5-2018

Optimal Allocation of Resources in Reliability Growth

Mohammadhossein Heydari
University of Arkansas, Fayetteville

Optimal Allocation of Resources in Reliability Growth

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Engineering with a concentration in Industrial Engineering

by

Mohammadhossein Heydari
Isfahan University of Technology
Bachelor of Science in Industrial & Systems Engineering, 2012
Sharif University of Technology
Master of Science in Industrial Engineering, 2009

May 2018
University of Arkansas

This dissertation is approved for recommendation to the Graduate Council.

Kelly M. Sullivan, Ph.D.
Dissertation Director

Edward A. Pohl, Ph.D.
Committee Member

C. Richard Cassady, Ph.D.
Committee Member

Raymond R. Hill, Ph.D.
Committee Member

Abstract

Reliability growth testing seeks to identify and remove failure modes in order to improve system reliability. This dissertation centers around the resource allocation across the components of a multi-component system to maximize system reliability. We summarize this dissertation's contributions to optimal resource allocation in reliability growth.

Chapter 2 seeks to deploy limited testing resources across the components of a series-parallel system in effort to maximize system reliability under the assumption that each component's reliability exhibits growth according to an AMSAA model with known parameters. An optimization model for this problem is developed and then extended to consider the allocation of testing resources in a series-parallel system with consideration for the possibility of testing at different levels (system, subsystem, and component). We contribute a class of exact algorithms that decomposes the problem based upon the series-parallel structure. We prove the algorithm is finite, compare it with heuristic approaches on a set of test instances, and provide detailed analyses of numerical examples.

In Chapter 3, we extend model in Chapter 2 to solve a robust optimization version of this problem in which AMSAA parameters are uncertain but assumed to lie within a budget-restricted uncertainty set. We model the problem of robust allocation of testing resources to maximize system reliability for both series and series-parallel systems, and we develop and analyze exact solution approaches for this problem based on a cutting plane algorithm. Computational results demonstrate the value of the robust optimization approach as compared to deterministic alternatives.

In the last chapter, we develop a new model that merges testing components and installing redundancies within an integrated optimization model that maximizes system reliability. Specifically, our model considers a series-parallel system in which the system reliability can be improved by both testing components and installing redundant components. We contribute an exact algorithm that decomposes the problem into smaller integer linear programs. We prove that this algorithm is finite and apply it to a set of instances. Experiments demonstrate that the integrated approach gen-

erates greater reliabilities than applying test planning and redundancy allocation models iteratively, and moreover, it yields significant savings in computational time.

Acknowledgments

First and foremost I would like to express my special appreciation and thanks to my advisor Dr. Kelly Sullivan for his continuous support of my PhD study, motivation, and valuable guidance. This dissertation would not have been achievable without his guidance and support. I gratefully acknowledge the members of my Ph.D. committee Dr. Edward Pohl, Dr. Richard Cassady, and Dr. Raymond Hill for serving as my committee members and for their feedback and valuable contributions. Many thanks to Dr. Haitao Liao and Dr. Shengfan Zhang for their continuous help during my research. I would also like to thank to all my friends in Fayetteville. In particular, I am grateful to Ben and Nicole Shaddox, Vera Hembitskaya, John Miller, Mohsen Dadashi, Jingming Liu, Russ Reynerson, and Abdoul Maiga.

Last but not least, I would like to thank my family: my parents, Hosseinali and Manijeh, and my siblings, Marjan and Keyvan, for all their love and support.

Dedication

To my family

Contents

List of Figures

List of Tables

List of Papers

Chapter 2: M. Heydari, K. M. Sullivan, and E. A. Pohl. Algorithms for Resource Allocation Problems in Reliability Growth Testing. *European Journal of Operational Research*, submitted, 2018.

Chapter 3: M. Heydari, and K. M. Sullivan. Robust Allocation of Testing Resources in Reliability Growth. *Reliability Engineering & System Safety*, In Press, 2017.

Chapter 4: M. Heydari, and K. M. Sullivan. An Integrated Approach to Redundancy Allocation and Test Planning for Reliability Growth. *Computers & Operations Research*, 92: 182–193, 2018.

1 Introduction

Reliability growth is the improvement in system reliability through identifying and correcting failure modes of the design. This typically occurs during developmental testing, wherein the objective is to improve system reliability in a systematic way by identifying and mitigating the failure modes of the design before the system is introduced to market for usage. After the completion of developmental testing, design changes are much more costly; thus, an effective reliability growth program is highly valuable.

Duane [19] proposed one of the earliest reliability growth models based on the empirical observation that the cumulative failure rate (i.e., total number of failures divided by total testing time) has a linear relationship with total testing time when it is plotted on a log-log scale. Crow [16] later demonstrated that if cumulative failure rate versus total testing time truly has a linear relationship on log-log scale, then failures occur according to a nonhomogenous Poisson process with Weibull intensity. The resulting model became known as the Army Materiel Systems Analysis Activity (AMSAA) model, a reliability growth method used often when there is a continuous scale usage. In the AMSAA model, during testing, reliability failures are modeled according to a nonhomogeneous Poisson process with decreasing rate; therefore, system reliability increases with testing. Because the AMSAA model does not need explicit assumptions about the number and design of failure modes, it can be used in many different systems.

The reliability growth concept has been studied over the last 50 years. Although reviewing the entire history of reliability growth modeling is tangential to the goals of this dissertation, we provide a brief overview here. Reliability growth models have been developed for both hardware (see, e.g., [12, 16, 17, 19, 38, 40, 54, 59]) and software [25, 29, 45, 56] systems and specialized for the case of discrete [26, 39] or continuous data [16, 17, 19, 24, 25]. Discrete reliability growth models refer to growth models that use countable data such as the number of failure or pass/fail tests, while continuous reliability growth models refer to growth models that use continuous data, mainly time. One of the challenges associated with reliability growth modeling is that obtaining a

substantial amount of failure data may require significant testing, which is costly. To address this, models have been adapted to incorporate accelerated testing [31, 57] (i.e., applying stress to a system to induce quicker failures for reliability analysis) and Bayesian methods [37, 58], which allow for the incorporation of expert opinion. All of the models summarized thus far assume an underlying (parametric) process that may be difficult to verify statistically given the cost associated with obtaining data. Nonparametric models [47, 49] have been developed to address this shortcoming.

Reliability growth has been used extensively by the United States Department of Defense to assess and enhance the reliability of large-scale complex systems, and its use has also moved into the commercial sector for complex technical systems. As today's systems become more and more technically complex and the demands for increased reliability continue to increase, the need for and use of reliability growth techniques will continue to grow. The idea behind reliability growth planning and reliability growth testing is to fully test the operationally representative test article for a set of operationally representative missions. The goal is to determine if the designed system will meet operational and reliability requirements. The test durations and test profiles are based upon the design and mission characteristics for the system. Reliability is improved during reliability growth testing when system failures are identified during the test and design fixes are identified and implemented to reduce the likelihood of the associated failure mode of recurring again during system operation. Once fixes are identified, they are applied to systems that continue to undergo reliability growth testing. It is the identification of failures and the resulting design changes made to the system that mitigate or reduce the identified failure modes in the system that results in improved reliability of the systems that are fielded after the reliability growth testing program.

Reliability growth models have also been used within optimization to allocate testing resources efficiently. Coit [15] considers a testing time allocation problem for a system of serially connected components. Coit assumes that reliability growth of each component occurs according to the AMSAA model. Dai et al. [18] and Levitin [36] consider series-parallel problems that are nearly identical to a problem studied in Chapter 2, but all utilize only heuristic methods in solving the models.

In this dissertation, we consider the allocation of some limited testing resources across the components of a multi-component (i.e. series and series-parallel) system to maximize the system's reliability. We begin by relating test times (the model's decision variables) to reliability via the AMSAA model. Let $\lambda > 0$ and $\beta > 0$ respectively denote scale and shape parameters for a given component. After $\tau > 0$ units of cumulative test time across components of this design, failures occur in components at instantaneous rate

$$u(\tau;\lambda,\beta) = \lambda\beta\tau^{\beta-1}. \tag{1}$$

We assume $0 < \beta < 1$; thus, the failure rate decreases during the test (i.e., as a result of design changes), and the components becomes more reliable. After testing, no further design alterations are applied; therefore, failures occur in components according to a homogeneous Poisson process. Such a component's after-testing reliability for a mission of length $T > 0$ is given as

$$R(\tau;T,\lambda,\beta) = e^{-u(\tau;\lambda,\beta)T} = e^{-\lambda\beta\tau^{\beta-1}T}. \tag{2}$$

Equation (2) will be used to model the reliability of each component as a function of τ, its component's time on test.

The remainder of this dissertation is organized as follows. In Chapter 2, we consider the allocation of testing times in a series-parallel system when components follows AMSAA reliability growth model with known parameters. We also consider the allocation of test times in a series-parallel system with consideration for the possibility of testing at different levels (system, subsystem, and component). We propose a class of exact algorithms based on the series-parallel structure, whereas all of the previous papers utilize heuristics to solve this problem.

In Chapter 3, we extend the model in Chapter 2 to solve a robust version of the problem in which AMSAA parameters are uncertain but assumed to lie within a budget-restricted uncertainty set. We model the problem of robust allocation of testing resources to maximize system reliability for both series and series-parallel systems, and we develop and analyze exact solution approaches

for this problem based on cutting plane algorithm. A Monte Carlo simulation is used to compare our model with the situation where AMSAA parameters are known. We also study the performance of the exact solution methods for both series and series-parallel systems.

In Chapter 4, a new model is developed that merges two concepts of testing components (to identify and remove failure modes, resulting in reliability growth) and installing redundancies within an integrated optimization model that maximizes system reliability. Specifically, our model considers a series-parallel system in which the system reliability can be improved by both testing components and installing redundant components. We contribute a branch-and-bound algorithm that solves the problem optimally.

2 Algorithms for Resource Allocation Problems in Reliability Growth

2.1 Introduction

In this chapter, we consider the allocation of test times in a series-parallel system, and we assume the components exhibit reliability growth according to the AMSAA model. Moreover, we extend the resource allocation problem to consider subsystem- and system-level testing. We propose an exact algorithm for this problem that employs bounds derived from discrete subproblems. We demonstrate the efficacy of our algorithm on instances containing as many as five subsystems. Because it may be desirable in practice to solve instances involving tens of subsystems, we have also examined the application of heuristics to this problem.

Our work extends several related works in the literature. Coit [15] considers a testing time allocation problem for a system of serially connected components. As in this chapter, Coit assumes that reliability growth of each component occurs according to the AMSAA model. Coit's method is exact, but it draws from properties resulting from special structures not assumed here. Dai et al. [18] and Levitin [36] consider series-parallel problems that are similar to our component-level testing model but all of the aforementioned papers utilize only heuristic methods in solving the models. Awad [2] and Coit [15] extend the series system model of Coit [15] to the case where each component's failure rate is uncertain but assumed to follow a Normal distribution.

Given the literature summarized above, we now state the contributions of this paper. We (i) formally state the discretize-and-refine algorithm and prove its finite convergence for a basic version of the series-parallel test time allocation model and (ii) demonstrate how the algorithm's ideas can be exploited within the context of a resource-based decomposition scheme to solve the series-parallel allocation problem more efficiently. We then (iii) show that the algorithm can be extended to solve a resource allocation problem in which we incorporate the possibility of testing at the system-, subsystem-, or component-level. We (iv) demonstrate insights upon solving numerical examples from each of the models, and (v) provide a computational study to demonstrate the algorithm's empirical performance.

The remainder of this chapter is organized as follows. In Section 2.2, an optimization model of resource allocation problem in reliability growth is presented. Section 2.3 develops an exact solution method, and we present the multi-level resource allocation problem in Section 2.4. Section 2.5 summarizes computational results obtained from implementing our algorithm and several heuristics on a set of test instances, and Section 2.6 concludes.

2.2 Optimization Model for Component-Level Testing

We now propose a mathematical model which maximizes the system reliability. A summary of the model's notation follows

Parameters

N	Number of subsystems (indexed by ℓ)
M_ℓ	Number of designs in subsystem $\ell = 1,\ldots,N$ (indexed by i)
$V_{\ell,i}$	Number of components in design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$
T	Mission length
$\lambda_{\ell,i}, \beta_{\ell,i}$	AMSAA model parameters for design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$
b	Available budget
$c_{\ell,i}$	Cost per unit time of testing design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$
$\tau^0_{\ell,i}$	Initial testing time of design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$ (assumed to be positive)

Decision Variables

$\tau_{\ell,i}$	Testing time of design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$
$R_{\ell,i}(\tau_{\ell,i}, T)$	Reliability of design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$

We consider a system consisting of serially connected *subsystems* $\ell = 1,\ldots,N$. Subsystem ℓ has a parallel structure and contains components associated with *designs* $(\ell, i),\ i = 1,\ldots,M_\ell$. Let $V_{\ell,i}$

denote the number of *components* in subsystem $\ell = 1,\ldots,N$ that are associated with design (ℓ,i). Our model assumes the following:

Assumption 1. All components of all designs are independent.

Assumption 2. Components in each subsystem are connected in an active redundancy.

Assumption 3. The redundancy level for each design is fixed.

Assumption 4. The components of each design exhibit reliability growth according to the AMSAA reliability growth model with known parameters $\lambda_{\ell,i}$ and $\beta_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, and these parameters never change.

Assumption 5. A fixed mission length T is considered for the purposes of evaluating reliability.

Defining $\tau_{\ell,i}$ as the testing time allocated to design (ℓ,i), the mission reliability for every component of design (ℓ,i) is $R(\tau_{\ell,i};T,\lambda_{\ell,i},\beta_{\ell,i})$. For simplicity of exposition, we use the shorthand $R_{\ell,i}(\tau_{\ell,i},T) \equiv R(\tau_{\ell,i};T,\lambda_{\ell,i},\beta_{\ell,i})$ throughout the remainder of this document to represent the reliability of design (ℓ,i). We consider the problem of selecting each design's time on test, subject to a limitation on the total (system-wide) testing time, in order to maximize the system's reliability. Assuming all components are independent, the testing time allocation problem defined by

$$R^* = \max \prod_{\ell=1}^{N} \left[1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}(\tau_{\ell,i},T)\right)^{V_{\ell,i}} \right], \tag{3a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} \sum_{i=1}^{M_\ell} c_{\ell,i}\tau_{\ell,i} \le b, \tag{3b}$$

$$\tau_{\ell,i} \ge \tau^0_{\ell,i}, \qquad \forall \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell. \tag{3c}$$

Objective (3a) maximizes system reliability. Constraint (3b) limits the budget and Constraints (3c) ensure the testing time for design (ℓ,i) is at least the initial testing time $\tau^0_{\ell,i}$.

The case of our model corresponding to only one design in each subsystem (i.e., $M_\ell = 1,\ \ell = 1,\ldots,N$) is an interesting special case, which is perhaps the most relevant to practice. We have dedicated attention to this important special case in the following analysis; however, we have not

prohibited the possibility that $M_\ell > 1$ in our models. This case is important, for instance, as it allows for incorporating redundancy and component-selection decisions (see Chapter 4).

We describe algorithms for Model (3) in the following section. In doing so, it will be helpful to define $R(\tau, T) \equiv \prod_{\ell=1}^{N} \left[1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}(\tau_{\ell,i}, T)\right)\right]$ as the system reliability given test times $\tau_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$.

2.3 Optimal Algorithm for Component-Level Testing

In this section, we develop methodology for solving Model (3) within an arbitrarily small optimality gap. If $M_\ell = V_{\ell,1} = 1,\ \ell = 1,\ldots,N$, Model (3) becomes a convex program via applying a logarithm to its objective (resulting in the Coit [15] model) and can therefore be solved to optimality using derivative-based methods. Unfortunately, when $M_\ell > 1$ for some ℓ, no convex formulation of Model (3) is apparent.

