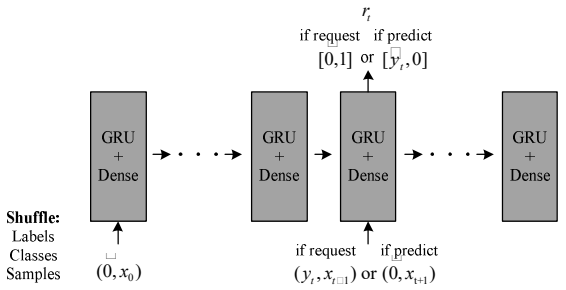


Figure 1. Task structure

Figure 2. Example of 3-way problem instance on Omniglot.

At each time step, the input of the model is an image along with a vector that depends on the output taken previous instance as input. The *N*-way task is set up as follows: pick *N* unseen classes per episode. Figure 2 shows an example of a 3-way problem on Omniglot. The output of the model is a one-hot vector of length *N +1*. If the model requests the label of the image x_t, it sets the final bit of the output vector of this timestep to 1, which means the output of timestep t is $[\vec{0}, 1]$. Thus, the reward of this label request action is R_{req}. The true label y_t of image x_t is then provided at the next time step along with the next image x_{t+1}, so the input of $t+1$ is (y_t, x_{t+1}). Alternatively, if the model makes a prediction of x_t, it sets one of the first k bits of the output vector to represent $\hat{y}$, so the output of this step is $[\hat{y}, 0]$. The reward of this action is R_{cor} if the prediction is correct or R_{inc} if incorrect. If a prediction is made at time step t, no information of its true label y_t is supplied at the next time step $t+1$, then the input is $(\vec{0}, x_{t+1})$ instead.

B. Methodology

We use a model-free reinforcement learning method Q-learning to learn an optimal action strategy, which can maximize the rewards. The loss function we use is defined as,

$$\mathcal{L}(\theta) \coloneqq \sum_t \Big[Q(o_t, a_t) - \Big(r_t + \gamma \max_{a_{t+1} \in \mathcal{A}} Q^*(s_{t+1}, a_{t+1}) \Big) \Big]^2 \quad (1)$$

where, θ are the parameters of the function approximator, o_t are the observations such as images that the agent receives, a_t is the action the model chooses at timestep t, Q^* is the optimal value of action-value function Q.

We use a GRU [13] network connected to a linear output layer to adopt the methodology of using action-value function $Q(o_t, a_t)$ in Q-learning. $Q(o_t)$ outputs a vector, where each element corresponds to an action:

$$Q(o_t, a_t) = Q(o_t) \cdot a_t \quad (2)$$

$$Q(o_t) = W^{hq} h_t + b^q \quad (3)$$

where, b^q is the action-value bias, h_t is the output of the GRU, W^{hq} are the weights mapping from the GRU output to action-values.

IV. EXPERIMENTS

A. Setup

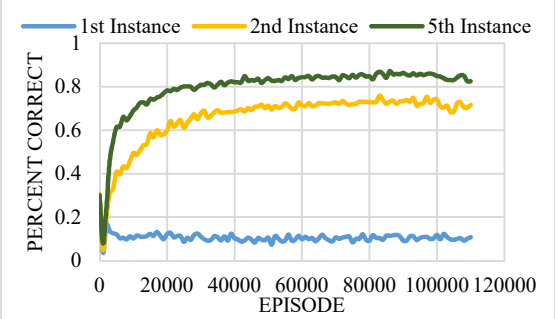
The experiments were carried out on the Omniglot dataset [14] that contains 32460 instances having 1623 classes of characters from 50 different alphabets, each hand-written by 20 different persons. The dataset was randomly divided into 1200 characters for training and the rest 423 characters are kept for testing. The images were downscaled to 28×28 pixels and each pixel was normalized between 0.0 and 1.0.

