A Formally Verified HOL Algebra for Dynamic Reliability Block Diagrams

Yassmeen Elderhalli, Osman Hasan, and Sofiène Tahar

Department of Electrical and Computer Engineering,
Concordia University, Montréal, QC, Canada

{y_elderh,o_hasan,tahar}@ece.concordia.ca

TECHNICAL REPORT

August 2019

Abstract

Dynamic reliability block diagrams (DRBDs) are introduced to overcome the modeling limitations of traditional reliability block diagrams, such as the inability to capture redundant components. However, so far there is no algebraic framework that allows conducting the analysis of a given DRBD based on its structure function and enables verifying its soundness using higher-order logic (HOL) theorem proving. In this work, we propose a new algebra to formally express the structure function and the reliability of a DRBD with spare constructs based on basic system blocks and newly introduced DRBD operators. We present several simplification properties that allow reducing the structure of a given DRBD. We provide the HOL formalization of the proposed algebra, and formally verify its corresponding properties using the HOL4 theorem prover. This includes formally verifying generic reliability expressions of the spare construct, series, parallel and deeper structures in an extensible manner that allows verifying the reliability of complex systems. Finally, we demonstrate the applicability of this algebra by formally analyzing the terminal reliability analysis of a shuffle-exchange network in HOL4.

1 Introduction

Reliability measures how reliable a system is by finding the probability that it will continue to provide its desirable service in a given period of time. Fault trees (FTs) [1] and reliability block diagrams (RBDs) [2] are the most commonly used reliability modeling techniques. FTs graphically model the sources of failure of a system or subsystem using FT gates. An RBD, on the other hand, is a graphical representation of the reliability of a system. The components of a system are modeled as blocks and are connected using connectors (lines) to create a path or multiple paths from the RBD input to its output. These paths represent the required working blocks (system components) for the system to have a successful operation. The modeled system fails when components fail in such a manner that leads to the disconnection of all the paths between the input and the output. RBDs can be connected in a *series*, *parallel*, *series-parallel* or *parallel-series* fashion to create the appropriate modeling structure depending on the behavior and the components redundancy of the modeled system, which provides flexible and extensible modeling configurations to represent complex systems. However, both of the traditional RBDs and FTs are unable to model the dynamic behavior of system components, where the change of a state of one component can affect the state of other system components.

Dynamic fault trees (DFTs) [1] are proposed as an extension to traditional FTs by introducing DFT gates, such as spare gates. However, the only behavior that is captured by DFTs is the dynamic failure effect of one system component in the failure or activation of other components. To overcome the modeling limitations of DFTs, RBDs are extended to *dynamic reliability block diagrams* (DRBDs) to model the dynamic dependency among system components in several scenarios by

introducing new constructs [3]. These new constructs are basically DRBD blocks that enable modeling dynamic relationships among system components, such as a load sharing construct that captures the effect of sharing a load on the reliability of system components and spare construct that models the reliability of spare parts in a DRBD.

Formal methods have been used in the analysis of RBDs; both the dynamic and static (traditional) ones. For instance, in [4], the formal semantics of DRBD constructs in Object-Z formalism [5] are proposed. However, analyzing and verifying the behavior of DRBDs based on this formalism is not feasible since there is a lack in the support of the used tools. Therefore, in [5], the DRBDs are then converted into a Colored Petri Net (CPN) to be analyzed using Petri nets tools. An algorithm to automatically convert a DRBD into a CPN is also proposed in [6]. Since CPN is used, only some state-based properties of the modeled system can be analyzed. In [7], the HOL4 theorem prover [8] is used to formalize several configurations of static RBDs. However, this formalization can only analyze the combinatorial behavior of systems and cannot provide support to analyze DRBDs. In addition, this formalization cannot be tailored to provide support for DRBDs, and thus, it is required to have a brand new higher-order logic (HOL) formalization to support this kind of analysis.

In system engineering, it is important to be able to analyze DRBDs qualitatively in order to identify the sources of system vulnerability, and quantitatively in order to evaluate the system reliability. However, to the best of our knowledge, so far there exists no algebraic approach that mathematically models a given DRBD and enables expressing its function based on basic components just like the DFT algebra [9]. Using such algebra in the reliability analysis will result in simpler and fewer proof steps than the DFT-based algebraic analysis [9], since the probabilistic principle of inclusion and exclusion will not be invoked. In this work, we propose a new algebraic approach for DRBD analysis that allows having a DRBD expression to be used for both qualitative and quantitative analyses. We introduce new operators to mathematically model the dynamic behavior in DRBD structures and constructs. In particular, we use these operators to model a DRBD spare construct besides traditional series, parallel, series-parallel and parallel-series structures. Moreover, we provide simplification theorems that allow reducing the structure of a given DRBD. This DRBD structure can be then analyzed to obtain a generic expression of the system reliability. The reliability expressions obtained using this approach are generic and independent of the distribution and density functions that represent the system components. Although basic operators, such as OR and AND, were introduced in [3], they are only useful to model parallel and series constructs of dependent components. Moreover, there is no general mathematical expression that would allow reasoning about the behavior of DRBDs. In addition, the DRBDs constructs of [3] are quite complex, which complicates modeling large systems. Therefore, we use the constructs proposed in [4] as they are much simpler, which facilitates defining the new algebra to model various new DRBD constructs. In this work, we use the DRBD constructs of [4]. Leveraging upon the expressive nature of HOL, we formally verify the soundness of the proposed DRBD algebra using HOL theorem proving. Although the formalization development can be conducted using many theorem provers, we choose the HOL4 theorem prover, as our existing formalization of

DFT algebra can be useful since our proposed DRBD algebra is compatible with the DFT's. The work contributions can be summarized as:

- A new DRBD algebra that includes DRBD operators and simplification theorems that allow expressing the structure of a given DRBD.
- A HOL formalization of the introduced DRBD algebra, i.e., modeling the DRBD operators and verifying their simplification theorems using HOL4 to ensure the soundness of the proposed approach.
- A mathematical expression and HOL formalization of the spare construct and its reliability.
- Mathematical models and reliability expressions of the traditional series, parallel and deeper structures for an arbitrary number of inputs using the new DRBD operators with their HOL formalization.
- Formal reliability analysis of two real-world systems

Our ultimate goal is to develop a formally verified algebra that follows the traditional reliability expressions of the series and parallel structures in an easily extensible manner and at the same time can capture the dynamic behavior of real-world systems. Our formalization differs from and overcomes the formalization of traditional RBDs presented in [7] in the sense that it can formally express the structure function of a DRBD using the introduced DRBD operators. In addition, it can formally model and analyze DRBD spare constructs. Furthermore, we model the traditional RBD structures, i.e., series, parallel and deeper structures in a way similar to the mathematical models available in the literature, which makes it easily understood and followed by reliability engineers that are not familiar with HOL theorem proving. Finally, we illustrate the usefulness of the proposed developments in conducting the formal analysis of two real-world systems; the terminal reliability of a shuffle-exchange network and the reliability of a drive-by-wire system.

2 DRBD Algebra

In this section, we present the proposed algebra for DRBD analysis. This algebra allows modeling the structure function of DRBDs with spare constructs. Moreover, we present some simplification properties that enable reducing the structure function when possible. Throughout this work, we assume that system components or blocks are represented by random variables that in turn represent their time-to-failures. In addition, we assume that system components are non-repairable, i.e., we are interested in expressing the reliability of the system considering that the failed components will not be repaired. It is worth mentioning that our proposed algebra follows the general lines for the DFT algebra [9], which allows DFTs conversion into DRBDs for conducting their analysis as well.

The reliability of a single component, which time-to-failure function is represented by random variable X, is mathematically defined as [2]:

$$R_X(t) = Pr\{s \mid X(s) > t\} = 1 - Pr\{s \mid X(s) \leq t\} = 1 - F_X(t) \qquad (1)$$

where $F_X(t)$ is the cumulative distribution function (CDF) of X.