Given the discussion above, we proceeded by developing (in this section) a new algorithm for which we could prove global optimality. This algorithm exploits the monotonicity of Objective (3a) in deriving upper and lower bounds for the optimal objective value. As we will demonstrate in Section 2.5, this algorithm is consistently able to prove ε-optimality for 25-variable instances (at $\varepsilon \approx 10^{-4}$) within one hour. However, the algorithm requires solving a number of integer programs, which can require a significant amount of computational effort for large instances. As a result, we have also implemented a simulated annealing heuristic (described in the following section), which provides no guarantee of optimality but seems to be effective at identifying near-optimal solutions quickly. (Of course, we can only comment that the solutions are near-optimal because we have developed the global optimal algorithm for comparison.)

We now develop the ε-optimal algorithm, which utilizes sequences of discrete approximations to Model (3) to develop arbitrarily tight upper and lower bounds on the optimal objective to Model (3) in a finite number of iterations.

We begin by stating a discrete version of Model (3). The notation used in this model is as follows

Parameters

P	Number of discrete points in each design's discretized reliability function (indexed by p, assumed without loss of generality to be equal for all designs)
$\tau_{\ell,i}^{p}$	p-th possible testing time associated with design i of subsystem ℓ ($p = 1,\ldots,P$; $\ell = 1,\ldots,N$; $i = 1,\ldots,M_\ell$)
$r_{\ell,i}^{p}$	Reliability of point p in design i of subsystem ℓ ($p = 1,\ldots,P$; $\ell = 1,\ldots,N$; $i = 1,\ldots,M_\ell$)

Decision Variables

$$x_{\ell,i}^{p} \quad \begin{cases} 1 & \text{if the testing time for design } i \text{ in subsystem } \ell \text{ is } \tau_{\ell,i}^{p} \\ 0 & \text{otherwise} \end{cases}$$

$(p = 1,\ldots,P;\ \ell = 1,\ldots,N;\ i = 1,\ldots,M_\ell)$

Using the above notation, the discretized model is given as

$$R^{\mathrm{D}}(\mathbf{r}) = \max \prod_{\ell=1}^{N}\left[1 - \prod_{i=1}^{M_\ell}\prod_{p=1}^{P}\left(1 - r_{\ell,i}^{p}\right)^{V_{\ell,i}x_{\ell,i}^{p}}\right], \tag{4a}$$

$$\text{s.t.} \sum_{\ell=1}^{N}\sum_{i=1}^{M_\ell}\sum_{p=1}^{P} c_{\ell,i}\tau_{\ell,i}^{p}x_{\ell,i}^{p} \le b, \tag{4b}$$

$$\sum_{p=1}^{P} x_{\ell,i}^{p} = 1, \qquad \forall \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell, \tag{4c}$$

where $\mathbf{r}$ refers to the vector of $r_{\ell,i}^{p}$-values and $r_{\ell,i}^{p}$ is the reliability associated with testing time $\tau_{\ell,i}^{p}$. Initially, we consider Model (4) with $r_{\ell,i}^{p} = R(\tau_{\ell,i}^{p}, T)$, which yields a lower bound on R^*; however, we later show that another definition of $r_{\ell,i}^{p}$ yields an upper bound on R^*. We assume

$$\tau_{\ell,i}^{1} < \tau_{\ell,i}^{2} < \cdots < \tau_{\ell,i}^{P},\ \forall \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell. \tag{5}$$

Sections 2.3.1–2.3.2 provide a formal definition of our solution method, which draw upon solving

instances of Model (4). Because the method for solving Model (4) may be customized depending on properties of the algorithm, we defer discussion of solution methodologies for Model (4) until after the algorithm has been defined.

2.3.1 Discretize-and-Refine Method

In this section, we formally define the discretize-and-refine (D&R) methodology. This methodology hinges upon the following observation: By solving Model (4), in which test times are restricted to a finite set of possible values, it is possible to obtain both upper and lower bounds on R^*, the optimal system reliability under continuous test times. By successive refinement of this discretization, the bounds converge within ε in finite iterations.

We now establish bounds on R^* that can be obtained by solving Model (4). Setting $r^p_{\ell,i} = R_{\ell,i}(\tau^p_{\ell,i}, T)$ in Model (4) yields a lower bound while $r^p_{\ell,i} = R_{\ell,i}(\tau^{p+1}_{\ell,i}, T)$ yields an upper bound. (Let $\tau^{P+1}_{\ell,i} \equiv \tau^{\max}_{\ell,i}$, where $\tau^{\max}_{\ell,i}$ is the maximum possible testing time for design (ℓ, i), which is either given or set equal to $(b + c_{\ell,i}\tau^0_{\ell,i} - \sum_{\ell'=1}^{N}\sum_{i'=1}^{M_\ell} c_{\ell',i'}\tau^0_{\ell',i'})/c_{\ell,i}$). The upper bound model is referred to as UB-MILP, and the lower bound as LB-MILP. Figure 1 diagrams the bounds for a particular design (ℓ, i). Figure 1(a) depicts an upper bound obtained associating reliability $R_{\ell,i}(\tau^{p+1}_{\ell,i}, T)$ with test time $\tau^p_{\ell,i}$, and Figure 1(b) demonstrates a lower bound defined by associating reliability $R_{\ell,i}(\tau^p_{\ell,i}, T)$ with test time $\tau^p_{\ell,i}$. We now establish that LB-MILP and UB-MILP respectively provide lower and upper bounds on R^*.

Theorem 1. Define $\bar{\mathbf{r}}$ by

$$\bar{r}^p_{\ell,i} = R_{\ell,i}(\tau^p_{\ell,i}, T),\ \forall p = 1, \ldots, P. \tag{6}$$

Then $R^{\mathrm{D}}(\bar{\mathbf{r}}) \leq R^*$. (That is, LB-MILP provides a lower bound for R^*.)

Proof. By setting $\bar{r}^p_{\ell,i} = R_{\ell,i}(\tau^p_{\ell,i}, T)$, any feasible solution $\hat{x}$ in Model (4) corresponds to a solution $\tau_{\ell,i} = \sum_{p=1}^{P} \tau^p_{\ell,i}\hat{x}^p_{\ell,i}$ in Model (3) with the same objective value. □

Theorem 2. Define $\bar{\bar{\mathbf{r}}}$ by

$$\bar{\bar{r}}^p_{\ell,i} = R_{\ell,i}(\tau^{p+1}_{\ell,i}, T),\ \forall p = 1, \ldots, P. \tag{7}$$

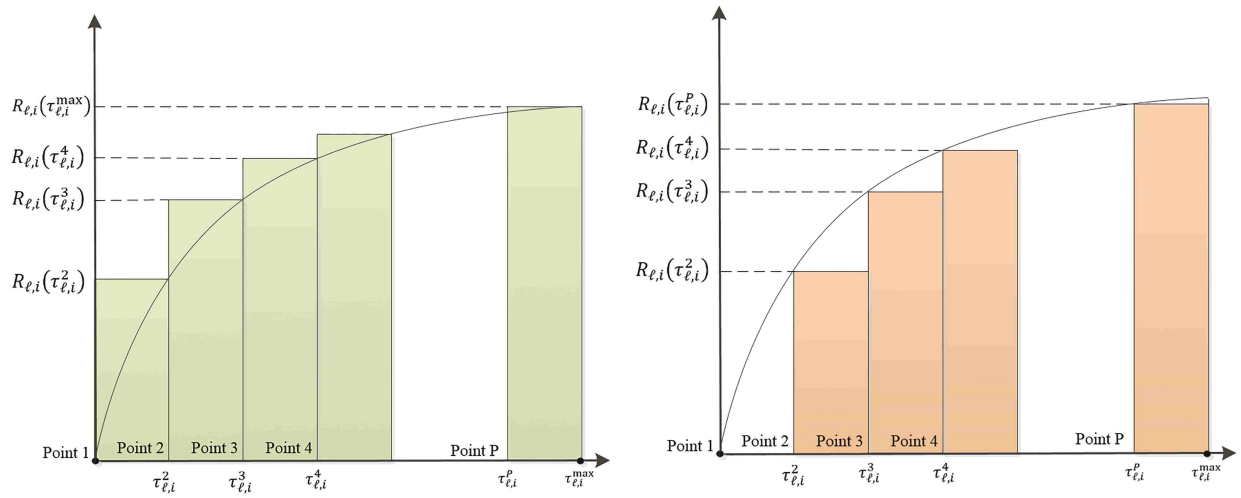


Figure 1: Upper bound and lower bound model for a particular component

Then $R^{\mathrm{D}}(\bar{\mathbf{r}}) \geq R^*$. (That is, UB-MILP provides an upper bound for R^*.)

Proof. Define $\tau^*_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, as an optimal solution for Model (3) with objective value R^*. For each design (ℓ,i), find the index point $\hat{P}_{\ell,i} \in \{1,\ldots,P\}$ such that $\tau^{\hat{P}_{\ell,i}}_{\ell,i} \leq \tau^*_{\ell,i} < \tau^{\hat{P}_{\ell,i}+1}_{\ell,i}$ and let $\hat{\tau}_{\ell,i} = \tau^{\hat{P}_{\ell,i}+1}_{\ell,i}$. The solution with $x^{\hat{P}_{\ell,i}}_{\ell,i} = 1,\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, is feasible to UB-MILP, and the objective value of this solution in UB-MILP is $R(\hat{\tau},T)$. However, this value is no less than R^* because $R(\tau,T)$ is nondecreasing in τ. Hence, $R^* \leq R(\hat{\tau},T) \leq R^{\mathrm{D}}(\bar{\mathbf{r}})$. □

A formal statement of the D&R algorithm, which utilizes the bounds established in the preceding theorems, follows. The number of discrete test times maintained for each design now varies throughout the algorithm; however, we still utilize notation P to refer to the (now variable) number of discrete points associated with each design. Likewise, $\tau^{P+1}_{\ell,i}$ again refers to $\tau^{\max}_{\ell,i}$.

Step 0: For each design, define P_{int} as the initial number of discrete points, and $\tau^{\min}_{\ell,i}$ and $\tau^{\max}_{\ell,i}$ as the minimum and maximum possible testing times for design (ℓ,i), respectively. Set $\tau^{\min}_{\ell,i} = \tau^0_{\ell,i}$ and $\tau^{\max}_{\ell,i} = (b + c_{\ell,i}\tau^0_{\ell,i} - \sum_{\ell'=1}^{N}\sum_{i'=1}^{M_\ell} c_{\ell',i'}\tau^0_{\ell',i'})/c_{\ell,i}$. Compute

$$\tau^p_{\ell,i} = \tau^{\min}_{\ell,i} + \frac{p-1}{P_{\mathrm{int}}}(\tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i}),\ \forall \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell,\ p = 1,\ldots,P_{\mathrm{int}}. \qquad (8)$$

Let UB and LB denote the best known upper and lower bounds on the optimal system reliability, and let τ^* denote the incumbent solution. Let $\varepsilon > 0$ denote the optimality tolerance. Set $P \leftarrow P_{\text{int}}$, $UB \leftarrow 1$, and $LB \leftarrow 0$.

Step 1: If $UB - LB \leq \varepsilon$, stop (τ^* is ε-optimal); otherwise, go to Step 2.

Step 2: Solve LB-MILP with objective $R^{\text{D}}(\bar{\mathbf{r}})$ and solution $\hat{\tau}$. If $R^{\text{D}}(\bar{\mathbf{r}}) > LB$, set $LB \leftarrow R^{\text{D}}(\bar{\mathbf{r}})$ and $\tau^* \leftarrow \hat{\tau}$.

Step 3: Solve UB-MILP and let $P^*_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, denote an optimal solution with objective $R^{\text{D}}(\bar{\bar{\mathbf{r}}})$. Define a new candidate test time $\bar{\tau}_{\ell,i} = 0.5\left(\tau_{\ell,i}^{P^*_{\ell,i}} + \tau_{\ell,i}^{P^*_{\ell,i}+1}\right)$ for each design (ℓ,i). Set $P \leftarrow P+1$ and define $\tau^P_{\ell,i} = \bar{\tau}_{\ell,i}$. Renumber each design's candidate test times in accordance with (5).

Step 4: If $R^{\text{D}}(\bar{\bar{\mathbf{r}}}) < UB$, set $UB \leftarrow R^{\text{D}}(\bar{\bar{\mathbf{r}}})$.

Step 5: Return to Step 1.

We now establish convergence of D&R via Lemmas 1–6 and Theorem 3. As will become apparent through the convergence proofs, the algorithm is quite general. If we impose additional constraints on τ in Model (3), the same discretization approach can be used to generate upper- and lower-bounds on R^*, and D&R therefore remains valid. Similarly, the approach can be adapted to other (i.e., non-series-parallel) system structures provided the reliability function is nondecreasing in the component design test times.

Lemma 1. Let design reliabilities $\bar{\rho}_{\ell,i}$ and $\bar{\bar{\rho}}_{\ell,i}$ be given such that

$$0 \leq \bar{\rho}_{\ell,i} \leq \bar{\bar{\rho}}_{\ell,i} \leq 1,\ \forall \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell. \tag{9}$$

Define $\varepsilon_{\ell,i} = \bar{\bar{\rho}}_{\ell,i} - \bar{\rho}_{\ell,i}$, $\varepsilon_{\max} \equiv \max_{\ell=1,\ldots,N,\ i=1,\ldots,M_\ell}\{\varepsilon_{\ell,i}\}$, $M_{\max} \equiv \max_{\ell=1,\ldots,N}\{M_\ell\}$, and $\varepsilon \equiv$

$NM_{\max}\varepsilon_{\max}/\left[1-(N+1)M_{\max}\varepsilon_{\max}\right]$. If $M_{\max}\varepsilon_{\max}\leq 1$ and $NM_{\max}\varepsilon_{\max}/\left(1-M_{\max}\varepsilon_{\max}\right)\leq 1$, then

$$\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\bar{\rho}}_{\ell,i}\right)\right]-\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\rho}_{\ell,i}\right)\right]\leq\varepsilon. \tag{10}$$

(That is, when $V_{\ell,i}=1,\ \ell=1,\ldots,N,\ i=1,\ldots,M_\ell$, the system reliabilities resulting from design reliabilities $\bar{\rho}_{\ell,i}$ and $\bar{\bar{\rho}}_{\ell,i}$ differ by at most ε.)