30 Omniglot images from 3 random classes were chosen in each episode. Each class of images was randomly rotated in $\{0°, 90°, 180°, 270°\}$. A GRU with 200 hidden units was used here. We optimized the parameters of our model using Adam with the default parameters [15]. A grid search was performed over the following parameters, and the parameters of the results reported in this article are listed here. During the training process, epsilon-greedy ($\epsilon = 0.23$) exploration is set for actions selection. The learning rate of training was set to 0.001 and the discount factor γ was set to 0.8. The reward values were set as: $R_{cor} = +1, R_{inc} = -1$, $R_{req} = -0.3$.

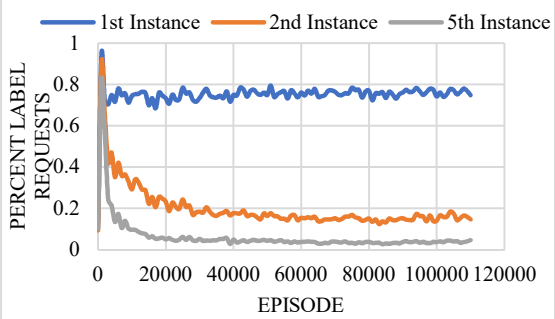
B. Results

Here, we present the results of our experiments. The 1st, 2nd, 5th of all classes in each episode were identified. After 100,000 episodes, training is ceased and the model was given 10,000 more test episodes. No further updates occurred during these episodes. The results can be seen in Figure 3.

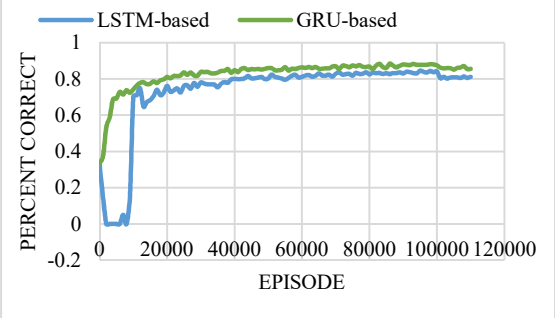
 ISBN: 978-1-61208-698-9



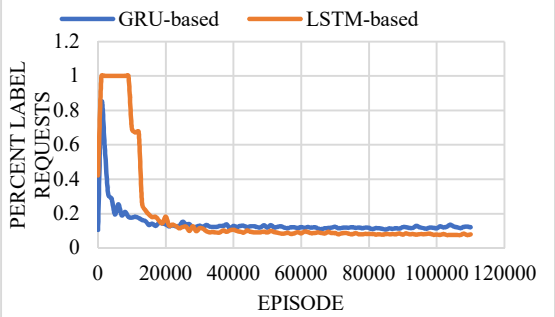
(a) Identified accuracy



(b) Identified label request rate



(c) Overall accuracy



(d) Overall request rate

Figure 3. Experiment results

As can be seen in the plot, the proposed GRU-based meta-learning model learns to query the label for early instances of a class, and makes more prediction for later instances. Simultaneously, the accuracy of the model is improved on later instances of a class. It shows that our model has learned an effective querying strategy that effectively requests tags when new classes are present, and quickly learns useful information to make accurate predictions when they encounter the same category in the future. After initial training, our model accuracy rate was stable at 85%, while the label request rate was stable at 12%. Compared with supervised learning, our model greatly reduces the dependence on the number of labels and human workload, and achieves decent prediction accuracy. At the same time, our method speeds up the convergence of the algorithm compared to the LSTM-based method [6].

V. CONCLUSION

We introduced a GRU-based meta-learning model that learns active learning in an reinforcement learning way and experimented it on Omniglot one-shot learning tasks. Our results show that our model can learn an optimal query strategy and achieve a good classification accuracy with a small amount of labeled data.

As we used a GRU network to approximate the action-value function in reinforcement learning, a promising direction is that the GRU network can be replaced by a more sophisticated one-shot learning approach such as Matching Network [16] or Memory-Augmented Neural Networks [17]. We will leave this as our future work.

REFERENCES

[1] B. Settles, “Active Learning Literature Survey,” University of Wisconsinmadison, vol. 39, no. 2, pp. 127–131, 2009.

[2] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, “Active Learning with Gaussian Processes for Object Categorization,” vol. 88, no. 2, pp. 1-8, 2015.