We call $\{s \mid X(s) > t\}$ as a DRBD event as it represents the set that we are interested in finding the probability of until time t:

$$event\ (X,\ t)\ =\ \{s \mid X(s) > t\} \tag{2}$$

2.1 Identity Elements, Operators and Simplification Properties

Similar to the identity elements of ordinary Boolean algebra and DFT algebra [9], we introduce two identity elements, i.e., ALWAYS and NEVER, that represent two states of any system block. The *ALWAYS* element represents a system component that always fails, i.e., it fails from time 0. While the *NEVER* element represents a component that never fails, i.e., the time of its failure is $+\infty$. These identity elements play an important role in the reduction process of the structure functions of DRBDs, as will be introduced in the following sections.

$$ALWAYS = 0 \tag{3}$$

$$NEVER = +\infty \tag{4}$$

We introduce operators to model the relationship between the various blocks in a DRBD. These operators can be divided into two categories: 1) The AND and OR operators that are not concerned with the dependencies among system components. 2) Temporal operators, i.e., *After*, *Simultaneous* and *Inclusive After*, that can capture the dependencies between system components. It is worth mentioning that DRBDs are concerned with modeling the several paths of success of a given system. Therefore, if we are concerned in knowing the success behavior of a DRBD until time t, it means that we are interested in knowing how the system would not fail until time t. As a result, we can use the time-to-failure random variables in modeling the time-to-failure of a given DRBD, i.e., its structure function. It is assumed that for any two system components that possess continuous failure distribution functions, the possibility that these components fail at the same time can be neglected.

In [3], AND and OR operators were introduced to model the parallel and series constructs between dependent components only without providing any mathematical model to these operators. We propose to use the AND ($\cdot$) and OR ($+$) operators to model series and parallel blocks in a DRBD, respectively without any restriction. We provide a mathematical model for each operator based on the time of failure of its inputs, as listed in Table 1, to be used in the proposed algebra. The AND operator models the series connection between two or more system blocks, as shown in Figure 1(a). For example, the DRBD in Figure 1(a) will continue to work only if component X and component Y are working. Once one of these blocks stops working, then there will be no connection between the input and the output of the DRBD and thus the system will no longer work. We model the AND operator as the minimum time of its input arguments. Similarly, the OR operator models the connection between parallel components in a DRBD. For example, the DRBD in Figure 1(b) will continue to work if X is working or Y is working. All the components in a parallel structure should fail

(a) Series DRBD   (b) Parallel DRBD

Figure 1: Two-Block Series and Parallel DRBDs

for this DRBD to fail. Therefore, we model the OR operator as the maximum time of failure of its input arguments, which represents the time of failure of basic system blocks or sub-DRBDs. This approach facilitates using these operators to model even the complex structures.

If X and Y are independent, then the reliability of the systems, shown in Figure 1, can be expressed as in Table 1. To reach these expressions, it is required first to express the DRBD events as the intersection and union for the AND and OR operators, respectively, as:

$$event\ ((X \cdot Y),\ t)\ =\ event\ (X,\ t)\ \cap\ event\ (Y,\ t) \tag{5}$$

$$event\ ((X + Y),\ t)\ =\ event\ (X,\ t)\ \cup\ event\ (Y,\ t) \tag{6}$$

In order to model the dynamic behavior of systems in DRBDs, we introduce new temporal operators: *after* ($\triangleright$), *simultaneous* (Δ), and *inclusive after* ($\trianglerighteq$), as listed in Table 2. The *after* operator represents a situation where it is required to model a component that continues to work after the failure of another. The time of failure of the after operator equals the time of failure of the last component, which is required to fail. However, if the required sequence does not occur, then the output can never fail, i.e., the time of failure equals $+\infty$. The behavior of the simultaneous operator is similar to the one introduced in the DFT algebra [9]. The output of this operator fails if both its inputs fail at the same time, otherwise it can never fail. Finally, the inclusive after operator encompasses the behavior of both the after and simultaneous operators, i.e, it models a situation where it is required that one component continues to work after another one or fail at the same time, otherwise it can never fail. In the case of dealing with basic components, the inclusive after will behave in a similar way as the after operator. Therefore, their probabilities can be expressed for independent

Table 1: Mathematical and Reliability Expressions of AND and OR Operators

Operator	Math. Expression	Reliability
AND	$X \cdot Y = min\ (X, Y)$	$R_{(X \cdot Y)}(t)\ =\ R_X(t)\ \times\ R_Y(t)$
OR	$X + Y = max\ (X, Y)$	$R_{(X+Y)}(t)\ = 1 - ((1 - R_X(t)) \times (1 - R_Y(t)))$

6

Table 2: Mathematical Expressions of Temporal Operators

After($\triangleright$)	Simultaneous(Δ)	Inclusive After($\trianglerighteq$)
$X \triangleright Y = \begin{cases} X, & X > Y \\ +\infty, & X \leq Y \end{cases}$	$X \Delta Y = \begin{cases} X, & X = Y \\ +\infty, & X \neq Y \end{cases}$	$X \trianglerighteq Y = \begin{cases} X, & X \geq Y \\ +\infty, & X < Y \end{cases}$

random variables in the same way as:

$$R_{(X \triangleright Y)}(t) = 1 - \int_0^t f_X(x) \times F_Y(x) \; dx \tag{7}$$

where F_X is the probability density function (PDF) of X and F_Y is the CDF of Y.

We introduce several simplification properties to reduce the structure function of a DRBD. These simplification properties range from simple ones, such as the associativity and idempotence of the operators, to more complex theorems. The idea of these properties is to reduce the algebraic expressions based on the time of failure. For example, $X \cdot ALWAYS = ALWAYS$ means that if a component in a series structure is not working, i.e., always fails, then the series structure is not working as well. Similarly, $X + NEVER = NEVER$ means that if a component in a parallel structure cannot fail, then the whole parallel structure cannot fail as well. $X + Y = Y + X$, $X \cdot Y = Y \cdot X$ and $X \Delta Y = Y \Delta X$ represent the commutativity property for the OR, AND and simultaneous operators, respectively. An example of a more complex theorem is $X \triangleright (Y \cdot Z) = (X \triangleright Y) \cdot (X \triangleright Z)$. In Section 3, a full list of the developed theorems will be introduced.

2.2   DRBD Constructs and Structures

The spare construct, shown in Figure 2 [4], is introduced in DRBDs to model situations where a spare part is activated and replaces the main part, after its failure, by introducing a spare controller to activate the spare [4]. Depending on the failure behavior of the spare part, we can have three variants, i.e., hot, warm and cold ($H|W|C$) spares. The hot spare possesses the same failure behavior in both its active and dormant states. The cold spare cannot fail in its dormant state and is only activated after the failure of the main part. The failure behavior of the warm spare in the dormant state is attenuated by a dormancy factor from the active state. In order to distinguish between the dormant and active states of the spare, just like the DFT algebra [9], we use two different symbols to model the spare part of the DRBD spare construct, one for the dormant state and the other for the active one. For the spare construct of Figure 2, the spare X is represented by X_a and X_d for the active and dormant states, respectively. After the failure (F) of the main part Y, X will be activated (A) by the spare controller. We model the structure function of the spare construct (Q_{spare}) using the DRBD operators based on the description of its behavior as:

$$Q_{spare} = (X_a \triangleright Y) \cdot (Y \triangleright X_d) \tag{8}$$

7

Thus, we need two conditions to be satisfied in order for the spare to work. The first one is that the active state of the spare will continue to work after the failure of the main part $(X_a \triangleright Y)$. The second condition is that the main part will continue to work after the failure of the spare in its dormant state $(Y \triangleright X_d)$. However, since the spare part can only fail in one of its states (X_a, X_d) but not both as it is non-repairable, only one of the terms in Equation (8) affects the behavior and the other term can never fail, i.e., it fails at $+\infty$.

Since the spare construct of the DRBD and the spare gate of the DFT exhibit complementary behavior, i.e., the DRBDs consider the success and the DFTs consider the failure, we can use the probability of failure of the spare DFT gate [9] to find the reliability of the spare DRBD construct. It is assumed that the dormant spare and the main part are independent since the failure of one does not affect the failure of the other. However, the failure of the active spare is affected by the time of failure of the main part, since it will be activated after the failure of the main part. We express the reliability of the spare as:

$$R_{spare}(t) = 1 - \int_0^t \int_y^t f_{(X_a|Y=y)}(x)\ f_Y(y)dxdy - \int_0^t f_Y(y)F_{X_d}(y)dy \qquad (9)$$

where $f_{(X_a|Y=y)}$ is the conditional density function of X_a given that Y failed at time y. Equations (8) and (9) represent the general behavior of the spare, i.e., the warm spare. The hot and cold spares represent special cases of the warm spare and can be expressed as:

$$Q_{hotspare} = X + Y \qquad (10)$$

$$Q_{coldspare} = X_a \triangleright Y \qquad (11)$$

In Equation (10), the spare part X has the same behavior in both states and thus there is no need to use any subscript to distinguish both states. The probability of Equation (10) can be expressed using the reliability of the OR operator, given in Table 1. While the reliability of the cold spare construct can be expressed as:

$$R_{cold\ spare}(t) = 1 - \int_0^t \int_y^t f_{(X_a|Y=y)}(x)\ f_Y(y)\ dx\ dy \qquad (12)$$