Proof. Let $\varepsilon_\ell=\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\bar{\rho}}_{\ell,i}\right)\right]-\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\rho}_{\ell,i}\right)\right],\ \ell=1,\ldots,N$. Then

$$1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i})+\varepsilon_\ell=1-\prod_{i=1}^{M_\ell}(1-\bar{\bar{\rho}}_{\ell,i}), \tag{11a}$$

$$=1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i}-\varepsilon_{\ell,i}), \tag{11b}$$

$$=1+\sum_{I\subseteq\mathscr{P}(M_\ell)}(-1)^{|I|+1}\left(\prod_{i\in I}\varepsilon_{\ell,i}\right)\left[\prod_{i\in\{1,\ldots,M_\ell\}\setminus I}\left(1-\bar{\rho}_{\ell,i}\right)\right], \tag{11c}$$

where $\mathscr{P}(M_\ell)$ denotes the powerset of $\{1,\ldots,M_\ell\}$ and Equation (11c) follows from the expansion of the product in Equation (11b). However, because $0\leq 1-\bar{\rho}_{\ell,i}\leq 1$ and $\varepsilon_{\ell,i}\geq 0$ for all $i=1,\ldots,M_\ell$, it follows that

$$1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i})+\varepsilon_\ell\leq 1-\prod_{i=1}^{M_\ell}\left(1-\bar{\rho}_{\ell,i}\right)+\sum_{I\subseteq\mathscr{P}(M_\ell):I\neq\emptyset}\left(\prod_{i\in I}\varepsilon_{\ell,i}\right), \tag{12}$$

which establishes

$$\varepsilon_\ell\leq\sum_{I\subseteq\mathscr{P}(M_\ell):I\neq\emptyset}\left(\prod_{i\in I}\varepsilon_{\ell,i}\right)\leq\sum_{k=1}^{M_\ell}\binom{M_\ell}{k}\left(\varepsilon_{\max}\right)^k, \tag{13}$$

where the second inequality in (13) follows due to replacing $\varepsilon_{\ell,i}$ with the upper bound $\varepsilon_{\max}$ and combining terms in the summation for which $\varepsilon_{\max}$ has the same exponent. Therefore,

$$\varepsilon_\ell\leq\sum_{k=1}^{M_\ell}M_\ell^k\varepsilon_{\max}^k\leq\sum_{k=1}^{M_\ell}M_{\max}^k\varepsilon_{\max}^k, \tag{14a}$$

$$\leq\sum_{k=1}^{\infty}\left(M_{\max}\varepsilon_{\max}\right)^k=\frac{M_{\max}\varepsilon_{\max}}{1-M_{\max}\varepsilon_{\max}}\equiv\delta_{\max}. \tag{14b}$$

In the first inequality of Equation (14a), $\binom{M_\ell}{k}$ is replaced by the upper bound M_ℓ^k. Because $M_\ell \leq M_{\max}$, $M_{\max}$ is used instead of M_ℓ in the second inequality in (14a). Letting

$$\varepsilon_s \equiv \prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}(1-\bar{\bar{\rho}}_{\ell,i})\right]-\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i})\right], \tag{15}$$

we have

$$\begin{aligned}
\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i})\right]+\varepsilon_s &= \prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}(1-\bar{\bar{\rho}}_{\ell,i})\right], && (16a)\\
&= \prod_{\ell=1}^{N}\left[\left(1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i})\right)+\varepsilon_\ell\right], && (16b)\\
&= \sum_{L\subseteq\mathscr{P}(N)}\left(\prod_{\ell\in L}\varepsilon_\ell\right)\left[\prod_{\ell\in\{1,\ldots,N\}\setminus L}\left(1-\prod_{i=1}^{M_\ell}\left(1-\bar{\rho}_{\ell,i}\right)\right)\right], && (16c)\\
&\leq \prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}(1-\bar{\rho}_{\ell,i})\right]+\sum_{L\subseteq\mathscr{P}(N):L\neq\emptyset}\left(\prod_{\ell\in L}\varepsilon_\ell\right), && (16d)
\end{aligned}$$

where $\mathscr{P}(N)$ is the powerset of $\{1,\ldots,N\}$. Since $0\leq 1-\bar{\rho}_{\ell,i}\leq 1$, $1-\bar{\rho}_{\ell,i}$ is substituted with 1 in Equation (16d). Because $\varepsilon_\ell \leq \delta_{\max}$, it follows that

$$\begin{aligned}
\varepsilon_s &\leq \sum_{L\subseteq\mathscr{P}(N):L\neq\emptyset}\left(\prod_{\ell\in L}\delta_{\max}\right)=\sum_{\ell=1}^{N}\binom{N}{\ell}\delta_{\max}^{\ell}, && (17a)\\
&\leq \sum_{\ell=1}^{N}N^\ell\delta_{\max}^{\ell}\leq\sum_{\ell=1}^{\infty}\left(N\delta_{\max}\right)^\ell, && (17b)\\
&= \frac{N\delta_{\max}}{1-N\delta_{\max}}=\frac{NM_{\max}\varepsilon_{\max}}{1-(N+1)M_{\max}\varepsilon_{\max}}, && (17c)
\end{aligned}$$

where the first inequality of (17b) follows because $\binom{N}{\ell}\leq N^\ell$. This completes the proof. □

Lemma 2. Let $\bar{\rho}_{\ell,i}$, $\bar{\bar{\rho}}_{\ell,i}$, $\varepsilon_{\ell,i}$, $\varepsilon_{\max}$, $M_{\max}$, and ε be defined as in Lemma 1 with the exception that $M_{\max}\equiv\max_{\ell=1,\ldots,N}\left\{\sum_{i=1}^{M_\ell}V_{\ell,i}\right\}$. Then

$$\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\bar{\rho}}_{\ell,i}\right)^{V_{\ell,i}}\right]-\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\rho}_{\ell,i}\right)^{V_{\ell,i}}\right]\leq\varepsilon. \tag{18}$$

(This extends Lemma 1 to give conditions under which the system reliabilities under $\bar{\rho}_{\ell,i}$ and $\bar{\bar{\rho}}_{\ell,i}$ differ by no more than ε when $V_{\ell,i} > 1$ for some (ℓ,i).)

Proof. The result follows upon creating replicates $(\ell,i,v),\ v = 1,\ldots,V_{\ell,i}$, of each design (ℓ,i) where $\bar{\rho}_{\ell,i,v} \equiv \bar{\rho}_{\ell,i}$ and $\bar{\bar{\rho}}_{\ell,i,v} \equiv \bar{\bar{\rho}}_{\ell,i}$. Under these definitions, Inequality (18) can be restated as

$$\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\prod_{v=1}^{V_{\ell,i}}\left(1-\bar{\bar{\rho}}_{\ell,i,v}\right)\right]-\prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\prod_{v=1}^{V_{\ell,i}}\left(1-\bar{\rho}_{\ell,i,v}\right)\right]\leq\varepsilon; \tag{19}$$

however, this follows directly from Lemma 1 as applied to the replicated designs (ℓ,i,v). Note that the statement of Lemma 1 now requires $M_{\max}$ to equal the maximum number of replicated designs (given by $\sum_{i=1}^{M_\ell} V_{\ell,i}$) in any subsystem ℓ. □

Lemma 3. Let $\bar{x}$ represent a binary optimal solution to UB-MILP (i.e., using $\bar{\bar{\mathbf{r}}}$ as defined in the statement of Theorem 2) with respect to candidate test times $\tau^p_{\ell,i},\ p = 1,\ldots,P$, and let $P^*_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, denote the index $p \in \{1,\ldots,P\}$ such that $\bar{x}^p_{\ell,i} = 1$. Define $\bar{\tau}_{\ell,i} \equiv \tau^{P^*_{\ell,i}}_{\ell,i}$ and $\bar{\bar{\tau}}_{\ell,i} \equiv \tau^{P^*_{\ell,i}+1}_{\ell,i}$, and let $\varepsilon_{\ell,i} \equiv R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}) - R_{\ell,i}(\bar{\tau}_{\ell,i})$. With respect to $\varepsilon_{\ell,i}$, define $\varepsilon_{\max}$ as in Lemma 1 (also Lemma 2) and define $M_{\max}$ as in Lemma 2. Let $\varepsilon \equiv NM_{\max}\varepsilon_{\max}/\left[1-(N+1)M_{\max}\varepsilon_{\max}\right]$. If $M_{\max}\varepsilon_{\max} \leq 1$ and $NM_{\max}\varepsilon_{\max}/\left(1-M_{\max}\varepsilon_{\max}\right) \leq 1$, then $R^{\mathrm{D}}(\bar{\bar{\mathbf{r}}}) - R^* \leq \varepsilon$.

Proof. Defining $\bar{\rho}_{\ell,i} = R_{\ell,i}(\bar{\tau}_{\ell,i})$ and $\bar{\bar{\rho}}_{\ell,i} = R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i})$, we have that

$$R^{\mathrm{D}}(\bar{\bar{\mathbf{r}}}) = \prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\bar{\rho}}_{\ell,i}\right)^{V_{\ell,i}}\right], \tag{20}$$

and, because $\bar{\tau}_{\ell,i}$ is feasible to Model (3),

$$R^* \geq \prod_{\ell=1}^{N}\left[1-\prod_{i=1}^{M_\ell}\left(1-\bar{\rho}_{\ell,i}\right)^{V_{\ell,i}}\right]. \tag{21}$$

Hence, from Lemma 2, we have that $R^{\mathrm{D}}(\bar{\bar{\mathbf{r}}}) - R^* \leq \varepsilon$. □

Lemma 4. For each $\varepsilon > 0$, there exists $\delta_{\ell,i} > 0$ such that $R_{\ell,i}(\tau_{\ell,i} + \delta_{\ell,i}, T) - R_{\ell,i}(\tau_{\ell,i}, T) \leq \varepsilon$ for $\tau^0_{\ell,i} \leq \tau_{\ell,i} \leq b/c_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$. (That is, $R_{\ell,i}(\tau_{\ell,i}, T)$ is Lipschitz continuous on

$\tau^0_{\ell,i} \leq \tau_{\ell,i} \leq b/c_{\ell,i}$.)

Proof. Differentiating the reliability function in Equation (2) with respect to τ, we see that

$$\frac{dR_{\ell,i}(\tau_{\ell,i},T)}{d\tau_{\ell,i}} = R_{\ell,i}(\tau_{\ell,i},T)\left[-\lambda\beta(\beta-1)\tau_{\ell,i}^{\beta-2}T\right], \tag{22a}$$

$$\leq R_{\ell,i}(b/c_{\ell,i},T)\left[-\lambda\beta(\beta-1)\tau_{\ell,i}^{0\ \beta-2}T\right], \tag{22b}$$

where (22b) holds because, $R_{\ell,i}(\tau_{\ell,i},T) \leq R_{\ell,i}(b/c_{\ell,i},T)$, $\lambda > 0$, and $0 < \beta < 1$. The right-hand side of (22b) provides the Lipschitz constant c such that $\delta_{\ell,i} = \varepsilon/c$. □

Lemma 5. Given $\delta > 0$, define $S = \lfloor 2b/(\delta c_{\ell,i})\rfloor$. Let $P^k_{\ell,i},\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, denote the index $p \in \{1,\ldots,P\}$ such that $\bar{x}^p_{\ell,i} = 1$ in UB-MILP's optimal solution in the k-th iteration D&R. Define $\bar{\tau}^k_{\ell,i} \equiv \tau^{P^k_{\ell,i}}_{\ell,i}$ and $\bar{\bar{\tau}}^k_{\ell,i} \equiv \tau^{P^k_{\ell,i}+1}_{\ell,i}$. Then $\Delta^k_{\ell,i} \equiv \bar{\bar{\tau}}^k_{\ell,i} - \bar{\tau}^k_{\ell,i} > \delta$ for at most S iterations k.

Proof. We prove the result initially for $P_{\text{int}} = 1$, and then extend this proof for $P_{\text{int}} > 1$. We establish the $P_{\text{int}} = 1$ case via induction on the following claim: $\Delta^k_{\ell,i} \equiv \bar{\bar{\tau}}^k_{\ell,i} - \bar{\tau}^k_{\ell,i} > \delta$ for no more than $2^n - 1$ iterations k provided that $2^{n-1}\delta \leq \tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i} < 2^n\delta$. For ease of exposition we define $N_{\ell,i}$ as the maximum number of iterations k for which $\Delta^k_{\ell,i} > \delta$.

In the base case, $n = 0$, in each iteration $\Delta^k_{\ell,i} \leq \tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i} \leq \delta$ so $N_{\ell,i} = 0 \leq 2^0 - 1$. Now, assume the claim holds for $n = n'$; therefore, if $2^{n'-1}\delta \leq \tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i} < 2^{n'}\delta$, then $N_{\ell,i} \leq 2^{n'} - 1$. When $n = n' + 1$, because $P_{\text{int}} = 1$, the first iteration divides the interval $[\tau^{\min}_{\ell,i}, \tau^{\max}_{\ell,i}]$ into two equal intervals with lengths less than or equal to $2^{n'}\delta$. As a result, when $n = n' + 1$, the value $N_{\ell,i}$ is less than or equal to $1 + 2(2^{n'} - 1) = 2^{n'+1} - 1$, which completes the proof of the claim. We can easily conclude that when $n - 1 = \lfloor \log_2(\tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i})/\delta \rfloor$, then $N_{\ell,i} \leq 2^n - 1$, hence

$$N_{\ell,i} \leq 2^n - 1 = 2^{\left\lfloor \log_2 \frac{\tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i}}{\delta} \right\rfloor + 1} - 1 \leq \frac{2b}{c_{\ell,i}\delta} - 1 \leq \left\lfloor \frac{2b}{c_{\ell,i}\delta} \right\rfloor, \tag{23}$$

where in the second inequality of Equation (23), the upper bound $b/c_{\ell,i}$ is used instead of $\tau^{\max}_{\ell,i} - \tau^{\min}_{\ell,i}$.

Now, if $P_{\text{int}} > 1$, there are initially P_{int} disjoint intervals with lengths less than or equal to $(\tau_{\ell,i}^{\max} - \tau_{\ell,i}^{\min})/P_{\text{int}}$. Therefore,

$$N_{\ell,i} \leq P_{\text{int}} \left\lfloor \frac{2b}{c_{\ell,i}\delta P_{\text{int}}} \right\rfloor \leq \left\lfloor \frac{2b}{c_{\ell,i}\delta} \right\rfloor. \tag{24}$$

This completes the proof. □

Lemma 6. Let $\bar{\tau}_{\ell,i}^k$ and $\bar{\bar{\tau}}_{\ell,i}^k$ be defined as in Lemma 5. For each $\varepsilon > 0$, the number of iterations for which some design (ℓ, i) satisfies $R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}^k, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}^k, T) > \varepsilon$ is bounded.

Proof. By Lemma 4, there exists $\delta_{\ell,i} > 0$ such that $R_{\ell,i}(\tau_{\ell,i} + \delta_{\ell,i}, T) - R_{\ell,i}(\tau_{\ell,i}, T) \leq \varepsilon$, for $\tau_{\ell,i}^0 \leq \tau_{\ell,i} \leq b/c_{\ell,i}$. Define $\delta = \min_{\ell=1,\ldots,N, i=1,\ldots,M_\ell}\{\delta_{\ell,i}\}$, $S_{\ell,i} = \lfloor 2b/(c_{\ell,i}\delta) \rfloor$, and $S \equiv \max_{\ell=1,\ldots,N, i=1,\ldots,M_\ell} \{S_{\ell,i}\}$. By Lemma 5, the number of iterations k for which design (ℓ, i) satisfies $\bar{\bar{\tau}}_{\ell,i}^k - \bar{\tau}_{\ell,i}^k > \delta$ is at most $S_{\ell,i}$; hence, the number of iterations k for which design (ℓ, i) satisfies $R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}^k, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}^k, T) > \varepsilon$ is at most $S_{\ell,i}$. Therefore, the number of iterations for which some design satisfies $R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}^k, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}^k, T) > \varepsilon$ is at most $\sum_{\ell=1}^N \sum_{i=1}^{M_\ell} S_{\ell,i} \leq S \sum_{\ell=1}^N M_\ell$. □

Theorem 3. For arbitrary $\varepsilon > 0$, D&R terminates in finite steps with an ε-optimal solution.