[3] Seung, S. H., Opper, and Sompolinsky, “Query by committee,” Proc of the Fith Workshop on Computational Learning Theory, vol. 284, pp. 287-294, 1992.

[4] S. Tong, and D. Koller, Support vector machine active learning with applications to text classification: JMLR.org, 2002.

[5] R. Chattopadhyay, Z. Wang, W. Fan, I. Davidson, S. Panchanathan, and J. Ye, “Batch Mode Active Sampling based on Marginal Probability Distribution Matching,” Acm Transactions on Knowledge Discovery from Data, vol. 7, no. 3, pp. 1-25, 2013.

[6] M. Woodward, and C. Finn, “Active One-shot Learning,” 2017.

[7] P. Bachman, A. Sordoni, and A. Trischler, “Learning Algorithms for Active Learning,” 2017.

[8] S. Huang, J. Chen, X. Mu, and Z. Zhou, "Cost-Effective Active Learning from Diverse Labelers." pp. 1879-1885.

[9] R. Giladbachrach, A. Navot, and N. Tishby, "Query by Committee Made Real." pp. 443-450.

[10] J. Chung, C. Gulcehre, K. H. Cho, and Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” Eprint Arxiv, 2014.

[11] K. Pang, M. Dong, Y. Wu, and T. Hospedales, “Meta-Learning Transferable Active Learning Policies by Deep Reinforcement Learning,” 2018.

[12] A. Puzanov, and K. Cohen, “Deep Reinforcement One-Shot Learning for Artificially Intelligent Classification Systems,” 2018.

[13] L. Agatha, D. Arnaud, P. D. Walshaw, A. L. Cho, R. M. Bilder, J. J. Mcgough, J. T. Mccracken, M. Scott, and S. K. Loo, “Electroencephalography correlates of spatial working memory deficits in attention-deficit/hyperactivity disorder: vigilance, encoding, and maintenance,” Journal of Neuroscience the Official Journal of the Society for Neuroscience, vol. 34, no. 4, pp. 1171-82, 2014.

[14] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, “Human-level concept learning through probabilistic program induction,” Science, vol. 350, no. 6266, pp. 1332-1338, 2015.

[15] D. P. Kingma, and J. Ba, “Adam: A Method for Stochastic Optimization,” Computer Science, 2014.

[16] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, “Matching Networks for One Shot Learning,” 2016.

[17] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap, “One-shot Learning with Memory-Augmented Neural Networks,” 2016.

 ISBN: 978-1-61208-698-9

A Systemic Approach for Safe Integration of Products and Systems

Mohammad Rajabalinejad
Department of Design, Production, and Management
University of Twente, Enschede, the Netherlands
Email: M.Rajabalinejad@utwente.nl

***Abstract*— Safe integration is an unresolved issue across different disciplines, and many problems happen due to improper integration of a product or system. Safe integration is beyond technical integration and requires both technical and nontechnical knowledge. This paper highlights the scope of integration challenges and sheds light on safe integration. It outlines a systemic view of safe integration and provides an example application for further clarification.**

Keywords - systems integration; safe integration; integration engineering; Safety Cube.

I. INTRODUCTION

Our society is becoming less tolerant to safety failures while it demands up-to-date technologies. People require seamless integration of new technologies with everyday life. We need products and services that are effortlessly usable in different contexts. Given the increasing complexity of high-tech systems, there is a need for new methods and techniques to support proper integration of newly developed systems or products. The challenge is far beyond technical installations and more than the integration of hardware, software, and human for a single product or system. The high pace of technological developments demands strategies that not only fulfil the technical requirements but also successfully address interoperability and dependability of systems, data integrity, security, or privacy matters. The main drivers and ingredients for safe integration are presented in Figure 1.

Integration creates a unique selling point for businesses. For example, Apple is conscious about seamless integration among its products aiming to deliver the ultimate use-experience for the users. In brief, proper integration is a prerequisite for a modern society. In the previous work [1], the author provides several examples of systems challenges for the rail industry. Yet, the scope of integration challenges crosses different industries.