Table 3 lists the mathematical and reliability expressions of these structures [2]. The series structure represents a collection of blocks that are connected in series, as shown in Figure 3(a). The system continues to work until the failure of one of these blocks. We define a series structure that represents the intersection of all events of the blocks in this structure as in Table 3, where X_i represents the ith block in the series structure and n is the number of blocks. Interestingly, any block in our proposed algebra can
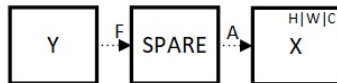

Figure 2: Spare Construct

8

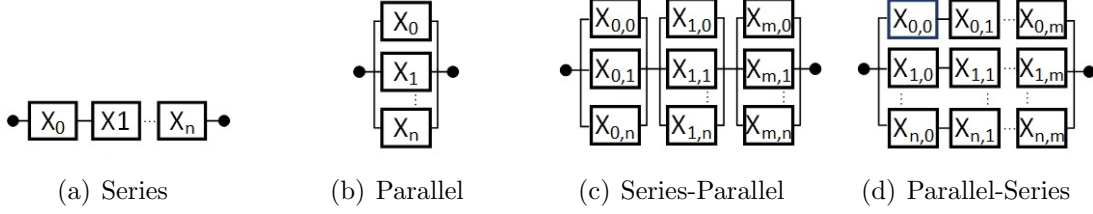(a) Series (b) Parallel (c) Series-Parallel (d) Parallel-Series

Figure 3: DRBD Structures

represent a basic system component or a complex structure, such as a spare construct. Moreover, since we are dealing with the events, we can use the ordinary reliability expressions for the series structure assuming the independence of the individual blocks. The parallel structure, shown in Figure 3(b), represents a system that continues to work until the failure of the last block in the structure. The behavior of the parallel structure can be expressed using the OR operator. We represent the parallel structure as the union of the individual events of the blocks. The series-parallel structure, shown in Figure 3(c), represents a series structure, where the blocks of the series structure are parallel structures. The structure function of this structure can be expressed using AND of ORs operators. Table 3 lists the model for this structure with its reliability expression, where n is the number of blocks in the parallel structure and m is the number of parallel structures that are connected in series. The parallel-series structure represents a group of series structures that are connected in parallel, as shown in Figure 3(d). Its structure function can be expressed using OR of ANDs operators.

3 Formalization of DRBDs in HOL

In this section, we present our formalization for the proposed DRBD algebra including DRBD events, operators and constructs, simplification theorems and reliability expressions. First, we review some HOL probability theory preliminaries required for understanding the rest of the work.

Table 3: Mathematical and Reliability Expressions of DRBD Structures

Structure	Math. Expression	Reliability
Series	$\bigcap_{i=1}^{n}(event\ (X_i,\ t))$	$\prod_{i=1}^{n} R_{X_i}(t)$
Parallel	$\bigcup_{i=1}^{n}(event\ (X_i,\ t))$	$1 - \prod_{1=1}^{n}(1 - R_{X_i}(t))$
Series-Parallel	$\bigcap_{i=1}^{m}\bigcup_{j=1}^{n}(event\ (X_{(i,j)},\ t))$	$\prod_{i=1}^{m}(1 - \prod_{j=1}^{n}(1 - R_{X_{(i,j)}}(t)))$
Parallel-Series	$\bigcup_{i=1}^{n}\bigcap_{j=1}^{m}(event\ (X_{(i,j)},\ t))$	$1 - (\prod_{i=1}^{n}(1 - \prod_{j=1}^{m}(R_{X_{(i,j)}}(t))))$

3.1   HOL Probability Theory

The probability space is defined in HOL as a measure space, where the measure (probability) of the entire space is 1. A probability space is a measure space and is defined as a triplet $(\Omega, \mathcal{A}, \mathcal{P}r)$, where Ω is the space, $\mathcal{A}$ are the probability events and $\mathcal{P}r$ is the probability [10]. Two functions are defined in HOL; `p_space p` and `events p`, that return the space (Ω) of the above triplet and the events $(\mathcal{A})$, respectively. A random variable is a measurable function that maps the probability space p to another space. It is defined in HOL as [10]:

Definition 3.1. $\vdash \forall$X p s.   random_variable X p s $\Leftrightarrow$
prob_space p $\wedge$ X $\in$ measurable (p_space p, events p) s

where X is the random variable, p is the probability space and s is the space that the random variable maps to. In our work, we use the `borel` space, which is defined over the real line [11].

For a random variable X, the probability distribution is defined as the probability that this random variable belongs to a certain set [12]:

Definition 3.2.
$\vdash \forall$p X. distribution p X = (λs.   prob p (PREIMAGE X s $\cap$ p_space p))

The cumulative distribution function (CDF) is defined as [13]:

Definition 3.3. $\vdash \forall$p X t.   CDF p X t = distribution p X $\{$y $\mid$ y $\leq$ (t:real)$\}$

Table 4: HOL4 Probability Functions

Function	Explanation
rv_gt0_ninfinity L	Random variables in list L are greater than 0 and not equal to $+\infty$
indep_var p lborel (real o X) lborel (real o Y)	Independence of random variables defined from the probability space p to the Lebesgue Borel measure (lborel)
distributed p lborel (real o X) f_x	Defines a density function f_x for the real version of random variable X defined from the probability space p to the Lebesgue-Borel measure
measurable_CDF p (real o Y)	Ensures that CDF (F_Y) is measurable
cont_CDF p (real o Y)	Ensures that CDF (F_Y) is continuous
cond_density lborel lborel p (real o X)(real o Y) y f_{xy} f_y $f_{X_a\mid Y}$	Defines a conditional density function $f_{X_a\mid Y}$ using the joint density function f_{xy} and the marginal density function f_y
den_gt0_ninfinity f_{X_aY} f_Y $f_{X_a\mid Y}$	Ensures the proper values for the density functions; joint, marginal and conditional, respectively. $0 \leq f_{X_aY}$, $0 < f_Y$ and $0 \leq f_{X_a\mid Y}$
indep_sets p X s	Ensures that the group of sets X indexed by the numbers in set s are independent over the probability space p

where `p` is a probability space, `X` is a real-valued random variable and `t` is a variable of type real and represents time.

Independence of random variables is an important property that ensures that the probability of the intersection of the events of these random variables equals the product of the individual events. This definition is ported from Isabelle/HOL [14] to HOL4 as [11]:

Definition 3.4. ⊢ `indep_vars p M X ii =`
(∀i. i ∈ ii ⇒
 `random_variable (X i) p (m_space (M i), measurable_sets (M i)))` ∧
`indep_sets p`
 (λi. {PREIMAGE f A ∩ p_space p|(f=X i) ∧ A ∈ measurable_sets (M i)}) ii

This definition ensures that a group X is composed of random variables indexed by the elements in set ii and that the events represented by the preimage of these random variables are independent using `indep_sets`. `indep_var` is defined, based on Definition 3.4, to capture the behavior of independence for two random variables [11].

Finally, the Lebesgue integral is defined in HOL4 based on positive simple functions and then extended for positive functions and functions with positive and negative values [10]. Throughout this work, we use the Lebesgue integral for positive functions, i.e., `pos_fn_integral`, since we are integrating cumulative distribution and probability density functions, which are always positive. The integration is over the real line and thus we use the Lebesgue-Borel measure (`lborel`) [11] for this purpose. The boundaries of this integral can be identified using an indicator function by specifying the set of elements used in the integration. For example, $\int_A f dx$ can be represented as `pos_fn_integral lborel` (λx. `indicator_fn A * f x`). However, for the ease of understanding, we use the regular mathematical expressions, i.e., we use $\int f \, dx$ to express integrals instead. Table 4 lists the probability theory functions used in the rest of the work [13].

3.2 DRBD Event

In our formalization, we define the inputs, or the random variables representing the time to failure of system components, as lambda abstracted functions with a return datatype of extended-real, which represents real numbers besides $\pm\infty$.

We define the DRBD event of Equation (2) as:

Definition 3.5.
⊢ ∀p X t. `DRBD_event p X t = {s | Normal t < X s}` ∩ `p_space p`

where `Normal` typecasts the real value of `t` from real to extended-real. This type conversion is required since we need real-valued random variables. However, we need to deal with the extended-real datatype to model the `NEVER` element. Therefore, we define the time-to-failure functions to return extended-real and typecast the values from extended-real to real using the function `real` and vice versa using `Normal`.