Proof. Define $\varepsilon' = \min\{N/(N+1), \varepsilon\}$ and $\varepsilon_{\max} = \varepsilon' / (NM_{\max} + (N+1)M_{\max}\varepsilon')$, where $M_{\max}$ is defined as in Lemma 2. As a result,

$$\varepsilon' = \frac{NM_{\max}\varepsilon_{\max}}{1 - (N+1)M_{\max}\varepsilon_{\max}}, \tag{25}$$

and

$$\varepsilon_{\max} = \frac{\varepsilon'}{NM_{\max} + (N+1)M_{\max}\varepsilon'} \leq \frac{\varepsilon'}{NM_{\max}} \leq \frac{1}{(N+1)M_{\max}}, \tag{26}$$

where the second inequality of (26) results because $\varepsilon' \leq N/(N+1)$. Using Equation (26), $M_{\max}\varepsilon_{\max}$ $\leq 1/(N+1) \leq 1$ and $NM_{\max}\varepsilon_{\max}/(1 - M_{\max}\varepsilon_{\max}) \leq 1$. By Lemma 6 (using $\bar{\tau}_{\ell,i}^k$ and $\bar{\bar{\tau}}_{\ell,i}^k$ defined in Lemma 5), the number of iterations for which some design satisfies $R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}^k, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}^k, T) > \varepsilon_{\max}$ is bounded. Since $M_{\max}\varepsilon_{\max} \leq 1$ and $NM_{\max}\varepsilon_{\max}/(1 - M_{\max}\varepsilon_{\max}) \leq 1$, Lemma 3 implies for $\bar{\mathbf{r}}$ and $\bar{\bar{\mathbf{r}}}$ (defined in Theorems 1–2) that $R^{\text{D}}(\bar{\bar{\mathbf{r}}}) - R^{\text{D}}(\bar{\mathbf{r}}) \leq \varepsilon' \leq \varepsilon$ after finite iterations. □

Having established convergence of D&R, we now discuss a general method for linearizing Model (4). Based upon Theorem 3, any finite approach to solving Model (4) yields a finite D&R algorithm for Model (3). One plausible approach to solving Model (4) is to linearize the model (and solve using a commericial mixed integer linear programming solver, such as CPLEX) in the vein of Sullivan [51] and O'Hanley et al. [41], who use a variable-expansion technique that represents probabilities of compound events as flows through a binary decision diagram. (As noted by Sullivan [51], this actually yields a valid linearized model even in the case of systems that are not series-parallel in structure; thus, D&R yields a finite algorithm for even more general systems than the one considered herein.) We have implemented such an approach but found that computational results favored an implementation of D&R that decomposes the original problem by resource. The resulting master and subproblems can each be solved via the D&R algorithm presented in this section, and the discrete versions of these problems are naturally formulated (without requiring a complex linearization approach) as mixed integer linear programs.

2.3.2 Discretize-and-Refine Decomposition Method

In this algorithm, we decompose Model (3) based on the observation that fixing the testing time allocated to each subsystem results in separable subproblems. In our problem, although the resulting subproblems are easier to solve, they are still nonlinear, nonconvex programs. We develop an approach (hereafter referred to as D&R decomposition, or D&RD) in which the master and subproblems iteratively are solved via D&R to yield a finite algorithm for Model (3). We now describe the optimization models used in D&RD.

The subproblem for subsystem $\ell = 1,\ldots,N$ is given as

$$R^{\mathrm{S}}(b_\ell, T) = \max\ 1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}(\tau_{\ell,i}, T)\right)^{V_{\ell,i}}, \tag{27a}$$

$$\text{s.t.} \sum_{i=1}^{M_\ell} c_{\ell,i}\tau_{\ell,i} \le b_\ell, \tag{27b}$$

$$\tau_{\ell,i} \ge \tau^0_{\ell,i}, \qquad \forall i = 1,\ldots,M_\ell, \tag{27c}$$

where $b_\ell \geq b_\ell^0 \equiv \sum_{i=1}^{M_\ell} c_{\ell,i}\tau_{\ell,i}^0$ is the budget available for testing in subsystem ℓ. (Because a different Model (27) is associated with each subsystem ℓ, the notation $R^{\mathrm{S}}(\cdot,T)$ could be subscripted by ℓ; however, we drop the cumbersome subscript because it is implied by the subscript on b.)

As before, D&R can be applied to generate arbitrarily tight bounds on $R^{\mathrm{S}}(b_\ell,T)$; however, this time the discretized model can be solved with increased efficiency, as we now explain. The discretized version of Model (27) is given as

$$R^{\mathrm{SD}}(b_\ell,\mathbf{r}_\ell) = \max\ 1-\prod_{i=1}^{M_\ell}\prod_{p=1}^{P}\left(1-r_{\ell,i}^p\right)^{V_{\ell,i}x_{\ell,i}^p}, \tag{28a}$$

$$\text{s.t. } \sum_{p=1}^{P} x_{\ell,i}^p = 1, \qquad \forall i=1,\ldots,M_\ell, \tag{28b}$$

$$\sum_{i=1}^{M_\ell}\sum_{p=1}^{P} c_{\ell,i}\tau_{\ell,i}^p x_{\ell,i}^p \leq b_\ell, \tag{28c}$$

$$x_{\ell,i}^p \in \{0,1\}, \qquad \forall i=1,\ldots,M_\ell,\ p=1,\ldots,P, \tag{28d}$$

where $\mathbf{r}_\ell$ denotes the vector of $r_{\ell,i}^p$-values corresponding to a single subsystem ℓ. Applying a natural logarithm to the product in Objective (28a) yields the equivalent (integer linear) model

$$w_\ell^*(b_\ell,\mathbf{r}_\ell) = \min \sum_{i=1}^{M_\ell}\sum_{p=1}^{P} \ln(1-r_{\ell,i}^p)V_{\ell,i}x_{\ell,i}^p, \tag{29}$$

$$\text{s.t. Constraints (28b)–(28d)},$$

where $R^{\mathrm{SD}}(b_\ell,\mathbf{r}_\ell) = 1-e^{w_\ell^*(b_\ell,\mathbf{r}_\ell)}$ provides the optimal objective value to Model (28).

The master problem allocates testing time to each subsystem. Letting $b_\ell,\ \ell=1,\ldots,N$ denote the budget allocated to subsystem ℓ, the master problem is given as

$$R^* = \max \prod_{\ell=1}^{N} R^{\mathrm{S}}(b_\ell,T), \tag{30a}$$

$$\text{s.t. } \sum_{\ell=1}^{N} b_\ell \leq b, \tag{30b}$$

$$b_\ell \geq b_\ell^0, \qquad \forall \ell=1,\ldots,N. \tag{30c}$$

To discretize Model (30), let

$$b_\ell^1 < b_\ell^2 < \cdots < b_\ell^P, \tag{31}$$

for each subsystem $\ell = 1,\ldots,N$ denote P candidate testing times. Let $x_\ell^p = 1$ if the p-th testing time is selected in subsystem ℓ, and 0 otherwise. Similarly, let r_ℓ^p denote the subsystem ℓ reliability (i.e., $r_\ell^p = R^S(b_\ell^p, T)$) if its p-th testing time is selected. The discretized master problem is given as

$$R^{\mathrm{MD}}(\mathbf{r}) = \max \prod_{\ell=1}^{N}\prod_{p=1}^{P} r_\ell^{p x_\ell^p}, \tag{32a}$$

$$\text{s.t.} \sum_{p=1}^{P} x_\ell^p = 1, \qquad \forall \ell = 1,\ldots,N, \tag{32b}$$

$$\sum_{\ell=1}^{N}\sum_{p=1}^{P} b_\ell^p x_\ell^p \le b, \tag{32c}$$

$$x_\ell^p \in \{0,1\}, \qquad \forall \ell = 1,\ldots,N,\ p = 1,\ldots,P, \tag{32d}$$

where $\mathbf{r}$ denotes the vector of r_ℓ^p-values. Applying a logarithm to the Objective (32a) again yields an equivalent linear model, given as $w^*(\mathbf{r}) = \max\left\{ \sum_{\ell=1}^{N}\sum_{p=1}^{P} \ln(r_\ell^p) x_\ell^p \,\middle|\, \text{Constraints (32b)–(32d)} \right\}$, where $e^{w^*(\mathbf{r})}$ provides the optimal objective value to Model (32).

One of the difficulties in using the discretized Model (32) in solving Model (3) is that the constants $r_\ell^p = R^{\mathrm{S}}(b_\ell^p, T)$ are difficult to compute, each requiring solution of an instance of Model (29). In fact, applying D&R to Model (29) yields only bounds on $R^{\mathrm{S}}(b_\ell^p, T)$, which further complicates the development of D&RD. We now extend the key results of Section 2.3.1 to establish convergence of D&RD. Regarding vectors $\mathbf{r}$ and $\bar{\mathbf{r}}$ of r_ℓ^p-values, we say that $\mathbf{r} \le \bar{\mathbf{r}}$ if $r_\ell^p \le \bar{r}_\ell^p$ for all $\ell = 1,\ldots,N$, and $p = 1,\ldots,P$. As before, we let b_ℓ^{P+1} refer to $b_\ell^{\max} \equiv b + b_\ell^0 - \sum_{\ell'=1}^{N} b_{\ell'}^0$. The lower- and upper-bound results for Model (32) follow.

Theorem 4. Define $\mathbf{r}$ by

$$r_\ell^p = R^{\mathrm{S}}(b_\ell^p, T),\ \forall p = 1,\ldots,P. \tag{33}$$

If $\bar{\mathbf{r}} \le \mathbf{r}$, then $R^{\mathrm{MD}}(\bar{\mathbf{r}}) \le R^*$.

Proof. The result follows directly from Theorem 1 (under $M_\ell = 1,\ \ell = 1,\ldots,N$) if $\bar{\mathbf{r}} = \mathbf{r}$. If $\bar{\mathbf{r}} \leq \mathbf{r}$, the result holds because $R^{\mathrm{MD}}(\cdot)$ is nondecreasing. □

Theorem 5. Define $\mathbf{r}$ by

$$r_\ell^p = R^{\mathrm{S}}(b_\ell^{p+1}, T),\ \forall p = 1,\ldots,P. \tag{34}$$

If $\bar{\bar{\mathbf{r}}} \geq \mathbf{r}$, then $R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}}) \geq R^*$.

Proof. If $\bar{\bar{\mathbf{r}}} = \mathbf{r}$, Theorem 2 implies the result under the special case $M_\ell = 1,\ \ell = 1,\ldots,N$. If $\bar{\bar{\mathbf{r}}} \geq \mathbf{r}$, the result holds because $R^{\mathrm{MD}}(\cdot)$ is nondecreasing. □

In what follows, we refer to Model (32) as UB-MD when providing an upper bound on R^* and as LB-MD when providing a lower bound. A formal statement of D&RD follows.

Step 0: Let $\varepsilon_{\mathrm{sub}}$ and P_{sub} denote the subsystem tolerance and initial number of points for each design, respectively. Also, define ε as the system tolerance. Set $b_\ell^{\min} = b_\ell^0$ and $b_\ell^{\max} = b + b_\ell^0 - \sum_{\ell'=1}^{N} b_{\ell'}^0$. For each subsystem $\ell = 1,\ldots,N$, define P_{int} initial budget according to

$$b_\ell^p = b_\ell^{\min} + \frac{(p-1)}{P_{\mathrm{int}}}\left(b_\ell^{\max} - b_\ell^{\min}\right),\ \forall \ell = 1,\ldots,N,\ p = 1,\ldots,P_{\mathrm{int}}. \tag{35}$$

Denote b^* as the incumbent solution. Set $P \leftarrow P_{\mathrm{int}}$, $UB \leftarrow 1$ and $LB \leftarrow 0$.

Step 1: Solve subproblems (28) via D&R with tolerance $\varepsilon_{\mathrm{sub}}$ and initial number of points P_{sub} for each $p = 1,\ldots,P$, and $\ell = 1,\ldots,N$ to obtain U_ℓ^p and L_ℓ^p such that

$$L_\ell^p \leq R^{\mathrm{S}}(b_\ell^p, T) \leq U_\ell^p, \tag{36}$$

and $U_\ell^p - L_\ell^p \leq \varepsilon_{\mathrm{sub}}$. Define $\bar{\mathbf{r}}$ according to

$$\bar{r}_\ell^p = L_\ell^p,\ \forall \ell = 1,\ldots,N,\ p = 1,\ldots,P, \tag{37}$$

and $\bar{\bar{\mathbf{r}}}$ according to

$$\bar{\bar{r}}_\ell^p = U_\ell^{p+1},\ \forall \ell = 1,\ldots,N,\ p = 1,\ldots,P. \tag{38}$$

Step 2: If $UB - LB \leq \varepsilon$, stop (b^* is ε-optimal); otherwise, go to Step 3.

Step 3: Solve LB-MD Model (32) using $\bar{\mathbf{r}}$ as defined in Equation (37) to obtain an optimal solution $\hat{b}$ with objective value $R^{\mathrm{MD}}(\bar{\mathbf{r}})$. (Because $\bar{\mathbf{r}} \leq \mathbf{r}$ for $\mathbf{r}$ defined in Equation (33), Theorem 4 guarantees $R^{\mathrm{MD}}(\bar{\mathbf{r}}) \leq R^*$.) If $R^{\mathrm{MD}}(\bar{\mathbf{r}}) > LB$, set $LB \leftarrow R^{\mathrm{MD}}(\bar{\mathbf{r}})$ and $b^* \leftarrow \hat{b}$.

Step 4: Solve UB-MD Model (32) using $\bar{\bar{\mathbf{r}}}$ as defined in Equation (38) to obtain an optimal solution P_ℓ^* with objective value $R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}})$. (Because $\bar{\bar{\mathbf{r}}} \geq \mathbf{r}$ for $\mathbf{r}$ defined in Equation (34), Theorem 5 guarantees $R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}}) \geq R^*$.) Set $P \leftarrow P + 1$. For each subsystem $\ell = 1, \ldots, N$, do the following:

4a: Define a new candidate test time $b_\ell^P = 0.5\left(b_\ell^{P_\ell^*} + b_\ell^{P_\ell^*+1}\right)$.

4b: Solve subproblems (27) via D&R with tolerance $\varepsilon_{\mathrm{sub}}$ and initial number of points P_{sub} to obtain U_ℓ^P and L_ℓ^P such that

$$L_\ell^P \leq R^{\mathrm{S}}(b_\ell^P, T) \leq U_\ell^P. \tag{39}$$

4c: Renumber each of the candidate test times to ensure $b_\ell^1 < b_\ell^2 < \cdots < b_\ell^P$.

Step 5: If $R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}}) < UB$, set $UB \leftarrow R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}})$.

Step 6: Go to Step 2.

We now prove finite convergence of D&RD to an ε-optimal solution. As in the convergence proof for D&R, we first establish that the master problem discretization occurs over a Lipschitz function (i.e., such that a given resolution in the objective function can be produced by a fine enough discretization).

Lemma 7. For each $\varepsilon > 0$, there exists $\delta_\ell > 0$ such that $R^{\mathrm{S}}(b_\ell + \delta_\ell, T) - R^{\mathrm{S}}(b_\ell, T) \leq \varepsilon$ for $b_\ell^0 \leq b_\ell \leq b$, $\ell = 1, \ldots, N$.

Proof. As in the proof of Theorem 3, define $\varepsilon' = \min\{N/(N+1), \varepsilon\}$ and $\varepsilon_{\max} = \varepsilon'/[NM_\ell + (N+1)M_\ell\varepsilon']$ where $N \equiv 1$. By Lemma 4, there exists $\delta_{\ell,i} > 0$ such that $R_{\ell,i}(\tau_{\ell,i} + \delta_{\ell,i}, T) - R_{\ell,i}(\tau_{\ell,i}, T) \leq$

$\varepsilon_{\max}$ for $\tau^0_{\ell,i} \le \tau_{\ell,i} \le \tau^{\max}_{\ell,i}$. Let $\delta_\ell = \min\{\delta_{\ell,i}\}_{i=1}^{M_\ell}$. For an optimal solution $\bar{\bar{\tau}}_{\ell,i},\ i = 1,\ldots,M_\ell$, to Model (27) with objective value $R^{\mathrm{S}}(b_\ell + \delta_\ell, T)$, define $\bar{\tau}_{\ell,i} = \max\{\bar{\bar{\tau}}_{\ell,i} - \delta_\ell/M_\ell, \tau^0_{\ell,i}\}$. We have that

$$R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}, T) \le \varepsilon_{\max},\ \forall \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell. \tag{40}$$

Now, applying Lemma 2 with $N = 1$ yields

$$\left[1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}\left(\bar{\bar{\tau}}_{\ell,i}, T\right)\right)^{V_{\ell,i}}\right] - \left[1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}(\bar{\tau}_{\ell,i}, T)\right)^{V_{\ell,i}}\right] \le \varepsilon. \tag{41}$$

(Note that, as in Theorem 3, we have that $M_\ell \varepsilon_{\max} \le 1$ and $N M_\ell \varepsilon_{\max}/(1 - M_\ell \varepsilon_{\max}) \le 1$.) Because $\bar{\tau}_{\ell,i}$ is feasible to Model (27) corresponding to b_ℓ, we have that

$$R^{\mathrm{S}}(b_\ell, T) \ge 1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}\left(\bar{\tau}_{\ell,i}, T\right)\right)^{V_{\ell,i}}. \tag{42}$$

From the definition of $\bar{\bar{\tau}}_{\ell,i}$,

$$R^{\mathrm{S}}(b_\ell + \delta_\ell, T) = 1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}\left(\bar{\bar{\tau}}_{\ell,i}, T\right)\right)^{V_{\ell,i}}, \tag{43}$$

and Equation (41) thus yields the result. □

Lemma 8. Given $\delta_\ell > 0$, define $S = \lfloor 2b/\delta_\ell \rfloor$. Allow $P^k_\ell,\ \ell = 1,\ldots,N$, to denote the index $p \in \{1,\ldots,P\}$ such that $\bar{x}^p_\ell = 1$ in the obtained optimal solution to UB-MD in iteration k of the algorithm, and define $\bar{b}^k_\ell = b_\ell^{P^k_\ell}$ and $\bar{\bar{b}}^k_\ell = b_\ell^{P^k_\ell + 1}$. Then $\Delta^k_\ell \equiv \bar{\bar{b}}^k_\ell - \bar{b}^k_\ell > \delta_\ell$ for at most S iterations k.