The public is sensitive to integration failures imposing extra costs and resources [2]. Examples of needs for integration are across different disciplines and industries. Augmented Reality (AR) and its integration with human-life in the form of camera, wearables, games, or education are examples for the need for safe integration of technology with everyday life. Artificial Intelligence (AI) is another

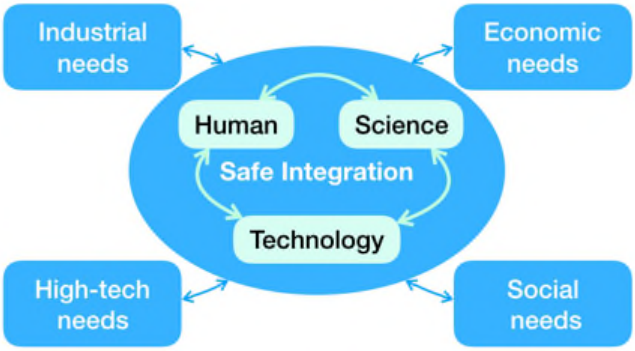


Figure 1. Drivers for safe integration

example where machines are being used to facilitate higher capabilities and performances. Here, safe integration is required at different levels. The first level of integration is superposition of components to make a product. If the components are properly put next to each other, then the product as a whole should be properly integrated and used. This is the second level of integration. For the third level of integration, the product has to be properly integrated into the environment and be safely used. Integration issues happen at all these levels, and the issues can go beyond technical matters. Figure 2 presents three different examples for the integration problem for bicycles. In all these three cases, the issues were dangerous to users and therefore the products were pulled out of the European market. Figure 2.A presents a city bike which was recalled under alert number A12/0134/19. The defect in the front mudguard may block the front wheel of this bicycle during the use and lead to an accident. Figure 2.B shows a children's bicycle where the nuts on the cranks have sharp edges, and they may harm children during the riding or maintenance of the bicycle. This product, which was recalled under the alert number A11/0066/17, is an example of faulty design with regard to human-product integration. The third example, Figure 2.C presents a bike which suffers from defective sealing for its batteries which may result in accumulation of humidity inside the battery and cause overheating and self-ignition. This is an example issue for integration of a product with its environment. This product was recalled under the alert number A12/0497/15.

 ISBN: 978-1-61208-698-9

(A) Recall of the product from end users in Europe (example of internal integration issues)

(B) Recall of the product from end users in Europe (example of product-user integration issues)

(C) Recall of the product from end users in Europe (example of product-environment integration issues)

In addition to highlighting the needs for integration, this paper reviews currently used tools and discusses the ingredients for safe integration. Section II provides a review of tools and techniques. The outcomes have been further discussed in Section III, where a systemic approach for safe integration is described. Section IV presents an example application for the safe integration of bicycles to the urban system. Conclusions are drawn in Section V.

II. SAFE INTEGRATION

Safe integration starts with a proper understanding of the stakeholders and their needs. Systems Engineering handbook highlights the human system integration (HSI). HSI considers domains such as human factors engineering (human performance, human interface, user centred design), workload (normal and emergency), training (skill, education, attitude), personnel (ergonomics, accident avoidance), working condition and health (hazard avoidance) [3]. These domains have direct links to safety. As a matter of fact, integration is similar to safety from several perspectives inheriting a multidisciplinary nature where different techniques and methods can be used for safe system integration. The Swiss cheese model of accidents developed by J.T. Reason presents a model for integration of different system layers in which the risk of a threat may become a reality [4]. The failure mode and effect analysis (FMEA) helps finding potential failure modes for hardware, software, or processes. The fault tree analysis is a systematic approach to present the possible faults related to a specific event. For analysing the operability problems, hazard and operability analysis (HAZOP) is used. The root cause analysis (RCA) focuses on the positive and negative consequences of events. ISO 12100, the reference standard for safety of machinery, pays special attention to safety matters during assembly of a machine or its integration with the surrounding environment [5]. IEC 61508 a seminal standard for functional safety delivered in several parts. Its first three parts focus respectively on general requirements, requirements for E/E/PE, and requirements for software for safety-related systems. Part 1 of this standard addresses issues on system safety validation and system integration (tests) including architecture, software, and PE integration tests. Part 2 addresses the module and system integration for safety-related systems, and Part 3 focuses on software testing and integration. Integration is comparable with safety inheriting multidimensional problems where stakeholders with shared goals need experience and technology to make proper decisions and remove, minimise, or control the risks. Technology readiness level (TRL), integration readiness level (IRL), safety by design and safety cubes are the methods to ensure better integration of products or systems.