We define the reliability as the probability of the DRBD event according to Equation (1):

Definition 3.6. ⊢ ∀p X t. Rel p X t = prob p (DRBD_event p X t)

We verify the relationship between the reliability and the CDF of Equation (1) as:

Theorem 3.1. ⊢ ∀p X t. rv_gt0_ninfinity [X] ∧
  random_variable (real o X) p borel ⇒
  (Rel p X t = 1- CDF p (real o X) t)

where `real` typecasts the values of the random variable from extended-real to real as
the CDF is defined for real-valued random variables, `random_variable (real o X) p
borel` ensures that `(real o X)` is a random variable over the real line represented by
the `borel` space, and `rv_gt0_ninfinity` ensures that the random variable is greater
than or equal to 0 and not equal to $+\infty$, as described in Table 4, which means that the
time of failure of any component cannot be negative or $+\infty$. Theorem 3.1 is verified
based on the fact that the `DRBD_event` and the set of the CDF are the complement of
each other. Therefore, the probability of one of them equals one minus the other. For
the rest of the work, we will denote `CDF p (real o X) t` by $F_X(t)$ to facilitate the
understanding of the theorems.

3.3 Identity Elements, Operators and Simplification Theorems

Our formalization of the identity elements and the DRBD operators is listed in Table 5,
where `extreal` is the extended-real datatype in HOL4, `PosInf` represents $+\infty$, `min`
and `max` are HOL functions that return the minimum and maximum values of their
arguments, respectively. This formalization follows the proposed definitions in Tables
1 and 2. However, we define the operators as lambda abstracted functions to be able
to conduct the probabilistic analysis later. In addition, we verify several simplification
theorems based on the properties of `extreal` numbers in HOL and the definitions of
the DRBD operators. For example, the following theorem represents the distributive
property of the after operator over the AND:

Theorem 3.2. ⊢ ∀X Y Z. X ▷ (Y · Z) = (X ▷ Y)·(X ▷ Z)

Table 6 lists the simplification theorems that we developed and verified in the
proposed algebra.

In order to verify the reliability of the DRBD constructs, such as the spare, we
need first to verify the reliability of the DRBD operators that are used to express the
structure function of these constructs. For the AND and OR operators, we verify their
reliability expressions as in Theorems 3.3 and 3.4, respectively.

Theorem 3.3. ⊢ ∀p X t. rv_gt0_ninfinity [X;Y] ∧
    indep_var p lborel (real o X) lborel (real o Y) ⇒
    (Rel p (X·Y) t = Rel p X t * Rel p Y t)

Theorem 3.4. ⊢ ∀p X t. rv_gt0_ninfinity [X;Y] ∧
    indep_var p lborel (real o X) lborel (real o Y) ⇒
    (Rel p (X + Y) t = 1 - (1 - Rel p X t) * (1 - Rel p Y t))

12

Table 5: Definitions of Identity Elements and DRBD Operators

Element/Operator	Mathematical Expression	Formalization
Always element	$ALWAYS = 0$	$\vdash$ R_ALWAYS = (λs. (0:extreal))
Never element	$NEVER = +\infty$	$\vdash$ R_NEVER = (λs. PosInf)
AND	$X \cdot Y = min(X, Y)$	$\vdash \forall$X Y. R_AND X Y =(λs. min (X s) (Y s))
OR	$X + Y = max(X, Y)$	$\vdash \forall$X Y. R_OR X Y = (λs. max (X s) (Y s))
After	$X \rhd Y = \begin{cases} X, & X > Y \\ +\infty, & X \leq Y \end{cases}$	$\vdash \forall$X Y. R_AFTER X Y = (λs. if Y s < X s then X s else PosInf)
Simultaneous	$X \Delta Y = \begin{cases} X, & X = Y \\ +\infty, & X \neq Y \end{cases}$	$\vdash \forall$X Y. R_SIMULT X Y = (λs. if X s = Y s then X s else PosInf)
Inclusive After	$X \unrhd Y = \begin{cases} X, & X \geq Y \\ +\infty, & X < Y \end{cases}$	$\vdash \forall$ X Y. R_INCLUSIVE_AFTER X Y = (λs. if Y s $\leq$ X s then X s else PosInf)

We verify Theorem 3.3 by first rewriting using Definition 3.6. Then, we prove that DRBD_event of the AND operator equals the intersection of the individual events, as in Equation (5). Utilizing the independence of the real-valued random variables `real o X` and `real o Y`, the probability of intersection of their events equals the product of the probability of the individual events. Since X and Y are greater than 0 and are not equal to $+\infty$, based on the function `rv_gt0_ninfinity`, the events in the probability space that correspond to X and Y are equal to the ones that correspond to `real o X` and `real o Y`. As a result, the DRBD_events of X and Y are independent. Hence, the probability of their intersection equals the product of the probability of the individual

Table 6: Formally Verified Simplification Theorems

Simplification Theorem
$\vdash \forall$X. ($\forall$s. 0 $\leq$ X s) $\Rightarrow$ (X $\cdot$ R_ALWAYS = R_ALWAYS)
$\vdash \forall$X Y Z. (X $\cdot$ Y) $\cdot$ Z = X $\cdot$ (Y $\cdot$ Z)
$\vdash \forall$X Y. X $\cdot$ Y = Y $\cdot$ X
$\vdash \forall$X. X $\cdot$ X = X
$\vdash \forall$X. X $\cdot$ R_NEVER = X
$\vdash \forall$X. ($\forall$s. 0 $\leq$ X s) $\Rightarrow$ (X + R_ALWAYS = X)
$\vdash \forall$X Y Z. (X + Y) + Z = X + (Y + Z)
$\vdash \forall$X Y. X + Y = Y + X
$\vdash \forall$X. X + X = X
$\vdash \forall$X. X + R_NEVER = R_NEVER
$\vdash \forall$X Y. X + (X $\cdot$ Y) =X
$\vdash \forall$X Y Z. X rhd (Y $\rhd$ Z) = ((X $\rhd$ Y) + (X $\rhd$ Z)) (Y $\rhd$ Z)
$\vdash \forall$X Y. (X $\rhd$ Y) + (Y $\rhd$ X) = R_NEVER
$\vdash \forall$X Y Z. X $\rhd$ (Y $\cdot$ Z) = (X $\rhd$ Y) $\cdot$ (X $\rhd$ Z)
$\vdash \forall$X Y Z. X $\cdot$ (Y + Z) = (X $\cdot$ Y) + (X $\cdot$ Z)
$\vdash \forall$X Y Z. X + (Y $\cdot$ Z) = (X + Y) $\cdot$ (X + Z)
$\vdash \forall$X Y. X $\unrhd$ Y = (X $\rhd$ Y) $\cdot$ (X Δ Y)
$\vdash \forall$X Y Z. X $\rhd$ (Y + Z) = (X $\rhd$ Y) + (X $\rhd$ Z)
$\vdash \forall$X Y. X Δ Y = Y Δ X

events, i.e., their reliability. Theorem 3.4 is verified in a similar way. However, we prove that the `DRBD_event` of the OR operator equals the union of the individual events, as in Equation (6). We verify that this union of events equals to the complement of the intersection of the complements of the individual events. Now, Theorem 3.4 can be proven using the independence of random variables.

We extend the definition of the AND and OR operators to n-ary operators, `nR_AND` and `nR_OR`, that can be used to represent the relationship between an arbitrary number of elements. We formally define n-ary AND (`nR_AND`) as:

Definition 3.7.
⊢ ∀X s. nR_AND X s = ITSET (λe acc. R_AND (X e) acc) s R_NEVER

where `ITSET` is the HOL function to iterate over sets. This definition applies the `R_AND` over the elements of `X` indexed by the numbers in `s`. `R_NEVER` is the identity element of the `R_AND` operator.

Similarly, we formally define n-ary OR (`nR_OR`) as:

Definition 3.8.
⊢ ∀X s. nR_OR X s = ITSET (λe acc. R_OR (X e) acc) s R_ALWAYS

where `R_ALWAYS` is the identity element of the `R_OR` operator. The reliability of these two operators would be similar to the reliability of the series and parallel structures, respectively, as will be described in the following section.