Proof. This follows analogously to Lemma 5 upon replacing $\Delta^k_{\ell,i}$ with Δ^k_ℓ, $N_{\ell,i}$ with N_ℓ, and $\bar{\tau}^k_{\ell,i}$, $\bar{\bar{\tau}}^k_{\ell,i}$, $\tau^{\min}_{\ell,i}$, and $\tau^{\max}_{\ell,i}$, respectively, with $\bar{b}^k_\ell$, $\bar{\bar{b}}^k_\ell$, $b^{\min}_\ell$, and $b^{\max}_\ell$, and the fact that the cost coefficients are equal to 1 in Model (30). □

Lemma 9. Let $\bar{b}^k_\ell$ and $\bar{\bar{b}}^k_\ell$ be defined as in Lemma 8. For each $\varepsilon > 0$, the number of iterations for which some subsystem ℓ satisfies $R^{\mathrm{S}}(\bar{\bar{b}}^k_\ell, T) - R^{\mathrm{S}}(\bar{b}^k_\ell, T) > \varepsilon$ is bounded.

Proof. By Lemma 7, there exists $\delta_\ell > 0$ such that $R^{\mathrm{S}}(b_\ell + \delta_\ell, T) - R^{\mathrm{S}}(b_\ell, T) \leq \varepsilon$, for $b_\ell^0 \leq b_\ell \leq b$. Define $\delta = \min_{\ell=1,\ldots,N}\{\delta_\ell\}$ and $S = \lfloor 2b/\delta \rfloor$. By Lemma 8, the number of iterations k for which subsystem ℓ satisfies $\bar{\bar{b}}_\ell^k - \bar{b}_\ell^k > \delta$ is at most S; thus, the number of iterations k for which subsystem ℓ satisfies $R^{\mathrm{S}}(\bar{\bar{b}}_\ell^k, T) - R^{\mathrm{S}}(\bar{b}_\ell^k, T) > \varepsilon$ is at most S. Therefore, the number of iterations for which some subsystem satisfies $R^{\mathrm{S}}(\bar{\bar{b}}_\ell^k, T) - R^{\mathrm{S}}(\bar{b}_\ell^k, T) > \varepsilon$ is at most NS. □

Theorem 6. For arbitrary $\varepsilon > 0$, D&RD terminates in finite steps with an ε-optimal solution.

Proof. Define $\varepsilon' = \min\{N/(N+1), \varepsilon\}$ and $\varepsilon'' = \varepsilon' / (NM + (N+1)M\varepsilon')$ where $M = 1$. Let $\varepsilon_{\max} = \varepsilon_{\mathrm{sub}} = \varepsilon''/3$. Following as Equation (26), we have that

$$\varepsilon'' = \frac{\varepsilon'}{NM + (N+1)M\varepsilon'} \leq \frac{\varepsilon'}{NM} \leq \frac{1}{(N+1)M}, \tag{44}$$

implying both $M\varepsilon'' \leq 1$ and $NM\varepsilon''/(1 - M\varepsilon'') \leq 1$.

Lemma 7 guarantees existence of $\delta_\ell > 0,\ \ell = 1,\ldots,N$, such that $R^{\mathrm{S}}(b_\ell + \delta_\ell, T) - R^{\mathrm{S}}(b_\ell, T) \leq \varepsilon_{\max}$ for $b_\ell^0 \leq b_\ell \leq b$. Let $\delta = \min_{\ell=1\ldots,N}\{\delta_\ell\}$. By Lemma 9, after at most $N\lfloor 2b/\delta \rfloor + 1$ iterations, we reach an iteration of D&RD where—using P_ℓ^k as defined in Lemma 8—$R^{\mathrm{S}}(b_\ell^{P_\ell^k+1}, T) - R^{\mathrm{S}}(b_\ell^{P_\ell^k}, T) \leq \varepsilon_{\max}$, for all $\ell = 1,\ldots,N$. In this iteration, Step 1 of D&RD indicates that

$$L_\ell^{P_\ell^k} \leq R^{\mathrm{S}}\left(b_\ell^{P_\ell^k}, T\right) \leq R^{\mathrm{S}}\left(b_\ell^{P_\ell^k+1}, T\right) \leq U_\ell^{P_\ell^k+1},\ \forall \ell = 1,\ldots,N, \tag{45}$$

where

$$\begin{aligned}
U_\ell^{P_\ell^k+1} - L_\ell^{P_\ell^k} &= \left[U_\ell^{P_\ell^k+1} - R^{\mathrm{S}}\left(b_\ell^{P_\ell^k+1}, T\right)\right] + \left[R^{\mathrm{S}}\left(b_\ell^{P_\ell^k+1}, T\right) - R^{\mathrm{S}}\left(b_\ell^{P_\ell^k}, T\right)\right] \\
&\quad + \left[R^{\mathrm{S}}\left(b_\ell^{P_\ell^k}, T\right) - L_\ell^{P_\ell^k}\right], && \text{(46a)} \\
&\leq \left[U_\ell^{P_\ell^k+1} - L_\ell^{P_\ell^k+1}\right] + \left[R^{\mathrm{S}}\left(b_\ell^{P_\ell^k+1}, T\right) - R^{\mathrm{S}}\left(b_\ell^{P_\ell^k}, T\right)\right] + \left[U_\ell^{P_\ell^k} - L_\ell^{P_\ell^k}\right], && \text{(46b)} \\
&\leq \varepsilon_{\mathrm{sub}} + \varepsilon_{\max} + \varepsilon_{\mathrm{sub}}, && \text{(46c)} \\
&= \varepsilon''. && \text{(46d)}
\end{aligned}$$

Now, because $M\varepsilon'' \leq 1$ and $NM\varepsilon''/(1-M\varepsilon'') \leq 1$, it follows from Lemma 1 (under $M_\ell = 1,\ \ell = 1,\ldots,N$, $\bar{\bar{\rho}}_\ell = U_\ell^{P_\ell^k+1}$, and $\bar{\rho}_\ell = L_\ell^{P_\ell^k}$) that $R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}}) - R^{\mathrm{MD}}(\bar{\mathbf{r}}) \leq \varepsilon' \leq \varepsilon$ (where $\bar{\mathbf{r}}$ and $\bar{\bar{\mathbf{r}}}$ are defined as in Step 1 of D&RD). □

Having established that D&RD terminates finitely, we now overview simplifications that result for subsystems ℓ in which $M_\ell = 1$.

Remark 1. When $M_\ell = 1$, Model (27) has a single variable ($\tau_{\ell,1}$) over which the objective is nondecreasing. Thus, $\tau_{\ell,1} = b_\ell / c_{\ell,1}$ is optimal for this problem. In Step 1 of the D&RD algorithm, this allows us to obtain $R^{\mathrm{S}}(b_\ell^p, T)$ directly (i.e., without applying the D&R algorithm). We would then set $L_\ell^p = U_\ell^p = R^{\mathrm{S}}(b_\ell^p, T)$ and proceed with the algorithm. □

We note that there are other ways of exploiting the decomposition structure used in D&RD to arrive at an exact algorithm for our problem. For instance, the feasible region of the master and/or subproblem could be explored via a branch and bound scheme wherein each branch restricts each testing time variable to fall within an interval and branching is used to refine these intervals over time. Preliminary computational results demonstrated that such an approach consistently results in larger optimality gaps than D&RD for instances with more than two subsystems, and so we have foregone its formal development.

2.4 Extension to the Multi-Level Testing Problem

In this section, we extend the results of Sections 2.2 and 2.3 to the problem of allocating test times in a series-parallel system with consideration for the possibility of testing at different levels (system, subsystem, and component). Motivated by Pohl and Dietrich [43], we model the concept of the connection-induced failure through the addition of pseudo-components to consider the connections in the system and subsystems. The idea is this: connecting components (subsystems) into a subsystem (system) may induce additional failure modes. In order to remove these failure modes, testing must be performed at a high enough assembly level. System-level testing is more expensive, but also more comprehensive rather than subsystem- and component-level testing in

that it may identify failure modes present in any component or connection. In Section 2.4.1, we state an optimization model that incorporates these tradeoffs among component-, subsystem-, and system-level testing activities.

2.4.1 Optimization Model for Multi-Level Testing

In what follows, a reliability growth model that accounts for component-, subsystem-, and system-level testing is given. The following notation will be used throughout this section.

Parameters

λ_ℓ, β_ℓ	AMSAA model parameters for subsystem-level connections in subsystem $\ell = 1, \ldots, N$
λ, β	AMSAA model parameters for system-level connection
c_ℓ	Cost per unit time of testing subsystem $\ell = 1, \ldots, N$
c	Cost per unit time of system-level testing
L_ℓ	Scaling factor of subsystem-level connection $\ell = 1, \ldots, N$ in system-level testing
$L_{\ell,i}$	Scaling factor of design $i = 1, \ldots, M_\ell$ of subsystem $\ell = 1, \ldots, N$ in system-level testing
$J_{\ell,i}$	Scaling factor of design $i = 1, \ldots, M_\ell$ of subsystem $\ell = 1, \ldots, N$ in subsystem-level testing
τ_ℓ^0	Initial subsystem-level testing time of subsystem $\ell = 1, \ldots, N$ (assumed to be positive)
τ^0	Initial system-level testing time (assumed to be positive)

Decision Variables

$\tau_\ell,\ \bar{\tau}_\ell$	Subsystem-level testing time of subsystem $\ell = 1, \ldots, N$
$\tau,\ \bar{\tau}$	System-level testing time

We now present a formal definition of the problem. The system comprises N subsystems connected in a series. Each subsystem $\ell = 1, \ldots, N$ has designs $(\ell, i),\ i = 1, \ldots, M_\ell$ in parallel, with $V_{\ell,i}$ redundant components for design $i = 1, \ldots, M_\ell$. Connection-induced failure is modeled through the

addition of a *pseudo-component* associated with the system and each subsystem. After adding pseudo-components, the system's structure is depicted in Figure 2. Therefore, a system failure takes place if a subsystem- or system-level connection has a failure or if all components in a subsystem fail. We assume that all components and pseudo-components exhibit reliability growth according to an AMSAA model, and that testing at a higher-level assembly (i.e., the outlined boxes in Figure 2) improves the reliability of all connections and component designs included in that assembly. We use scaling factors $0 \leq J_{\ell,i} \leq 1$ and $0 \leq L_{\ell,i} \leq 1$ for design (ℓ, i) in subsystem- and system-level testing respectively to take into account subsystem- and system-level testing effect on components' reliabilities. For instance, $L_{\ell,i} = 0.1$ would indicate an hour of dedicated testing on component design (ℓ, i) would yield the same improvement to that design as would 0.1 hours of testing on an assembled system that contains design (ℓ, i). Also, we incorporate the scaling factor $0 \leq L_\ell \leq 1$ to capture the effect of system-level testing in subsystem-level connection ℓ. We use scaling factors because the subsystem- and system-level testing are less effective in revealing failure modes of a component than dedicated component-level testing. We define the shorthand notation $R_\ell(\tau_\ell, T) \equiv R(\tau_\ell; T, \lambda_\ell, \beta_\ell)$ and $R(\tau, T) \equiv R(\tau; T, \lambda, \beta)$, respectively, to refer to the reliability of subsystem ℓ's connection and the system's connection.

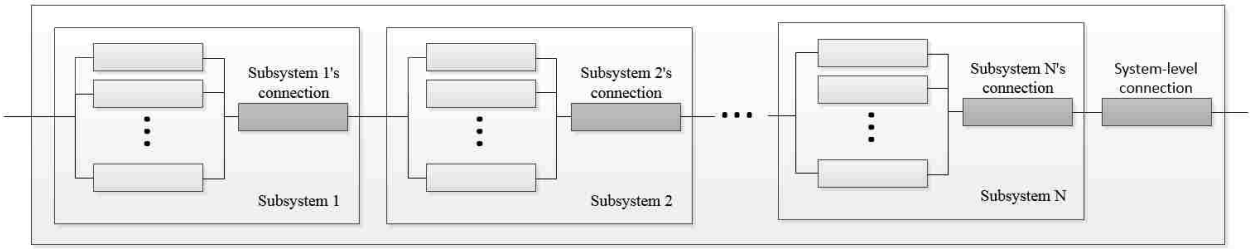


Figure 2: A series-parallel system considering subsystem- and system-level connection's failures

This problem may be formulated as

$$R^* = \max R(\tau, T) \prod_{\ell=1}^{N} \left(R_\ell(\tau_\ell + L_\ell \tau, T) \left[1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}(\tau_{\ell,i} + J_{\ell,i}\tau_\ell + L_{\ell,i}\tau, T) \right)^{V_{\ell,i}} \right] \right), \quad (47a)$$

$$\text{s.t.} \sum_{\ell=1}^{N} \sum_{i=1}^{M_\ell} c_{\ell,i}\tau_{\ell,i} + \sum_{\ell=1}^{N} c_\ell \tau_\ell + c\tau \leq b, \quad (47b)$$

$$\tau_{\ell,i} \geq \tau_{\ell,i}^0, \qquad \forall \ell = 1, \ldots, N,\ i = 1, \ldots, M_\ell, \quad (47c)$$

$$\tau_\ell \geq \tau_\ell^0, \qquad \forall \ell = 1,\ldots,N, \tag{47d}$$

$$\tau \geq \tau^0. \tag{47e}$$

Objective (47a) maximizes the reliability of the system (computed as the product of all subsystems and connections' reliabilities) through a mission of length T. Equation (47b) limits the total cost of testing, where b is the available budget and $c_{\ell,i}$, c_ℓ, and c represent the cost per unit time of the testing design (ℓ,i), the assembled subsystem ℓ, and the assembled system. Constraint sets (47c)–(47e) ensure the testing times for the design (ℓ,i), the subsystem-level connection ℓ, and the system-level connection should be respectively greater than the initial testing times $\tau_{\ell,i}^0$, τ_ℓ^0, and τ^0. Noting the monotonicity of $R_{\ell,i}(\tau_{\ell,i}+J_{\ell,i}\tau_\ell+L_{\ell,i}\tau,T)$, Model (47) can equivalently be formulated, after adding (continuous) variables $\bar{\tau}$ and $\bar{\tau}_\ell$, $\ell = 1,\ldots,N$, as

$$R^* = \max R(\tau,T)\prod_{\ell=1}^{N}\left(R_\ell(\tau_\ell+L_\ell\tau,T)\left[1-\prod_{i=1}^{M_\ell}\left(1-R_{\ell,i}(\tau_{\ell,i}+J_{\ell,i}\bar{\tau}_\ell+L_{\ell,i}\bar{\tau},T)\right)^{V_{\ell,i}}\right]\right), \tag{48a}$$

$$\text{s.t. Constraints (47b)–(47c)}, \tag{48b}$$

$$\tau_\ell^0 \leq \bar{\tau}_\ell \leq \tau_\ell, \qquad \forall \ell = 1,\ldots,N, \tag{48c}$$

$$\tau^0 \leq \bar{\tau} \leq \tau. \tag{48d}$$

The following section explains how the algorithm of Section 2.3.2 can be extended to solve Model (48).