As a result of reviewing these references, three common blocks have been identified for these as discussed earlier in [1]. Human (or people), system and environment are the

 ISBN: 978-1-61208-698-9

three building blocks for both of the design process and safety management process.

III. PRINCIPLES FOR SAFE INTEGRATION

One of the primary tasks for engineering design, systems engineering, or risk management is to ensure seamless and safe integration of a system with its environment. In this perspective, dealing with relations among the system, subsystems, environment, and people is of primary concern. These relations, or the so-called interfaces, represent one of the core issues for proper integration. Figure 3 schematically shows the main building blocks for safe integration and their relations. These are principles elements of the so-called safety cube, will be discussed in further details next.

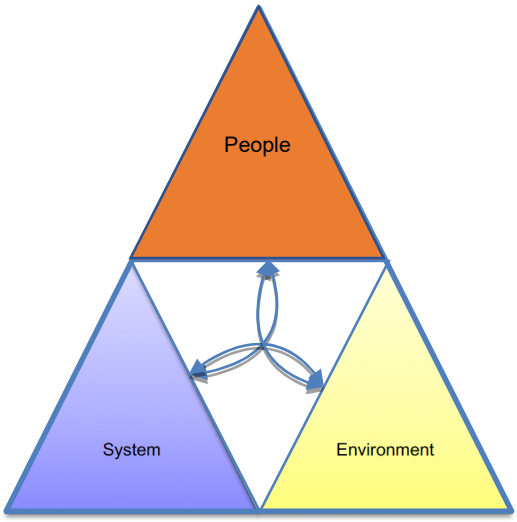


Figure 3. Elements of safe integration and the safety cube.

A. Human

Human or people in this context refers to individual or group of individuals who have connections to the system of interest. They can be stakeholders, designers, users, operators, owners, service providers, producers, or other humans who directly or indirectly have interest in the system and cooperate or compete with it. People have their own individual or organisational culture.

B. System

System refers to the system (or product) of interest that delivers the required functions. The system of interest is independent, and it can be a part of a system of systems. The system includes subsystems or components that form its structure to deliver the required functions under its specific behaviour. Equipment, facilities, and procedures for operation are parts of the system.

C. Environment

Environment includes the system of interest, the cooperating systems, and the competing systems which influence the system of interest. This excludes people which have been discussed earlier. Relevant regulations, industry standards, or supporting facilities are part of the system environment.

D. Human-system relation

Human can have different roles and consequently different relations with the system of interest. For a system, the user, stakeholder, operator, owner, or supplier may have different, competing, or even conflicting interests. This relation can be in the form of (physical) interface, operation, control, maintenance, or cognitive which can directly or indirectly influence the system. Operational and safety culture influences human-system relation.

E. System-environment relation

The system of interest connects to its environment. The relation between a system and its environment is often seen in the form of interfaces for technical installation under three categories of structure, information, and energy. It is important to note that the system is also under the influence of regulations, policy, and political interests of the environment.

F. Human-environment relation

Although, the relation between human and environment often falls out of the scope of system of interest in technological design, it has dominant influence on the system of interest. Change of regulations in a dynamic and competitive (geo)political context, policy making and governance are examples of human-environment relations. This often becomes very complex for systems where multiple stakeholders are involved.