Finally, we verify the reliability expression of the after operator utilizing our formalization in [13], where the description of the assumptions is listed in Table 4:

Theorem 3.5. ⊢ ∀X Y p f_x t. rv_gt0_ninfinity [X; Y] ∧ 0 ≤ t ∧
    indep_var p lborel (real o X) lborel (real o Y) ∧
    distributed p lborel (real o X) f_x ∧ (∀x. 0 ≤ f_x x) ∧
    cont_CDF p (real o Y) ∧ measurable_CDF p (real o Y) ⇒
    (Rel p (X▷Y) t = 1- $\int_0^t f_X(x) \times F_Y(x)\ dx$)

The proof of this theorem is based on $Pr(Y < X < t) = \int_0^t f_X(x) \times F_Y(x)\ dx$, which has been verified in [13] using the properties of the Lebesgue integral and independence of random variables. The DRBD *after* operator represents a situation where the system continues to work until two components fail in sequence. Thus, the above expressions allow us to verify the reliability expression of the *after* operator, as the DRBD and DFT events complement one another.

3.4   DRBD Constructs and their Reliability Expressions

As mentioned previously, the spare construct can have three variants according to the type of the spare block. We formally define the generic case, i.e., the warm spare (WSP) as:

Definition 3.9. ⊢ ∀Y X_a X_d. R_WSP Y X_a X_d = (X_a ▷ Y) · (Y ▷ X_d)

Since the DRBD and DFT events complement one another, we use our formalization of the probability of failure of the warm spare gate [13] to verify the reliability of the WSP construct:

Theorem 3.6. $\vdash \forall$p Y X$_a$ X$_d$ t f$_Y$ f$_{X_aY}$ f$_{X_a|Y}$. $\quad$ 0 $\leq$ t $\wedge$
$\quad$ ($\forall$s. $\quad$ ALL_DISTINCT [X$_a$ s; X$_d$ s; Y s]) $\wedge$ DISJOINT_WSP Y X$_a$ X$_d$ t $\wedge$
$\quad$ rv_gt0_ninfinity [X$_a$; X$_d$; Y] $\wedge$ den_gt0_ninfinity f$_{X_aY}$ f$_Y$ f$_{X_a|Y}$ $\wedge$
$\quad$ ($\forall$y. $\quad$ cond_density lborel lborel p (real o X$_a$)(real o Y) y f$_{X_aY}$ f$_Y$ f$_{X_a|Y}$) $\wedge$
$\quad$ indep_var p lborel (real o X$_d$) lborel (real o Y) $\wedge$
$\quad$ cont_CDF p (real o X$_d$) $\wedge$ measurable_CDF p (real o X$_d$) $\Rightarrow$
$\quad$ (Rel p (R_WSP Y X$_a$ X$_d$) t) =
$\quad$ 1 $-$ $\left(\int_0^t f_Y(y) * \left(\int_y^t \text{f}_{(X_a|Y=y)}(x) \text{ dx} \right) \text{ dy} + \int_0^t \text{f}_Y(y) F_{X_d}(y) dy \right)$

where ALL_DISTINCT ensures that the main and spare parts cannot fail at the same time, DISJOINT_WSP Y X$_a$ X$_d$ t ensures that until time t, the spare can only fail in one of its states and den_gt0_ninfinity ascertains the proper values of the density functions; joint (f_{XY}), marginal (f_Y) and conditional ($f_{X_a|Y}$). The description of these assumptions is listed in Table 4 [13]. More details about the formal definitions of these functions can be found in [13]. Theorem 3.6 is verified by first defining a conditional density function f$_{X_a|Y}$ for random variables (real o X$_a$) and (real o Y). This is required as the failure of the spare part is affected by the time of failure of the main part. Therefore, it is required to define this conditional density function then prove the expression based on the probability of failure of the DFT spare gate, which is verified based on the properties of the Lebesgue integral. We formally define the cold spare construct (CSP), which is a special case of the WSP, as:

Definition 3.10. $\vdash \forall$Y X. R_CSP Y X = (λs. $\quad$ if Y s < X s then X s else PosInf)

This definition means that the CSP construct will continue to work until the main part fails then the spare part is activated and fails in its active state. It is worth noting that since the spare part has only one state that affects the behavior of the CSP, which is the active state, therefore, we do not use any subscript with the active state, as the dormant state has no effect here in the behavior. We verify the reliability of the CSP construct based on the probability of failure of the CSP gate as [13]:

Theorem 3.7. $\vdash \forall$p X Y f$_{XY}$ f$_Y$ f$_{X|Y}$ t. $\quad$ 0 $\leq$ t $\wedge$
$\quad$ rv_gt0_ninfinity [X; Y] $\wedge$ den_gt0_ninfinity f$_{XY}$ f$_Y$ f$_{X|Y}$ $\wedge$
$\quad$ ($\forall$y. $\quad$ cond_density lborel lborel p (real o X)(real o Y) y f$_{XY}$ f$_Y$ f$_{X|Y}$) $\wedge$
$\quad$ $\left(\text{Rel p (R\_CSP Y X) t}\right)$ = 1 $-$ $\left(\int_0^t f_Y(y) * \left(\int_y^t \text{f}_{(X|Y=y)}(x) \text{ dx} \right) \text{ dy} \right)$

The conditions required for this theorem are similar to the ones of Theorem 3.6, as the WSP exhibits the behavior of the CSP if the main part fails before the spare.

Finally, we define the hot spare construct (HSP) as:

Definition 3.11. $\vdash \forall$Y X. R_HSP Y X = (λs. $\quad$ max (Y s) (X s))

This means that the HSP acts like the OR operator, where at least one of the main or the spare parts should continue to work for the HSP construct to maintain

15

its successful behavior. Therefore, we can use Theorem 3.3 to express the reliability of the HSP construct.

We formally define the series structure as:

Definition 3.12. $\vdash \forall$Y s. `DRBD_series` Y s = $\bigcap\limits_{i \in s}$ (Y i)

We define the series structure as a function that accepts a group of sets, Y, that are indexed by the numbers in set s and returns the intersection of these sets.

The parallel structure is defined in a similar way but it returns the union of the sets rather than the intersection. We formally define it as:

Definition 3.13. $\vdash \forall$Y s. `DRBD_parallel` Y s = $\bigcup\limits_{i \in s}$ (Y i)

The group of sets, Y, in both structures represents a family of events, i.e, Y will be instantiated later with DRBD events. The reliability expressions of the series and parallel structures are given in Table 3. We verify these expressions as:

Theorem 3.8.
$\vdash \forall$p X t s. s $\neq$ {} $\wedge$ FINITE s $\wedge$
 indep_sets p (λi. {rv_to_event p X t i}) s $\Rightarrow$
 (prob p (DRBD_series (rv_to_event p X t) s) =
  Normal ($\prod\limits_{i \in s}$ (real (Rel p (X i) t))))

Theorem 3.9.
$\vdash \forall$p X t s. s $\neq$ {} $\wedge$ FINITE s $\wedge$
 indep_sets p (λi. {rv_to_event p X t i}) s $\wedge$
 ($\forall$i. i $\in$ s $\Rightarrow$ rv_to_event p X t i $\in$ events p) $\Rightarrow$
 (prob p (DRBD_parallel (rv_to_event p X t) s) =
  1 - Normal ($\prod\limits_{i \in s}$ (real (1 - Rel p (X i) t))))

where s$\neq${} $\wedge$ FINITE s ensures that the set of indices, s, is nonempty and finite. The reliability of the series structure is verified based on the independence of the input events using `indep_sets`, which ensures that for the probability space p, the given group of sets ((λi. {rv_ti_event p X t i}) indexed by the numbers in set s are independent, as described in Table 4. The family of sets ((λi. {rv_ti_event p X t i}) represents the DRBD events of the group of time-to-failure functions, X. This is defined as:

Definition 3.14. $\vdash \forall$p X t. rv_to_event p X t = (λi. DRBD_event p (X i) t)

The function `rv_to_event` enables us to create the group of `DRBD_event` of time-to-failure functions of system blocks (X). Based on the independence of these sets and the definition of the series structure (intersection of sets), we verify that the probability of the series structure is equal to the product of the reliability of the individual blocks (Rel p (X i) t), where i$\in$s. The product function ($\prod$) in HOL4 returns a real value and the probability returns `extreal`, therefore, it is required to typecast the product function to

`extreal` using `Normal`. Similarly, the product function finds the product of real-valued functions, therefore, it is required to typecast the reliability function(`Rel`) to real using the `real` function. The parallel structure is verified in a similar way. We replace the parallel structure (the union of events) with the complement of the intersection of the complements of the events. Then, we verify that the probability of this complement equals one minus the probability of the intersection of the complements. This requires the added condition that all DRBD events created using `rv_to_event` belong to the events of the probability space `p`.