2.4.2 Discretize-and-Refine Decomposition Method for Multi-Level Testing

In this algorithm, we decompose Model (48) into a master problem and N subproblems. In subproblem $\ell = 1,\ldots,N$, resources are allocated, given fixed system-level test time $\bar{\tau} \geq \tau^0$ and susbsystem-level test times $\bar{\tau}_\ell \geq \tau_\ell^0$, $\ell = 1,\ldots,N$, to the designs of subsystem ℓ. The subproblem ℓ

for fixed value of $\bar{\tau}_\ell$ and $\bar{\tau}$, and available budget $b_\ell \geq b_\ell^0 \equiv \sum_{i=1}^{M_\ell} c_{\ell,i}$ can be stated as

$$R^{\mathrm{S}}(b_\ell, \bar{\tau}_\ell, \bar{\tau}, T) = \max \left[1 - \prod_{i=1}^{M_\ell} \left(1 - R_{\ell,i}(\tau_{\ell,i} + J_{\ell,i}\bar{\tau}_\ell + L_{\ell,i}\bar{\tau}, T)\right)^{V_{\ell,i}}\right], \tag{49a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} \sum_{i=1}^{M_\ell} c_{\ell,i}\tau_{\ell,i} \leq b_\ell, \tag{49b}$$

$$\tau_{\ell,i} \geq \tau_{\ell,i}^0, \qquad \forall i = 1, \ldots, M_\ell, \tag{49c}$$

in which $R^{\mathrm{S}}(b_\ell, \bar{\tau}_\ell, \bar{\tau}, T)$ defines the reliability of subproblem ℓ for budget b_ℓ, subsystem testing $\bar{\tau}_\ell$, and system-level testing $\bar{\tau}$. The subproblem (49) is similar to Model (27), so we use the discretize-and-refine method to solve subproblem (49).

In the master problem, we determine the system-level testing time τ, the testing time τ_ℓ for each subsystem $\ell = 1, \ldots, N$, and the budget b_ℓ allocated to dedicated testing on designs of subsystem $\ell = 1, \ldots, N$. The master problem is defined as follows:

$$R^* = \max R(\tau, T) \prod_{\ell=1}^{N} \left(R_\ell(\tau_\ell + L_\ell\tau, T) R^{\mathrm{S}}(b_\ell, \bar{\tau}_\ell, \bar{\tau}, T)\right), \tag{50a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} b_\ell + \sum_{\ell=1}^{N} c_\ell\tau_\ell + c\tau \leq b, \tag{50b}$$

$$b_\ell \geq b_\ell^0, \qquad \forall \ell = 1, \ldots, N, \tag{50c}$$

$$\tau_\ell^0 \leq \bar{\tau}_\ell \leq \tau_\ell, \qquad \forall \ell = 1, \ldots, N, \tag{50d}$$

$$\tau^0 \leq \bar{\tau} \leq \tau. \tag{50e}$$

We let $\tau^p,\ p = 1, \ldots, P$ denote candidate values of τ such that $(\tau^0 \equiv)\tau^1 < \tau^2 < \cdots < \tau^P$ and define $r^p = R(\tau^p, T)$ as the system connection reliability associated with τ^p. For each $p = 1, \ldots, P$, we define $\tau_\ell^{p,q},\ q = 1, \ldots, Q_\ell^p$ as candidate values of τ_ℓ that are available when $\tau = \tau^p$, such that $(\tau_\ell^0 \equiv)\tau_\ell^{p,1} < \tau_\ell^{p,2} < \cdots < \tau_\ell^{p,Q_\ell^p}$. Let $r_\ell^{p,q} = R_\ell(\tau_\ell^{p,q} + L_\ell\tau^p)$ denote the subsystem ℓ connection reliability associated with $\tau_\ell^{p,q}$. Similar to the discretization of τ and τ_ℓ, we discretize $\bar{\tau}$ and $\bar{\tau}_\ell$; however, because solving subproblems can be time consuming, we limit the candidate values of $\bar{\tau}$

and $\bar{\tau}_\ell$ to a subset of those considered for τ and τ_ℓ. Let $p(j) \in \{1,\ldots,P\},\ j = 1,\ldots,J$ denote a subset of indices such that

$$p(1) < p(2) < \cdots < p(J). \tag{51}$$

We define $\tau^{p(j)},\ j = 1,\ldots,J$ as the candidate values of $\bar{\tau}$. Similarly, let $q(k) \in 1,\ldots,Q_\ell^{p(j)},\ k = 1,\ldots,K_\ell^j$, denote a subsequence of indices such that

$$q(1) < q(2) < \cdots < q(K_\ell^j). \tag{52}$$

We define $\tau_\ell^{p(j),q(k)},\ j = 1,\ldots,J,\ k = 1,\ldots,K_\ell^j$ as the candidate values of $\bar{\tau}_\ell$. Associated with each $j = 1,\ldots,J$, and $k = 1,\ldots,K_\ell^k$, we define $b_\ell^{j,k,s},\ s = 1,\ldots,S_\ell^{j,k}$, as the candidate values of $b_\ell,\ \ell = 1,\ldots,N$, to consider when $\bar{\tau} = \tau^{p(j)}$ and $\bar{\tau}_\ell = \tau_\ell^{p(j),q(k)}$. Moreover, we define $r_\ell^{j,k,s} = R^{\mathrm{S}}(b_\ell^{j,k,s}, \tau_\ell^{p(j),q(k)}, \tau^{p(j)}, T)$ as the subsystem reliability associated with system testing time $\tau^{p(j)}$, subsystem testing time $\tau_\ell^{p(j),q(k)}$, and budget $b_\ell^{j,k,s}$. The notation to discretize the master problem (50) is defined as follows

Parameters

P	Number of candidate values of τ (index by $p = 1,\ldots,P$)
Q_ℓ^p	For each $\ell = 1,\ldots,N$, number of candidate values for τ_ℓ when $\tau = \tau^p$ (index by $q = 1,\ldots,Q_\ell^p$)
J	Number of candidate values of $\bar{\tau}$ (index by $j = 1,\ldots,J$)
K_ℓ^j	Number of candidate values for $\bar{\tau}_\ell$ when $\bar{\tau} = \tau^{p(j)}$ (index by $k = 1,\ldots,K_\ell^p$)
$S_\ell^{j,k}$	Number of candidate values for budget b_ℓ when $\bar{\tau}_\ell = \tau_\ell^{p(j),q(k)}$ and $\bar{\tau} = \tau^{p(j)}$ (index by $s = 1,\ldots,S_\ell^{j,k}$)
τ^p	p-th candidate value of τ ($p = 1,\ldots,P$)
r^p	System connection reliability associated with τ^p ($p = 1,\ldots,P$)
$\tau_\ell^{p,q}$	q-th candidate value of τ_ℓ when $\tau = \tau^p$ ($p = 1,\ldots,P$; $q = 1,\ldots,Q_\ell^p$)
$r_\ell^{p,q}$	Subsystem ℓ connection reliability associated with $\tau_\ell^{p,q}$ ($p = 1,\ldots,P$; $q = 1,\ldots,Q_\ell^p$)

$\tau^{p(j)}$ j-th candidate value of $\bar{\tau}$ ($j = 1, \ldots, J$)

$\tau_\ell^{p(j),q(k)}$ k-th candidate value of $\bar{\tau}_\ell$ when $\tau = \tau^{p(j)}$ ($j = 1, \ldots, J$; $k = 1, \ldots, K_\ell^p$)

$b_\ell^{j,k,s}$ s-th candidate value of budget when $\bar{\tau}_\ell = \tau_\ell^{p(j),q(k)}$ and $\bar{\tau} = \tau^{p(j)}$ ($\ell = 1, \ldots, N$; $j = 1, \ldots, J$; $k = 1, \ldots, K_\ell^p$; $s = 1, \ldots, S_\ell^{j,k}$)

$r_\ell^{j,k,s}$ Subsystem reliability when $b_\ell = b_\ell^{j,k,s}$, $\bar{\tau}_\ell = \tau_\ell^{p(j),q(k)}$, and $\bar{\tau} = \tau^{p(j)}$ ($\ell = 1, \ldots, N$; $j = 1, \ldots, J$; $k = 1, \ldots, K_\ell^p$; $s = 1, \ldots, S_\ell^{j,k}$)

Decision Variables

$$x_\ell^{p,q,s} \quad \begin{cases} 1 & \text{if } b_\ell = b_\ell^{j,k,s},\ \bar{\tau}_\ell = \tau_\ell^{p(j),q(k)}, \text{ and } \bar{\tau} = \tau^{p(j)} \\ 0 & \text{otherwise} \end{cases}$$

$(\ell = 1, \ldots, N;\ j = 1, \ldots, J;\ k = 1, \ldots, K_\ell^p;\ s = 1, \ldots, S_\ell^{j,k})$

$$y^p \quad \begin{cases} 1 & \text{if } \tau = \tau^p \\ 0 & \text{otherwise} \end{cases}$$

$(p = 1, \ldots, P)$

$$z_\ell^{p,q} \quad \begin{cases} 1 & \text{if } \tau_\ell = \tau_\ell^{p,q} \text{ and } \tau = \tau^p \\ 0 & \text{otherwise} \end{cases}$$

$(p = 1, \ldots, P;\ \ell = 1, \ldots, N;\ q = 1, \ldots, Q_\ell^p)$

$$\bar{y}^j \quad \begin{cases} 1 & \text{if } \bar{\tau} = \tau^{p(j)} \\ 0 & \text{otherwise} \end{cases}$$

$(j = 1, \ldots, J)$

$$\bar{z}_\ell^{j,k} \quad \begin{cases} 1 & \text{if } \bar{\tau}_\ell = \tau_\ell^{p(j),q(k)} \text{ and } \bar{\tau} = \tau^{p(j)} \\ 0 & \text{otherwise} \end{cases}$$

$(j = 1, \ldots, J;\ \ell = 1, \ldots, N;\ k = 1, \ldots, K_\ell^j)$

To discretize the master problem, we replace variables τ, τ_ℓ, b_ℓ, $\bar{\tau}$, $\bar{\tau}_\ell$ respectively with $\sum_{p=1}^{P} \tau^p y^p$, $\sum_{p=1}^{P} \sum_{q=1}^{Q_\ell^p} \tau_\ell^{p,q} z_\ell^{p,q}$, $\sum_{p=1}^{P} \sum_{q=1}^{Q_\ell^p} \sum_{s=1}^{S_\ell^{p,q}} b_\ell^{p,q,s} x_\ell^{p,q,s}$, $\sum_{j=1}^{J} \tau^{p(j)} \bar{y}^j$, and $\sum_{j=1}^{J} \sum_{k=1}^{K_\ell^j} \tau_\ell^{p(j),q(k)} \bar{z}_\ell^{j,k}$. The dis-

cretized version of Model (50) is given as

$$R^{\mathrm{MD}}(\mathbf{r}) = \max \prod_{p=1}^{P} (r^p)^{y^p} \prod_{\ell=1}^{N} \prod_{p=1}^{P} \prod_{q=1}^{Q_\ell^p} (r_\ell^{p,q})^{z_\ell^{p,q}} \prod_{\ell=1}^{N} \prod_{j=1}^{J} \prod_{k=1}^{K_\ell^j} \prod_{s=1}^{S_\ell^{j,k}} (r_\ell^{j,k,s})^{x_\ell^{j,k,s}}, \tag{53a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} \sum_{j=1}^{J} \sum_{k=1}^{K_\ell^j} \sum_{s=1}^{S_\ell^{j,k}} b_\ell^{j,k,s} x_\ell^{j,k,s} + \sum_{\ell=1}^{N} c_\ell \sum_{p=1}^{P} \sum_{q=1}^{Q_\ell^p} \tau_\ell^{p,q} z_\ell^{p,q} + c \sum_{p=1}^{P} \tau^p y^p \le b, \tag{53b}$$

$$\sum_{j=1}^{J} \sum_{k=1}^{K_\ell^j} \sum_{s=1}^{S_\ell^{j,k}} b_\ell^{j,k,s} x_\ell^{j,k,s} \ge b_\ell^0, \qquad \forall \ell = 1, \ldots, N, \tag{53c}$$

$$\tau_\ell^0 \le \sum_{j=1}^{J} \sum_{k=1}^{K_\ell^j} \tau_\ell^{p(j),q(k)} \bar{z}_\ell^{j,k} \le \sum_{p=1}^{P} \sum_{q=1}^{Q_\ell^p} \tau_\ell^{p,q} z_\ell^{p,q}, \ \forall \ell = 1, \ldots, N, \tag{53d}$$

$$\tau^0 \le \sum_{j=1}^{J} \tau^{p(j)} \bar{y}^j \le \sum_{p=1}^{P} \tau^p y^p, \tag{53e}$$

$$\sum_{p=1}^{P} y^p = 1, \tag{53f}$$

$$\sum_{q=1}^{Q_\ell^p} z_\ell^{p,q} = y^p, \qquad \forall \ell = 1, \ldots, N, \ p = 1 \ldots, P, \tag{53g}$$

$$\sum_{j=1}^{J} \bar{y}^j = 1, \tag{53h}$$

$$\sum_{k=1}^{K_\ell^j} \bar{z}_\ell^{j,k} = \bar{y}^j, \qquad \forall \ell = 1, \ldots, N, \ j = 1 \ldots, J, \tag{53i}$$

$$\sum_{s=1}^{S_\ell^{j,k}} x_\ell^{j,k,s} = \bar{z}_\ell^{j,k}, \qquad \forall \ell = 1, \ldots, N, \ j = 1 \ldots, J,$$

$$k = 1, \ldots, K_\ell^j, \tag{53j}$$

$$y^p \in \{0,1\}, \quad \forall p = 1, \ldots, P, \tag{53k}$$

$$z_\ell^{p,q} \in \{0,1\}, \ \forall \ell = 1, \ldots, N, \ p = 1, \ldots, P, \ q = 1, \ldots, Q_\ell^p, \tag{53l}$$

$$x_\ell^{j,k,s} \in \{0,1\}, \forall \ell = 1, \ldots, N, \ j = 1, \ldots, J, \ k = 1, \ldots, K_\ell^j, \ s = 1, \ldots, S_\ell^{j,k}, \tag{53m}$$

$$\bar{y}^j \in \{0,1\}, \quad \forall j = 1, \ldots, J, \tag{53n}$$

$$\bar{z}_\ell^{j,k} \in \{0,1\}, \ \forall \ell = 1, \ldots, N, \ j = 1, \ldots, J, \ k = 1, \ldots, K_\ell^j. \tag{53o}$$

Constraint (53f) ensures exactly one system test time τ is selected, and Constraint (53g) states that all subsystems have the same system testing time τ. Constraint (53h) ensures exactly one system test time $\bar{\tau}$ is chosen, Constraint (53i) implies that the same system testing time $\bar{\tau}$ is chosen for all subsystems, and Constraint (53j) suggests variables $x_\ell^{j,k,s}$ and $\bar{z}_\ell^{j,k}$ choose the same subsystem testing time $\bar{\tau}_\ell$. By applying a natural logarithm to Objective (53a), the equivalent integer linear model is equal to

$$\begin{aligned} w^*(\mathbf{r}) = \min \ & \sum_{p=1}^{P} r^p y^p + \sum_{\ell=1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{Q_\ell^p} r_\ell^{p,q} z_\ell^{p,q} + \sum_{\ell=1}^{N}\sum_{j=1}^{J}\sum_{k=1}^{K_\ell^j}\sum_{s=1}^{S_\ell^{j,k}} r_\ell^{j,k,s} x_\ell^{j,k,s}, \\ & \text{s.t. Constraints (53b)–(53o)}, \end{aligned} \tag{54}$$

in which $R^{\mathrm{MD}}(\mathbf{r}) = e^{w^*(\mathbf{r})}$. Model (54) is a linear model, so we can use a mixed integer linear programming solver to solve it. We show that Model (53) provides a lower and upper bound on system reliability in Theorems 7 and 8, respectively. We refer to the lower bound and upper bound models as LB-MD and UB-MD, respectively. We define τ^{P+1} and $\tau^{p(J+1)}$ as $\tau^{\max} \equiv (b - \sum_{\ell=1}^{N} b_\ell^0 - \sum_{\ell=1}^{N} c_\ell \tau_\ell^0)/c$, τ_ℓ^{p,Q_ℓ^p+1} and $\tau_\ell^{p(j),q(K_\ell^j+1)}$ as $\tau_\ell^{\max} \equiv (b - c\tau^0 - \sum_{\ell'=1}^{N} b_{\ell'}^0 - \sum_{\ell'=1}^{N} c_{\ell'}\tau_{\ell'}^0 + c_\ell \tau_\ell^0)/c_\ell$, and $b_\ell^{j,k,S_\ell^{j,k}+1}$ as $b_\ell^{\max} \equiv b - c\tau^0 - \sum_{\ell'=1}^{N} b_{\ell'}^0 + b_\ell^0 - \sum_{\ell'=1}^{N} c_{\ell'}\tau_{\ell'}^0$.