To summarise, Table I provides an overview for the outcomes of this section. This is the information needed for forming a safety cube. The diagonals of this table specify the human, system, and environment for the system of interest where the other cells provide information about the connection between diagonals. The off-diagonals have to be read clock-wise in such a way that the associated row provides input for the associated column. For example, the

TABLE I. THE ELEMENTS OF SAFETY CUBE FOR SAFE INTEGRATION

	Human	*System*	*Environment*
Human	users, direct/indirect stakeholders, operators	human input for the system, intended use or misuse scenarios	human input for environment or its system of systems, use or misuse scenarios
System	system inputs, functions, malfunctions, or services for human	system of interest, its structure, functions, procedures, …	system input for environment, intended use or misuse scenarios
Environment	environmental inputs, functions, malfunctions, or services for human	environmental inputs, functions, malfunctions, or services for the system	cooperating or competing systems, physical environment, policy, regulations

 ISBN: 978-1-61208-698-9

human-system cell at the top row describes the human output as input for the system whereas the system-human cell at the second row describes the system output as input for human.

Table I summarises the system definition and provides an overview of the building blocks and their connections for safe integration. Although this is an important starting point and it is necessary to have a good understanding of the system and interaction between its elements, it does not focus on the system of interest. Therefore, there is a need to reorganise this information and move the focus to the system, subsystems, functions, structure, and behaviour. For this purpose, the points below need to be considered.

- The system of interest needs to be elaborated and relations between system, its subsystems, and super-systems need to be elaborated in further details.
- System of systems and environment can be merged. As result, the term environment refers to both system of systems and environments.
- Human is partly related to use and partly related to the environment of the system.

In order to address these points, Table II is produced representing the results of Table I with more focus on the system of interest. This presents the information for the so-called system safety cube. The rows of this table focus on the system of interest, its super system (or environment) and subsystems. The columns focus on requested functions (or malfunctions), physical structure, and the use (or misuse) scenarios. The questions below help to keep the focus per each column.

For the first column of Table II, the relevant questions are the following.

- Why does the (super/sub) system of interest exist?
- What is its purpose?
- What does it do?
- What are the requirements?
- What are the functions and services?
- What if it malfunctions or the services are interrupted?

For the second column of the table, one may ask the following questions.

- What are the elements of this (super/sub) system of interest?
- How do they connect?
- How is the energy provided?
- How is the information flow?
- What are the interfaces?
- How does it work?
- What if some components, subsystems, or interfaces fail?

For the third column of this table, or the use purpose, one may ask the following questions:

- Who are the people who have interest in the system?
- How do they influence the system?
- How do they use it?
- What are the foreseeable misuse scenarios?

TABLE II. SAFE INTEGRATION WITH FOCUS ON SYSTEM, THE SO-CALLED SYSTEM SAFETY CUBE

	System requirements, functions, and behaviour	*Physical system (system-SoS/environment relation)*	*Use/misuse scenarios (human-system relation)*
Environment and super systems	environmental requirements, policy, regulations	environmental/ super-system interfaces	user specifications/ interest, information for use, use/ safety culture
System	system requirements and functions. Modes of operation.	system level specifications: structure/ interfaces and subsystem failures	system level use/misuse scenarios, operation scenarios, accident history
sub-systems	sub-systems and components failures	sub-system level specifications structure/ interfaces and component failures	sub-system level use and misuse cases, intervention procedures

IV. EXAMPLE APPLICATION

This section presents an example application for safe integration of a bicycle to the urban environment. This is an interesting example because cycling is economic, healthy, and green for urban transportation. Yet, safety of cyclists is essential for making that a popular way of urban transportation. In the Netherlands, about 35% of people use frequently bicycles on a daily basis and this backs the public demand for safety. In 1970, people protested against a high number of child death on the roads and started the movement entitled "stop the child murderer" because of a high rate of casualties, especially on the cross-overs [6]. This demand influenced the government policy in the Netherlands perceiving bicycle as a critical means for safe

TABLE III. THE ELEMENTS OF SAFETY CUBE FOR SAFE INTEGRATION OF BICYCLES

	Human	*System*	*Environment*
Human	cyclist, other road users, regulators, service providers	traffic rules, quality & condition control, human-power input, steering	driving culture of e-bikes, cars, motorcycles, or other road users
System	safe, comfortable, economic, healthy, and enjoyable personal-transport	bicycle	visibility in day light, night, or at rain
Environment	traffic regulations, and traffic management system, climate requirements	bicycle (or safe) path, spare parts, fallen trees, snow or ice on the path, fallen trees or bushes	road, signs, curbs, markings, other road-vehicles, crossing, parking, climate, policy, regulations