In order to express the series and parallel structures using DRBD operators, we verify that these structures are equal to the DRBD events of the `nR_AND` and `nR_OR`, respectively:

Theorem 3.10. ⊢ ∀p X t s. FINITE s ∧ s ≠ {} ⇒
  (DRBD_event p (nR_AND X s) t = DRBD_series (rv_to_event p X t) s)

Theorem 3.11. ⊢ ∀p X t s. FINITE s ∧ 0 ≤ t ⇒
  (DRBD_event p (nR_OR X s) t = DRBD_parallel (rv_to_event p X t) s)

We verify Theorems 3.10 and 3.11 by inducting on set `s` using `SET_INDUCT_TAC` that will create two subgoals to be solved; one for the empty set and another one for inserting an element to a finite set. Furthermore, we use the fact that the DRBD event of the AND and OR operators equal the intersection and the union of the individual events, respectively. For Theorem 3.11, an additional condition is required, `0≤t`, to be able to manipulate the sets and reach the final form of the theorem.

Interestingly, these structures can be easily extended to model and verify more complex structures, such as two-level structures, i.e., series-parallel and parallel series structures. We formally verify the reliability of the series-parallel structure as:

Theorem 3.12. ⊢ ∀p X t s J.
 indep_sets p
  (λi. {rv_to_event p X t i}) ($\bigcup\limits_{j \in J}$ (s j)) ∧
 (∀i. i ∈ J ⇒ s i ≠ {} ∧ FINITE (s i)) ∧
 FINITE J ∧ J ≠ {} ∧ disjoint_family_on s J ⇒
 (prob p
  (DRBD_series
   (λj. DRBD_parallel
    (rv_to_event p X t) (s j)) J) =
 Normal
  ($\prod\limits_{j \in J}$ (1 - $\prod\limits_{i \in (s\ j)}$ (real (1 - Rel p (X i) t)))))

We formally verify the reliability of the parallel-series structure as:

Theorem 3.13.
$\vdash \forall$p X t s J.
 indep_sets p
  (λi. {rv_to_event p X t i}) ($\bigcup_{j \in J}$ (s j)) $\wedge$

 ($\forall$i. i $\in \bigcup_{j \in J}$ (s j) $\Rightarrow$ rv_to_event p X t i $\in$ events p) $\wedge$

 ($\forall$i. i $\in$ J $\Rightarrow$ s i $\neq$ {} $\wedge$ FINITE (s i)) $\wedge$
 FINITE J $\wedge$ J $\neq$ {} $\wedge$ disjoint_family_on s J $\Rightarrow$
 (prob p
  (DRBD_parallel
   (λj. DRBD_series
    (rv_to_event p X t) (s j)) J) =
 1 -
 Normal
  ($\prod_{j \in J}$ (1 - $\prod_{i \in (s\ j)}$ (real (Rel p (X i) t)))))

The main idea in building these two-level structures is to partition the family of blocks into distinct groups, where we use a set, J, to index these partitions, i.e., it includes the number of groups in the first top level. Then, for each group in this top level, we have another set, {s j| j $\in$ J}, that includes the indices of the blocks in the second level, i.e. the subgroups. For example, consider the parallel-series structure of Figure 3(d), if $n = m = 1$, then the outer parallel structure has two series structures, where each series structure has two blocks. Thus, J = {0;1}. For each j$\in$J, we have a certain set s j that has the indices of the blocks in the inner series structure. Thus, s = (λj. if j = 0 then {0;1} else {2;3}). The same concept is applied to the series-parallel structure. Therefore, the structure of the DRBD can be determined based on the given sets of indices.

We verify Theorems 3.12 and 3.13 by extending the proofs of the series and parallel structures. However, it is required to deal with the intersection of unions in case of the series-parallel structure and the union of intersections in case of parallel-series structure. Therefore, we need to extend the independence of sets properties to include the independence of union and intersection of partitions of the events. We verify these properties as:

Theorem 3.14. $\vdash \forall$p s J Y. indep_sets p (λi. {Y i}) $\bigcup_{j \in J}$ (s j) $\wedge$ J $\neq$ {} $\wedge$
  ($\forall$i. i $\in$ J $\Rightarrow$ countable (s i)) $\wedge$ FINITE J $\wedge$ disjoint_family_on s J $\Rightarrow$
  indep_sets p (λj. {$\bigcup_{i \in s\ j}$ (Y i)}) J

Theorem 3.15. $\vdash \forall$p s J Y. indep_sets p (λi. {Y i}) $\bigcup_{j \in J}$ (s j) $\wedge$ J $\neq$ {} $\wedge$
  ($\forall$i. i $\in$ J $\Rightarrow$ countable (s i) $\wedge$ s i $\neq$ {}) $\wedge$ FINITE J $\wedge$
  disjoint_family_on s J $\wedge$ ($\forall$i. i $\in \bigcup_{i \in J}$ (s j) $\Rightarrow$ Y i $\subset$ m_space p) $\Rightarrow$
  indep_sets p (λj. {$\bigcap_{i \in s\ j}$ (Y i)}) J

where set J includes the indices of the partitions and s has the indices of the individual blocks of each partition, disjoint_family_on ensures that the indices of the blocks in different partitions are disjoint and indep_sets p (λi. {Y i}) $\bigcup_{j \in J}$ (s j) ensures

18

the independence of the family of blocks $\{$Y i$\}$ where the indices of the individual blocks are given by the union of s. In order to verify Theorems 3.14 and 3.15, we need the fact that the σ-algebras generated by $(\lambda$j. $\bigcup_{i\in s\ j}\{$Y i$\})$ with index set J are independent. Then we verify that $\forall$j. j $\in$ J, set $\{\bigcup_{i\in s\ j}\{$Y i$\}\}$ is a subset of the σ-algebra generated by $\bigcup_{i\in s\ j}\{$Y i$\}$. Finally, based on these intermediate verified steps and the definition of indep_sets, we are able to verify these theorems.

In order to verify the reliability of the series-parallel structure, we need to ensure the independence of the individual blocks. Therefore it is required to combine the indices of all blocks into a single set using $\bigcup_{j\in J}$ (s j) to be used with indep_sets. To be able to use the reliability of the series structure in this proof, we use Theorem 3.14 to verify the independence of the unions of partitions of events. This means verifying that the parallel structures are independent, i.e., the probability of intersection of these parallel structures equals the product of the reliability of the parallel structures. Finally, several assumptions related to sets $\{$s i$\mid$ i $\in$ J$\}$ and J are required, which include that these sets are finite and nonempty. Finally, it is required that every block has a unique index, which is ensured using disjoint_family_on. The reliability of the parallel-series structure is verified in a similar manner based on the reliability of the parallel structure. We verify the independence of the intersection of partitions of events rather than the union using Theorem 3.15. In addition, it is required that all DRBD events belong to the events of the probability space.

We extend the reliability of the two-level series-parallel structure to verify the reliability of a more nested structure, i.e., series-parallel-series-parallel, as:

Theorem 3.16. $\vdash \forall$p X t s L A J.
```
(∀i.  i ∈ nested_BIGUNION s L A J ⇒ rv_to_event p X t i ∈ events p) ∧
indep_sets p (λi.  {rv_to_event p X t i}) (nested_BIGUNION s L A J) ∧
sets_finite_not_empty s L A J ⇒
(prob p
 (DRBD_series (λj.
   DRBD_parallel (λa.
    DRBD_series (λl.
     DRBD_parallel (rv_to_event p X t) (s l)) (L a)) (A j)) J) =
 Normal
```
$(\prod_{j\in J}$
$\quad$ (1 - $\prod_{a\in(A\ j)}$(1 - $\prod_{l\in(L\ a)}$ (1 - $\prod_{i\in(s\ l)}$(real (1 - Rel p (X i) t)))))))))

For this four-level nested structure, we have four sets (indexed sets) that determine the structure of the DRBD, which are: J, A, L and s. This is similar to the two-level nested structure but with a deeper hierarchy. Therefore, in order to combine the indices of all the individual blocks in the DRBD in a single set, we define nested_BIGUNION s L A J to union the elements of all s i, where i$\in$ L a, a$\in$ A j and j$\in$J. This is done in a hierarchical manner and can be extended easily to deeper levels. We use the previously mentioned function to ensure that all the individual events belong to the probability events and are independent as well. Moreover, it is required to ensure that the sets are finite, disjoint and nonempty, just like the series-parallel structure. We combine these set-related conditions using the function sets_finite_not_empty.
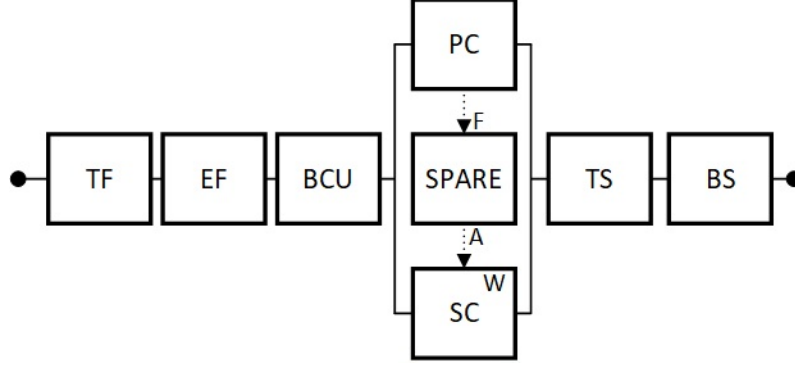
19