Theorem 7. Define $\mathbf{r}$ by

$$r^p = R(\tau^p, T), \qquad \forall p = 1,\ldots,P, \tag{55a}$$

$$r_\ell^{p,q} = R_\ell(\tau_\ell^{p,q} + L_\ell \tau^p, T), \qquad \forall \ell = 1,\ldots,N,\ p = 1,\ldots,P,\ q = 1,\ldots,Q_\ell^p, \tag{55b}$$

$$r_\ell^{j,k,s} = R^{\mathrm{S}}(b_\ell^{j,k,s}, \tau_\ell^{p(j),q(k)}, \tau^{p(j)}, T), \qquad \forall \ell = 1,\ldots,N,\ j = 1,\ldots,J,\ k = 1,\ldots,K_\ell^j,\ s = 1,\ldots,S_\ell^{j,k}. \tag{55c}$$

If $\bar{\mathbf{r}} \le \mathbf{r}$, then $R^{\mathrm{MD}}(\bar{\mathbf{r}}) \le R^*$.

Proof. By considering $\mathbf{r}$, any feasible solution $(y, z, x, \bar{y}, \bar{z})$ in Model (53) corresponds to a solution as $\tau = \sum_{p=1}^{P} \tau^p y^p$, $\tau_\ell = \sum_{p=1}^{P}\sum_{q=1}^{Q_\ell^p} \tau_\ell^{p,q} z_\ell^{p,q}$, $b_\ell = \sum_{j=1}^{J}\sum_{k=1}^{K_\ell^j}\sum_{s=1}^{S_\ell^{j,k}} b_\ell^{j,k,s} x_\ell^{j,k,s}$, $\bar{\tau} = \sum_{j=1}^{J} \tau^{p(j)} \bar{y}^j$, and

$\tau_\ell = \sum_{j=1}^{J} \sum_{k=1}^{K_\ell^j} \tau_\ell^{p(j),q(k)} \bar{z}_\ell^{j,k}$ in Model (50) with the objective equal to

$$R(\tau, T) \prod_{\ell=1}^{N} (R_\ell(\tau_\ell, T) R^{\mathrm{S}}(b_\ell, \bar{\tau}_\ell, \bar{\tau}, T)). \tag{56}$$

If $\bar{\mathbf{r}} \leq \mathbf{r}$, the results hold since $R^{\mathrm{MD}}(.)$ is nondecreasing in $\mathbf{r}$. □

Theorem 8. Define $\mathbf{r}$ by

$$\begin{aligned}
r^p =& R(\tau^{p+1}, T), && \forall p = 1, \ldots, P, && (57a)\\
r_\ell^{p,q} =& R_\ell(\tau_\ell^{p,q+1} + L_\ell \tau^{p+1}, T), && \forall \ell = 1, \ldots, N,\ p = 1, \ldots, P,\ q = 1, \ldots, Q_\ell^p, && (57b)\\
r_\ell^{j,k,s} =& R^{\mathrm{S}}(b_\ell^{j,k,s+1}, \tau_\ell^{p(j),q(k+1)}, \tau^{p(j+1)}, T), && \forall \ell = 1, \ldots, N,\ j = 1, \ldots, J, \\
& && k = 1, \ldots, K_\ell^j,\ s = 1, \ldots, S_\ell^{j,k}. && (57c)
\end{aligned}$$

If $\bar{\bar{\mathbf{r}}} \geq \mathbf{r}$, then $R^{\mathrm{MD}}(\bar{\bar{\mathbf{r}}}) \geq R^*$.

Proof. Define $(\tau^*, \tau_\ell^*, \bar{\tau}^*, \bar{\tau}_\ell^*, b_\ell^*)$ for $\ell = 1, \ldots, N$ as an optimal solution for Model (50) with objective R^*. Find the indices $(\hat{P}, \hat{Q}_\ell, \hat{J}, \hat{K}_\ell, \hat{S}_\ell)$ for all $\ell = 1, \ldots, N$ such that $\tau^{\hat{P}} \leq \tau^* < \tau^{\hat{P}+1}$, $\tau_\ell^{\hat{P},\hat{Q}_\ell} \leq \tau_\ell^* < \tau_\ell^{\hat{P},\hat{Q}_\ell+1}$, $\tau^{p(\hat{J})} \leq \bar{\tau}^* < \tau^{p(\hat{J}+1)}$, $\tau_\ell^{p(\hat{J}),q(\hat{K}_\ell)} \leq \bar{\tau}_\ell^* < \tau_\ell^{p(\hat{J}),q(\hat{K}_\ell+1)}$, and $b_\ell^{\hat{J},\hat{K}_\ell,\hat{S}_\ell} \leq b_\ell^* < b_\ell^{\hat{J},\hat{K}_\ell,\hat{S}_\ell+1}$, and set

$$\hat{R} \equiv R\left(\tau^{\hat{P}+1}, T\right) \prod_{\ell=1}^{N} \left(R_\ell\left(\tau_\ell^{\hat{P},\hat{Q}_\ell+1}, T\right) R^{\mathrm{S}}(b_\ell^{\hat{J},\hat{K}_\ell,\hat{S}_\ell+1}, \tau_\ell^{p(\hat{J}),q(\hat{K}_\ell+1)}, \tau^{p(\hat{J}+1)}, T\right)\right). \tag{58}$$

The solution $(\hat{P}, \hat{Q}_\ell, \hat{J}, \hat{K}_\ell, \hat{S}_\ell)$ is feasible to UB-MD with objective $\hat{R}$. However, $\hat{R}$ is not less than R^* since $R(\tau, T)$ and $R_\ell(\tau_\ell, T)$ are respectively nondecreasing in τ and τ_ℓ, and $R^{\mathrm{S}}(b_\ell, \bar{\tau}_\ell, \bar{\tau}, T)$ is nondecreasing in b_ℓ, $\bar{\tau}_\ell$ and $\bar{\tau}$. □

A statement of the master problem is described as follows. Let J_{int}, K_{int}, S_{int}, $P_{\mathrm{int}} \equiv H \times J_{\mathrm{int}}$, and $Q_{\mathrm{int}} \equiv H \times K_{\mathrm{int}}$ define the initial number of points for J, K_ℓ^j, $S_\ell^{j,k}$, P, and Q_ℓ^p, respectively. Initially, set $p(j) \equiv \{1, 1+H, 1+2H, \ldots, 1+(J_{\mathrm{int}}-1)H\}$ and $q(k) \equiv \{1, 1+H, \ldots, 1+(K_{\mathrm{int}}-1)H\}$. Define an initial discretization of τ as τ^p, $p = 1, \ldots, P$; τ_ℓ as $\tau_\ell^{p,q}$, $\ell = 1, \ldots, N$, $p = 1, \ldots, P$, $q = 1, \ldots, Q_\ell^p$;

$\bar{\tau}$ as $\tau^{p(j)},\ j=1,\ldots,J$; $\bar{\tau}_\ell$ as $\tau_\ell^{p(j),q(k)},\ \ell=1,\ldots,N,\ j=1,\ldots,J,\ k=1,\ldots,K_\ell^j$; and b_ℓ as $b_\ell^{j,k,s},\ \ell=1,\ldots,N,\ j=1,\ldots,J,\ k=1,\ldots,K_\ell^j,\ s=1,\ldots,S_\ell^{j,k}$. Solve the master problem twice, once using the coefficients r^p, $r_\ell^{p,q}$ and $r_\ell^{j,k,s}$ as defined in Theorem 7 and once as defined in Theorem 8. (Note: Both of these problems require utilizing constants $r_\ell^{j,k,s}$ that are calculated via solving the subproblem, Model(49). Similar to Section 2.3.2, this may done at initialization for each $\ell=1,\ldots,N,\ j=1,\ldots,J_{\text{int}},\ k=1,\ldots,K_{\text{int}},\ s=1,\ldots,S_{\text{int}}$ and then stored in memory for usage in future iterations.) By Theorems 7 and 8, this yields lower and upper bounds on R^*. As in Section 2.3.2, these bounds can be improved by refining the discretization locally near the upper-bound model's optimal solution. We accomplish this by applying one of the following three cases. In the following, define $(J^*,K_\ell^*,S_\ell^*),\ \ell=1,\ldots,N$ as the optimal solution of UB-MD.

Case 1: For each $\ell=1,\ldots,N$, define a new budget as

$$b_\ell^{J^*,K_\ell^*,S_{\text{new}}}=0.5\left(b_\ell^{J^*,K_\ell^*,S_\ell^*}+b_\ell^{J^*,K_\ell^*,S_\ell^*+1}\right). \tag{59}$$

Then solve subproblems (49) via the D&R algorithm to obtain constants $r_\ell^{J^*,K_\ell^*,S_{\text{new}}}$ for the new point $(J^*,K_\ell^*,S_{\text{new}})$ in the lower and upper bound models.

Case 2: For each $\ell=1,\ldots,N$, define a new subsystem testing time as

$$\tau_\ell^{p(J^*),q(K_{\text{new}})}=0.5\left(\tau_\ell^{p(J^*),q(K_\ell^*)}+\tau_\ell^{p(J^*),q(K_\ell^*+1)}\right). \tag{60}$$

For the new subsystem testing time, define $b_\ell^{J^*,K_{\text{new}},s},\ s=1,\ldots,S_{\text{int}}$ evenly spaced between b_ℓ^0 and $b_\ell^{\max}$. Then, solve subproblems (49) via the D&R algorithm to obtain constants $r_\ell^{J^*,K_{\text{new}},s}$ for the new points $(J^*,K_{\text{new}},s),\ s=1,\ldots,S_{\text{int}}$ in the lower and upper bound models. Sort the subsystem testing times, and set $Q_\ell^{J^*}\leftarrow H\times K_\ell^{J^*}$ and $q(k)\in\{1,1+H,\ldots,1+(K_\ell^{J^*}-1)H\}$.

Case 3: Define a new system testing time as $\tau^{p(J_{\text{new}})}=0.5\left(\tau^{p(J^*)}+\tau^{p(J^*+1)}\right)$. For the new system testing time, define $\tau_\ell^{p(J_{\text{new}}),q(k)},\ k=1,\ldots,K_{\text{int}}$ for each subsystem $\ell=1,\ldots,N$ evenly spaced between τ_ℓ^0 and $\tau_\ell^{\max}$. Then, for each new subsystem testing time $\tau_\ell^{p(J_{\text{new}}),q(k)}$, define

$b_\ell^{J_{\text{new}},k,s},\ s=1,\ldots,S_{\text{int}}$ evenly spaced between b_ℓ^0 and $b_\ell^{\max}$, and solve subproblems (49) via the D&R algorithm to obtain constants $r_\ell^{J_{\text{new}},k,s}$ for the new points $(J_{\text{new}},k,s),\ \ell=1,\ldots,N,\ k=1,\ldots,K_{\text{int}},\ s=1,\ldots,S_{\text{int}}$ in the lower and upper bound models. Sort the subsystem testing times, and set $P \leftarrow H \times J$ and $Q_\ell^j \leftarrow H \times K_\ell^j,\ j=1,\ldots,J$. Then, set $p(j) \in \{1,1+H,\ldots,1+(J-1)H\}$ and $q(k) \in \{1,1+H,\ldots,1+(K_\ell^j-1)H\},\ j=1,\ldots,J$.

In our implementation, we randomly select one of the above cases to apply in each iteration and define parameters H_1,H_2,H_3 to represent the probability that each case is selected.

For the sake of brevity, we do not present the formal proof for the convergence of master problem algorithm; however, the arguments similar to Lemmas 7–9 can be used to prove the convergence.

2.5 Computational Results

In this section, we present and analyze computational results. We provide an in-depth analysis of a numerical example and then compare performance of D&RD with available heuristics. In addition, we apply our solution approaches to a set of instances to examine the effect of redundancy on the effectiveness of an optimal test strategy. Finally, we consider an example to study the multi-level testing. We use a server with an Intel core i12 with 2.9 GHz and 96 GB RAM for all of our solution approaches. The algorithms are coded in C++, and CPLEX 12.4 is used to solve all integer programming models.

2.5.1 Numerical Example

We consider a numerical example for component-level testing Model (3). The numerical example is a system with two subsystems. The first subsystem includes two designs, and the second subsystem has one design. We solve the model for scenarios $s=1,\ldots,30$, where the available budget in scenario s is equal to $36s$. Table 1 contains the parameters for the numerical example, in which we set $c_{\ell,i}=1$ for all $\ell=1,\ldots,N,\ i=1,\ldots,M_\ell$.

Table 1: Parameters for numerical example

$\lambda_{1,1}$	$\beta_{1,1}$	$V_{1,1}$	$\lambda_{1,2}$	$\beta_{1,2}$	$V_{1,2}$	$\lambda_{2,1}$	$\beta_{2,1}$	$V_{2,1}$
0.00005	0.8	1	0.0008	0.7	1	0.00003	0.7	2

For all scenarios, we choose $\varepsilon_{\text{sub}} = 2 \times 10^{-5}$, $P_{\text{sub}} = 20,000$, $P_{\text{int}} = 3$ and $\varepsilon = 10^{-7}$, and consider solving time limit equal to $3,600$ seconds. Figure 3 summarizes the testing time allocated to each of the three designs in each scenario. In the first subsystem, the first design has a smaller λ-value and larger β-value in comparison to the second subsystem; therefore, when the available budget b is small, it needs more testing than the second design to reach a particular reliability, so its τ-value is larger. On the other hand, when the available budget b is large, the first design needs less testing than the second design to reach a particular reliability, so its τ-value is smaller, e.g., $\tau_{1,1} = 23.49$ and $\tau_{1,2} = 7.82$ for $b = 36$ while $\tau_{1,1} = 489.84$ and $\tau_{1,2} = 518.66$ for $b = 1080$. The design of the second subsystem has smaller λ-value and lesser-or-equal β-value as compared to the designs of first subsystem, so its τ-value is smaller in every scenario.