 ISBN: 978-1-61208-698-9

TABLE IV. SAFE INTEGRATION WITH FOCUS ON SYSTEM, THE SO-CALLED SYSTEM SAFETY CUBE, FOR BICYCLES

	System requirements, functions, and behavior	*Physical system (system-SoS/environment relation)*	*Use/misuse scenarios (human-system relation)*
Environment and super systems	traffic regulations in Netherlands and Europe, control functions	bicycle path, roads, crossing, traffic lights, infrastructure, and natural environment	driving behavior of other users on bicycle path or adjacent roads
System	ergonomically safe, CE marking, meet the expected safety level, visible to other users	a two-wheels personal vehicle powered & steered by human	cyclist cycles in a (non) specified path at night, rain, or cross roads, cyclist uses unassigned paths (shortcuts)
sub-systems	components need to comply with standards	two wheels, frame, pedals chain, tires may go flat	cyclists sits on (side) saddle, inaccurate adjustment, stands on pedals, steers by one hand

transportation in urban areas. Along with geographical considerations, bike-friendly infrastructures and bike-friendly policy are the keys for the safe integration of bicycles into the system [7].

Here in this example, elements for safe integration have been described and listed through the approach introduced earlier in this paper. For this purpose, three elements of human, system, and environment are the starting points. Table III describes these three elements and their connections. This table shows what the needs are for creating safe cycling experience for users. It is far beyond a design of a safe bicycle and safe helmet requiring an integral view that combines proper infrastructures with supportive policy and embracing culture in order to achieve the optimum results.

Table IV represents this information with the focus on the system of interest, its subsystems, and super-system. It is important to note that the tables presented here for this example do not present all the detailed information for the safe integration of bicycles into urban areas.

In order to verify if the proposed approach can capture the essential elements of safe integration, a number of references have been reviewed as mentioned earlier in this section. The results confirm that the elements of safe integration have been captured in this approach. Yet, further elaboration is needed capture the details elements of safe design and their connections for safe integration.

V. CONCLUSIONS

For safe integration, one needs to pay attention to the system, its environment, and people who have connection to the system. As a matter of fact, the prerequisite of safe integration is proper system definition describing the system of interest, its structure, requirements and behaviour, people who influence it, its environment or super-system, and the relations. For safe integration, one needs to pay attention to use and misuse, function and malfunction, and components or interfaces as well as their failures. The proposed approach seems to be able to help for a quick verification and validation plan in early design phases, and this is a subject to further research.

REFERENCES

[1] M. Rajabalinejad, “System Integration: Challenges and Opportunities for Rail Transport”, System of Systems Engineering Conference, 2018, Paris, France.

[2] C. Perrow, “Normal accidents: Living with high risk technologies”, Princeton University Press, 2011.

[3] D. D. Walden et al., Systems Engineering Handbook - A Guide for System Life Cycle Processes and Activities, International Council on Systems Engineering (INCOSE), 2015.

[4] J. Reason, "Beyond the organizational accident: the need for "error wisdom" on the frontline," Quality and Safety in Health Care, vol. 13, no. suppl_2, pp. ii28-ii33, 2004.

[5] M. Rajabalinejad, "Incorporation of Safety into Design by Safety Cube" in Journal of Industrial and Manufacturing Engineering vol. 12, no. 3, WASET, International Scholarly and Scientific Research & Innovation, 2018.

[6] M. Wagenbuur "How Child Road Deaths Changed the Netherlands". *BBC World Service - Witness programme*. BBC World Service, November, 2013.

[7] "Cycling in the Netherlands", (Press release) The Netherlands: Ministry of Transport, Public Works and Water Management. Fietsberaad (Expertise Centre for Cycling Policy), 2009.

 ISBN: 978-1-61208-698-9