Figure 4: DRBD of drive-by-wire system

Finally, we verify Theorem 3.16 within two main steps. The first step is to verify the reliability of the outer series-parallel, which requires verifying the independence of the intersection of union of partition of the DRBD blocks, i.e., the inner series-parallel structures are independent. The second step is to verify the reliability of the inner series-parallel structures, which can be done based on some set manipulation. This theorem can be used to verify even deeper structures, which would require verifying the independence of more nested structures. We use Theorem 3.16 to verify the reliability of the series-parallel-series structure as it represents a special case of the series-parallel-series-parallel, where each of the innermost parallel structures has only one block. Our formalization follows the natural definitions of parallel and series structures. Moreover, our verified lemmas of independence allow verifying deeper structures, which makes our formalization flexible and applicable to model the most complex systems. The proof script, which is available at [15], of our formalization required around 3200 lines. In the following section, we utilize our formalization in the verification of the reliability of two real-world systems.

4 Applications

To demonstrate the applicability of our proposed DRBD algebra, we present the formal reliability analysis of a drive-by-wire system (DBW) [16] and a shuffle-exchange network (SEN) [17] to verify generic expressions that are independent of the failure distribution of the system components, i.e., we can use different types of distributions to model the failure of system components as long as they satisfy the required conditions, such as the continuity.

The DRBD of the DBW system, shown in Figure 4, models the successful behavior of a key part of the modern automotive industry. This system controls the functionality of the vehicle using a computerized controller. We provide the analysis of the throttle and brake subsystems. The throttle subsystem continues to work as long as the throttle (TF) and the engine (EF) are working. In addition, the system successful operation requires the operation of the brake control unit (BCU). The system includes a primary control unit (PC) with a warm spare (SC) that replaces the main part after failure. Finally, the system needs the operation of the throttle sensor (TS) and the brake sensor
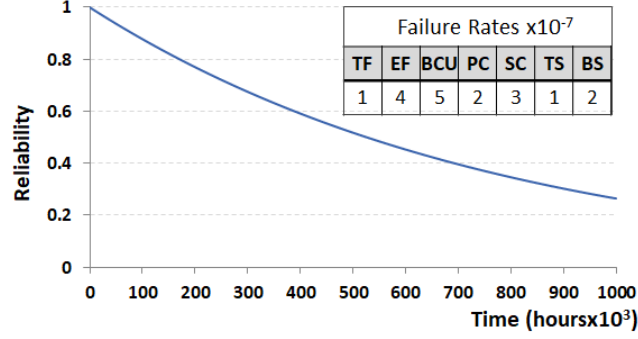
Figure 5: Reliability of DBW system

(BS). The DRBD of this system is modeled as a series structure with a spare construct. We express the structure function of this DRBD using our operators:

$$Q_{DBW} = TF \cdot EF \cdot BCU \cdot (R\_WSP\ PC\ SC_a\ SC_d) \cdot TS \cdot BS$$

Then we verify the DBW reliability as:

Theorem 1. $\vdash \forall p$ TF EF BCU PC SC_a SC_d TS BS t.
 DBW_set_req p TF EF BCU PC SC_a SC_d TS BS t $\Rightarrow$
 (prob p (DRBD_event p Q_{DBW} t) =
 Rel p TF t * Rel p EF t * Rel p BCU t * Rel p (R_WSP PC SC_a SC_d) t *
 Rel p TS t * Rel p BS t)

where `DBW_set_req` ensures the proper conditions for the independence of the blocks in the DBW system. In Figure 5, we evaluate, using MATLAB, the reliability of the DBW system assuming exponential distributions for the system components with failure rates as given in the figure and a dormancy factor of 0.5.

In multi-processor systems, it is required to have an efficient communication method among system components, such as processors and memories. Multistage interconnection networks (MINs) can provide the necessary switching in multi-processor systems. A MIN consists of sources (inputs) and destinations (outputs) and is divided into a single-path MIN or a multiple-path MIN. In single-path MINs, there is only one possible path between each source and destination. Therefore, loosing any of the intermediate connections may lead to a failure. A SEN is an example of a single-path MIN. In order
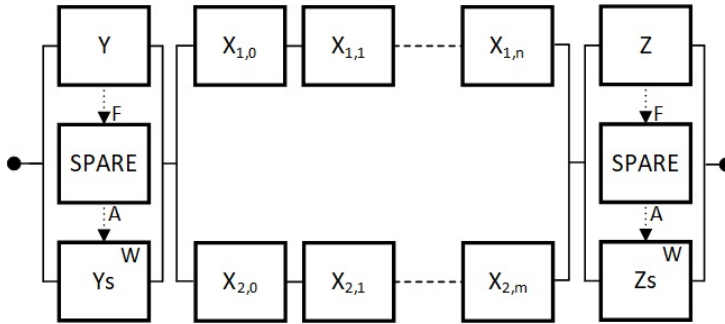

Figure 6: DRBD of Shuffle-exchange Network with Spare Constructs

21

to increase the reliability of such network, additional switching elements are added to the network which provide additional paths between each source and destination. A SEN having two paths between each source and destination is usually called SEN+. The terminal reliability analysis, which is the reliability of the connection between a given source and destination, is usually conducted using traditional RBDs [17]. Although the reliability of the system is increased going from SEN to SEN+, each source is always connected to a single switch and the same thing applies to the destination. The failure of these single switches would lead to the failure of the connection. Therefore, we propose to further enhance the reliability of this connection by using spare parts for these single switches so they can be replaced after failure. The DRBD of the modified SEN+ is shown in Figure 6, where Y and Z are the main single switches that are connected to the source and destination with their spares Ys and Zs, respectively. The parallel structure in the middle represents the reliability model of the two alternative paths between the source and the destination. Therefore, this DRBD consists of a series of two spare constructs and one parallel structure that consists of two series structures.

Using our DRBD operators, we formally express the structure function of this DRBD as:

$$
\begin{aligned}
\text{Q}_{\text{SEN}} = \text{nR\_AND } (\lambda\text{i}. \quad &\text{if i = 0 then R\_WSP Y Ys}_\text{a}\text{ Ys}_\text{d} \\
&\text{else if i = 1 then } \big((\text{nR\_AND X L1}) + (\text{nR\_AND X L2})\big) \quad (13) \\
&\text{else R\_WSP Z Zs}_\text{a}\text{ Zs}_\text{d}) \ \{0;\ 1;\ 2\}
\end{aligned}
$$

Thus, the outer series structure is expressed using the `nR_AND` operator over the set $\{0; 1; 2\}$ as this structure has three different structures; i.e., two spare constructs and one parallel structure. In order to re-utilize the verified expressions of reliability, it is required to express this DRBD using the series and parallel structures. Therefore, we verify that the DRBD event of the `Q`$_{\text{SEN}}$ is equal to a nested series-parallel-series structure as:

Theorem 4.1. $\vdash \forall\text{p X Y Ys}_\text{a}\text{ Ys}_\text{d}\text{ Z Zs}_\text{a}\text{ Zs}_\text{d}\text{ t L1 L2}.$
 `DISJOINT3` $\{0;\ 3\}$ `L1 L2` $\wedge$ `FINITE L1` $\wedge$ `FINITE L2` $\wedge$ `L1` $\neq \{\}$ $\wedge$ `L2` $\neq \{\}$ $\Rightarrow$
 (`DRBD_event` p `Q`$_{\text{SEN}}$ t =
  `DRBD_series` (λj.
    `DRBD_parallel` (λa.
      `DRBD_series` (λi.
        `event_set`
          [(`DRBD_event` p (`R_WSP` Y Ys$_\text{a}$ Ys$_\text{d}$) t,0);
           (`DRBD_event` p (`R_WSP` Z Zs$_\text{a}$ Zs$_\text{d}$) t,3)]
          (`rv_to_event` p X t) i)
      `ind_set` [$\{0\}$; L1; L2; $\{3\}$] a))
    (`ind_set` [$\{0\}$; $\{1;\ 2\}$; $\{3\}$] j)) $\{0;\ 1;\ 2\}$)

where `DISJOINT3` ensures that all sets are disjoint. Since `DRBD_series` accepts a group of indexed sets, we define a function `event_set` that accepts a list of pairs each of which is composed of a DRBD event with its index. This function also accepts the remaining

blocks of the DRBD that have their indices embedded in a set (that can be generic of any size), such as the parallel structure of the SEN. We also define `ind_set` that accepts a list of sets and returns a group of indexed sets. Since we are dealing with a series-parallel-series structure, we need three sets to identify the hierarchy of this nested structure. Set $\{0; 1; 2\}$ in Theorem 4.1 indicates that the outer series structure has three elements, i.e., three parallel structures. `ind_set [{0}; {1;2}; {3}]` indicates that the first parallel structure has only one series structure with index 0, the second parallel structure has two series structures with indices 1 and 2, and the third parallel structure has only one series structure with index 3. Finally, `ind_set [{0}; L1; L2; {3}]` implies that the first series structure has only one element with index 0, the second and third series structures have an arbitrary number of blocks indexed by $L1$ and $L2$. The last series structure has one element with index 3. We verify Theorem 4.1 using Theorem 3.10 and the equivalence of the event of the OR with the union of events besides some set-related theorems.

Based on Theorem 4.1, we verify a generic expression for the reliability of the SEN system:

Theorem 4.2. $\vdash \forall$p X Y Ys$_a$ Ys$_d$ Z Zs$_a$ Zs$_d$ t L1 L2.
 SEN_set_req p L1 L2 (ind_set [{0}; L1; L2; {3}])
   (ind_set [{0}; {1; 2}; {3}]) {0; 1; 2}
   (event_set [(DRBD_event p (R_WSP Y Ys$_a$ Ys$_d$) t,0);
               (DRBD_event p (R_WSP Z Zs$_a$ Zs$_d$) t,3)] (rv_to_event p X t)) $\Rightarrow$
 (prob p (DRBD_event p Q$_{SEN}$ t) =
  Rel p (R_WSP Y Ys$_a$ Ys$_d$) t * Rel p (R_WSP Z Zs$_a$ Zs$_d$) t *
  (1 - (1 - Normal ($\prod_{l \in L1}$ (real (Rel p (X l) t)))) *
      (1 - Normal ($\prod_{l \in L2}$ (real (Rel p (X l) t)))))))

where `SEN_set_req` ensures the required conditions of the input sets including that the sets are finite and nonempty. It also ensures the independence of the input events over the probability space and that they belong to the probability events. We first rewrite the goal using Theorem 4.1, then we use the reliability of the series-parallel-series to verify the final expression. The reliability of the spare constructs can be further rewritten using Theorem 3.6 given that the required conditions are ensured, such as the continuity of the CDFs. The final theorem with the expressions of the reliability of the spare constructs is available in [15]. Finally, we evaluate the reliability of the SEN system assuming the same failure rate of 1×10^{-5} for all switching elements. We also assume that each series structure has 16 switching elements. We evaluate the reliability for the SEN system without and with spare parts with a dormancy factor of 0.1, as shown in Figure 7. This result shows that considering the spares in the reliability analysis leads to having more reliable and realistic system than the traditional RBDs.

To sum up, we are able to provide a generic expression of reliability of the SEN+ system that is verified in HOL theorem proving, which cannot be obtained using any other formal method. In addition, through the verified reliability expressions of the SEN+ and DBW systems, we demonstrated that our formalization is flexible and can be used to model more complex systems of an arbitrary number of blocks by
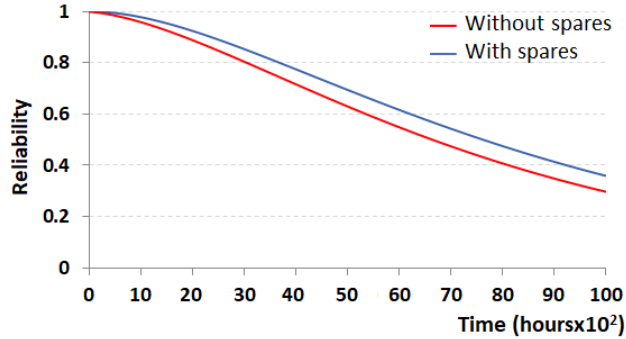
Figure 7: Reliability of SEN with/without spare constructs

implementing its hierarchy using sets that can be instantiated later to model a specific system structure, which is an added feature of our formalized algebra.

5 Conclusion

In this work, we proposed a new algebra to analyze dynamic reliability block diagrams (DRBDs). We developed the HOL formalization of this algebra in HOL4, which ensures its correctness and allows conducting the analysis within a theorem prover. Furthermore, this algebra provides formalized generic expressions of reliability that cannot be verified using other formal tools. This HOL formalization is the first of its kind that takes into account the system dynamics by providing the HOL formal model of spare constructs and temporal operators. The proposed algebra is compatible with the reliability expressions of traditional RBDs as demonstrated by the reliability expressions of the series and parallel structures. It also facilitates extending the verified reliability expressions to model complex systems using nested structures. Finally, we demonstrated the usefulness of this work by formally conducting the analysis of a drive-by-wire and a shuffle-exchange network systems to verify generic expressions of reliability, which are independent of the failure probability distribution of system components. We plan to extend this algebra to include other DRBD constructs, such as state dependencies, with their formalization in HOL, which would provide a more complete framework to algebraically analyze DRBDs within HOL theorem proving.

References

[1] E. Ruijters and M. Stoelinga. Fault Tree Analysis: A Survey of the State-of-the-art in Modeling, Analysis and Tools. *Computer Science Review*, 15-16:29 – 62, 2015.

[2] O. Hasan, W. Ahmed, S. Tahar, and M. S. Hamdi. Reliability Block Diagrams based Analysis: A Survey. In *Numerical Analysis and Applied Maths*, volume 1648, pages 850129.1–4, 2015.

[3] S. Distefano. *System Dependability and Performances: Techniques, Methodologies and Tools* . PhD thesis, University of Messina,, Italy, 2005.

[4] H. Xu and L. Xing. Formal Semantics and Verification of Dynamic Reliability Block Diagrams for System Reliability Modeling. In *Software Engineering and Applications*, pages 155–162, 2007.

[5] G. Smith. *The Object-Z Specification Language*, volume 1. Springer Science & Business Media, 2012.

[6] R. Robidoux, H. Xu, L. Xing, and M. Zhou. Automated Modeling of Dynamic Reliability Block Diagrams using Colored Petri Nets. *IEEE Transactions on Systems, Man and Cybernetics*, 40(2):337, 2010.

[7] W. Ahmed, O. Hasan, and S. Tahar. Formalization of Reliability Block Diagrams in Higher-order Logic. *Journal of Applied Logic*, 18:19–41, 2016.

[8] HOL4. https://hol-theorem-prover.org/, 2019.

[9] G. Merle. *Algebraic Modelling of Dynamic Fault Trees, Contribution to Qualitative and Quantitative Analysis*. PhD thesis, ENS, France, 2010.

[10] T. Mhamdi, O. Hasan, and S. Tahar. Formalization of Entropy Measures in HOL. In *Interactive Theorem Proving*, LNCS 6898, pages 233–248. Springer, 2011.

[11] M. Qasim, O. Hasan, M. Elleuch, and S. Tahar. Formalization of Normal Random Variables in HOL. In *Intelligent Computer Mathematics*, LNCS 9791, pages 44–59. Springer, 2016.

[12] T. Mhamdi. *Information-theoretic Analysis using Theorem Proving*. PhD thesis, Concordia University, Montreal, QC, Canada, 2012.

[13] Yassmeen Elderhalli, Waqar Ahmad, Osman Hasan, and Sofiène Tahar. Probabilistic Analysis of Dynamic Fault Trees using HOL Theorem Proving. *Journal of Applied Logics*, 2631(3):469, 2019.

[14] Isabelle. https://isabelle.in.tum.de/, 2019.

[15] Y. Elderhalli. DRBD Formal Analysis: HOL4 Script, http://hvg.ece.concordia.ca/code/hol/DRBD/index.php, 2019.

[16] A. Altby and D. Majdandzic. Design and Implementation of a Fault-tolerant Drive-by-wire System. Master's thesis, Chalmers University of Technology, Sweden, 2014.

[17] F. Bistouni and M. Jahanshahi. Analyzing the Reliability of Shuffle-exchange Networks using Reliability Block Diagrams. *Reliability Engineering & System Safety*, 132:97–106, 2014.