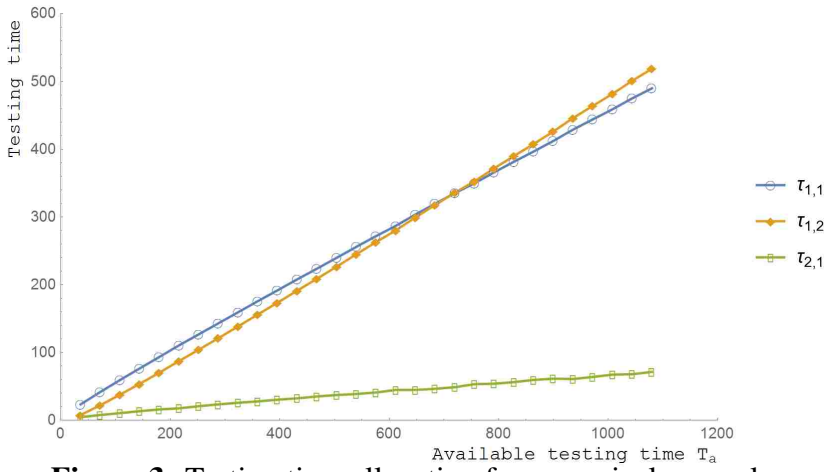


Figure 3: Testing time allocation for numerical example

2.5.2 Algorithm Performance

In order to compare algorithm performance, we consider 16 instances. A time limit of $3,600$ seconds is used for all methods, both exact and heuristic, and we implement heuristic methods using parameter values taken from the reference papers. We provide some specifics about these heuristics below.

We develop a simulated annealing (SA) heuristic to solve the problem. In this heuristic, if a candidate solution improves the reliability, then the candidate solution will be accepted. Otherwise, the inferior candidate solution is accepted using a probability of acceptance that is modeled by an annealing schedule, defined by parameters $T_1 > 0$ and $0 < \alpha \leq 1$. In iteration $k = 1, 2, \ldots$, the temperature is defined by $T_k = \alpha T_{k-1}$, and a non-improving solution is accepted with probability $e^{(R_k - R)/T_k}$. The simulated annealing (SA) heuristic uses an initial temperature of $3,000$, and $\alpha = 0.95$. We define our candidate neighborhood of solutions as follows. The initial values of $\tau_{\ell,i}$ are selected randomly between 0 and 20. In each iteration, a set of 20 candidate solutions, called a neighborhood, is created. Candidate solutions are created by modifying some $\tau_{\ell,i}$. For each candidate solution, each $\tau_{\ell,i}$ is modified with the probability of 50%. For each τ-value that is to be modified, the new τ-value is obtained from the old τ-value by adding a random number between -1 and 1.

The genetic algorithm (GA) tested is intended to replicate the GA used in [18], using a set of randomly created initial chromosomes—each of which includes a testing time for each design. An elitist crossover schema—in which parents are selected according to their objective function value—is used, and mutation is carried out via perturbing the designs' test times. Crossover/mutation probabilities of 0.9/0.1 are used in this approach.

We also implement the genetic algorithm (referred to as GENITOR) utilized by Levitin [36]. As in [36], we create $N_P = 50$ initial chromosomes randomly and perform crossover via a two-point approach and perform mutation by swapping two designs' test times. In the selection procedure, the new chromosome, acquired by crossover or mutation, is compared with the chromosome having the worst objective function in the population, and if its objective function is better than the worst chromosome in the population, it joins the population; otherwise, it is disregarded. Moreover, if crossover and mutation create redundancies in the population, all redundant solutions are deleted. In each iteration, $N_{\text{rep}} = 2,000$ new chromosomes are produced—adding randomly generated chromosomes as necessary to replenish the population—and feasibility of these chromosomes is ensured by normalizing testing times (so that the sum of costs is equal to b).

The 16 instances tested include each combination of 2, 3, 4 or 5 subsystems with 2, 3, 4, or 5 designs in each subsystem. We now describe the instance generation procedure. For all instances, we consider $V_{\ell,i} = 1$. Each design's β-value is chosen with equal probability from values 0.65, 0.7, 0.75, 0.8, and 0.85. We define initial testing times $\tau^0_{\ell,i} = 1,\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, and set the cost $c_{\ell,i} = 1,\ \ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$. We select $b = 720$ and $T = 8{,}760$ for all instances, and each design's λ-value is generated uniformly within the intervals described in Table 2. We implement the D&RD algorithm on each instance (using $\varepsilon = 10^{-7}$, $P_{\text{int}} = 3$ and a time limit of $3{,}600$ seconds) and compare to each of the above heuristics described above. The values for P_{sub} and ε_{sub} for D&RD are given in Table 2.

Table 2: Parameters for instances studying algorithm performance

Inst.	Number of subsystems	Number of designs in each subsystem	λ range	P_{sub}	ε_{sub}
1	2	2	[0.00001, 0.0001]	20,000	2×10^{-6}
2	2	3	[0.00005, 0.00025]	20,000	2×10^{-6}
3	2	4	[0.0001, 0.00025]	20,000	2×10^{-6}
4	2	5	[0.0001, 0.0005]	15,000	5×10^{-6}
5	3	2	[0.00001, 0.0001]	20,000	2×10^{-6}
6	3	3	[0.00005, 0.00025]	20,000	3×10^{-6}
7	3	4	[0.0001, 0.00025]	15,000	6×10^{-6}
8	3	5	[0.0001, 0.0005]	15,000	2×10^{-5}
9	4	2	[0.00001, 0.0001]	20,000	3×10^{-6}
10	4	3	[0.00005, 0.00015]	20,000	3×10^{-6}
11	4	4	[0.0001, 0.00025]	20,000	6×10^{-6}
12	4	5	[0.0001, 0.00025]	15,000	3×10^{-5}
13	5	2	[0.00001, 0.0001]	20,000	2×10^{-6}
14	5	3	[0.00005, 0.00025]	20,000	6×10^{-6}
15	5	4	[0.00005, 0.00025]	15,000	2×10^{-5}
16	5	5	[0.0001, 0.00025]	20,000	4×10^{-5}

The results are summarized in Table 3. In Table 3, LB and UB denote the lower and upper bounds of reliability acquired by the D&RD algorithm. According to Table 3, D&RD provides tight bounds even for the largest instances and produces greater lower bounds than both GA and GENITOR; however, SA produces greater lower bounds than our exact algorithm. A follow-on set of experiments revealed that, in all 16 instances, SA generates a solution within 10^{-4} of optimality in less than a second, thus suggesting that SA is likely to identify high-quality solutions quickly.

Of course, SA provides no upper bound on the optimal objective value and hence comes with no guarantee of optimality.

Table 3: Computational comparison

Inst.	LB	UB	Gap	SA	GA	GENITOR
1	0.96877657	0.96877793	1.362×10^{-6}	0.96877676	0.96866712	0.96877604
2	0.97667526	0.97667781	2.553×10^{-6}	0.97667554	0.97554303	0.97667523
3	0.98319327	0.98319840	5.128×10^{-6}	0.98319333	0.98226540	0.98319137
4	0.96506727	0.96508031	1.304×10^{-5}	0.96506740	0.96227264	0.96502206
5	0.97794473	0.97794519	4.619×10^{-7}	0.97794482	0.97754561	0.97792183
6	0.96164354	0.96164859	5.047×10^{-6}	0.96164376	0.95898590	0.96156142
7	0.96578494	0.96579538	1.044×10^{-5}	0.96578534	0.95996997	0.96573422
8	0.91354268	0.91357048	2.780×10^{-5}	0.91354334	0.90068946	0.91352284
9	0.91305039	0.91305297	2.579×10^{-6}	0.91305073	0.90827255	0.91299274
10	0.96202986	0.96203675	6.885×10^{-6}	0.96203076	0.95906679	0.96200378
11	0.96815461	0.96817421	1.960×10^{-5}	0.96815492	0.96221063	0.96814310
12	0.94774059	0.94776901	2.842×10^{-5}	0.94774131	0.93516419	0.94769670
13	0.94448573	0.94448865	2.924×10^{-6}	0.94448602	0.94209628	0.94445546
14	0.94597493	0.94599256	1.763×10^{-5}	0.94597599	0.94089367	0.94596832
15	0.93575476	0.93578183	2.707×10^{-5}	0.93575523	0.92757697	0.93574860
16	0.96990016	0.96992689	2.673×10^{-5}	0.96990070	0.96015332	0.96989514

2.5.3 Testing Components versus Installing Redundancies

We now consider 5 instances—solving each over 5 potential values of b—in order to demonstrate the effects of the component redundancy versus allocating more testing time. As depicted in Table 4, each instance consists of a system with 5 subsystems, and all of components in a given subsystem are identical (i.e., there is one design in each subsystem). For instance $i = 1,\ldots,5$, we define $M_\ell = 1$ and $V_{\ell,1} = i+2$ for all $\ell = 1,\ldots,5$. We choose $T = 8,760$, $c_{\ell,i} = 1$, $\ell = 1,\ldots,N$, $i = 1,\ldots,M_\ell$, and the parameters of the instances are depicted in Table 4. The time limit is equal to $3,600$ seconds and we consider $\varepsilon = 10^{-7}$. In these instances, we use $P_{\text{int}} = 3$, $P_{\text{sub}} = 50,000$ and $\varepsilon_{\text{sub}} = 5 \times 10^{-7}$.

The results from these instances are summarized in Table 5. All instances are solved by the D&RD method. Table 5 summarizes the lower bound reliabilities for all instances. Although we do not display the upper bounds, the optimality gap for each instance is no more than 1.8×10^{-6}.

Table 4: Parameters for experiments on varying redundancy levels

Inst.	$\lambda_{1,i}$	$\beta_{1,i}$	$\lambda_{2,i}$	$\beta_{2,i}$	$\lambda_{3,i}$	$\beta_{3,i}$	$\lambda_{4,i}$	$\beta_{4,i}$	$\lambda_{5,i}$	$\beta_{5,i}$
1	0.00025	0.75	0.00011	0.65	0.00013	0.6	0.0001	0.85	0.00012	0.75
2	0.00025	0.75	0.00011	0.65	0.00013	0.6	0.0001	0.85	0.00012	0.75
3	0.00025	0.75	0.00011	0.65	0.00013	0.6	0.0001	0.85	0.00012	0.75
4	0.00025	0.75	0.00011	0.65	0.00013	0.6	0.0001	0.85	0.00012	0.75
5	0.00025	0.75	0.00011	0.65	0.00013	0.6	0.0001	0.85	0.00012	0.75

In each instance, because the components in a subsystem are identical, only one of the identical components is tested. When fewer parallel components exist in each subsystem, the system reliability is lesser even for a large amount of available testing time, i.e, the system reliability is a small number for instance 1 where there only exist 3 components in each subsystem. However, when there are enough redundant components in each subsystem, the effect of increasing available testing time on system reliability is much greater than increasing redundancy in subsystems. In instance 4, for example, if the available b is increased from 120 to 360, it has a greater effect on reliability than adding one component to each subsystem.

Table 5: Results for different redundancy levels

Inst.	$b = 120$	$b = 240$	$b = 360$	$b = 480$	$b = 600$
1	0.81359498	0.87068118	0.89620755	0.91137826	0.92167343
2	0.92587899	0.95498267	0.96654302	0.97293813	0.97705911
3	0.97085468	0.98440488	0.98923439	0.99173492	0.99327060
4	0.98852868	0.99457886	0.99652040	0.99746298	0.99801527
5	0.99546800	0.99810700	0.99887004	0.99921747	0.99941166

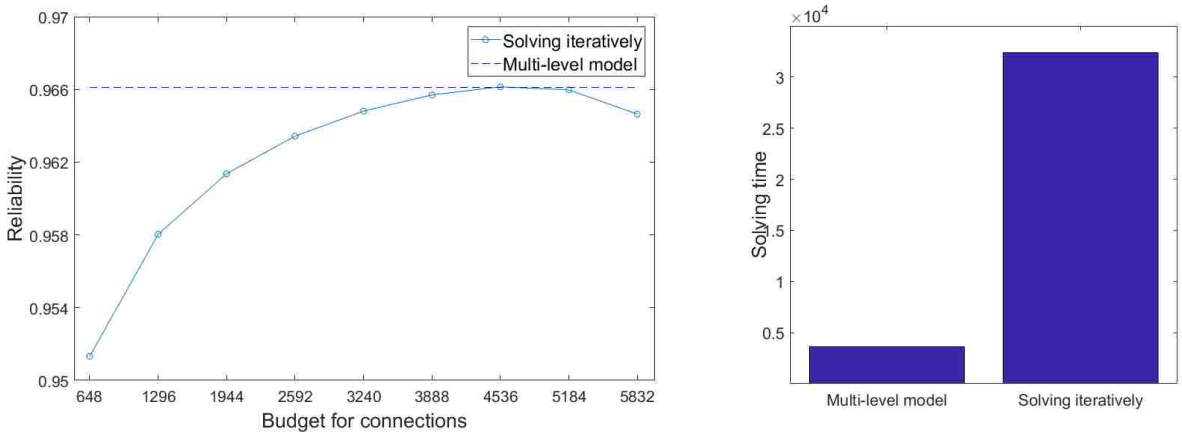
2.5.4 Multi-Level Testing versus Component-Level Testing

We consider 10 instances to study the effectiveness of our multi-level model in comparison with an approach that partitions the budget for testing component designs and connections. The instances consider a system with $N = 3$, $M_\ell = 3$ and $V_{\ell,i} = 1$ for all $\ell = 1,\ldots,3,\ i = 1,\ldots,3$. Using a budget of $b = 6,480$, we solve Model (47) and compare solutions to the multi-level model against single-level (i.e., component design testing only) models that were available prior to this research. Specifically, we partition the budget into two parts, with one portion dedicated to testing assembled subsystems and systems and the other portion dedicated to testing component designs. We then

solve the single-level twice in sequence, once to allocate testing times to subsystem- and system-level assemblies in order to maximize the reliability due only to connections. (Note: the subsystem- and system-connections collectively form a series system, so this can be accomplished by using the convex optimization model of Coit [15]. We solve this convex optimization problem using the CVX solver in Matlab.) We repeat this for a sequence of instances in which we vary the budget partition in each instance: For instances $j = 1,\ldots,9$, we assume there is a budget of $648j$ dedicated to testing at the subsystem- and system-levels, with the remaining budget dedicated to testing component designs. The AMSAA parameters, cost data, and mission length for these instances are given in Table 6, and we consider $\tau^0_{\ell,i} = \tau^0_\ell = \tau^0 = 1$, $J_{\ell,i} = L_{\ell,i} = L_\ell = 0.1$ for all $\ell = 1,\ldots,N,\ i = 1,\ldots,M_\ell$, $H_1 = 0.65$, $H_2 = 0.3$, and $H_3 = 0.05$. We employ a time limit of 3600 seconds and $\varepsilon = 1 \times 10^{-7}$ in the discretize-and-refine method for both component-level and multi-level testing models. We set $J_{\text{int}} = 4$, $K_{\text{int}} = 4$, $S_{\text{int}} = 100$, and $H = 200$. By defining ε_{sub} and P_{sub} as the subsystem tolerance and initial number of points, consider $\varepsilon_{\text{sub}} = 1 \times 10^{-4}$ and set $P_{\text{sub}} = 50{,}000$ for instances $j = 1,\ldots,9$ and $P_{\text{sub}} = 100$ for the multi-level testing instance. For the multi-level testing instance, the lower bound, upper bound and optimal gap equal 0.966111, 0.966628 and 5.17×10^{-4}, respectively. The lower bound comparison of reliabilities are shown in Figure 4(a), and we show the computational times in Figure 4(b). Although we do not show the upper bound reliabilities for instances $j = 1,\ldots,9$, the optimal gap is less than 2×10^{-4} for these instances. The results suggest that if we allocate small or large budget for testing connections, the system reliability is a small number, e.g., the system reliability, when the budget for testing connections equals 648, is 0.951322. One could employ a search over the budget partition to improve the solution available from the single-level models, and indeed, this solution might do fairly well; however, finding such budget needs higher solving time as depicted in Figure 4(b).

Table 6: Parameters for the multi-level testing instances

ℓ	$\lambda_{\ell,1}$	$\lambda_{\ell,2}$	$\lambda_{\ell,3}$	λ_ℓ	$\beta_{\ell,1}$	$\beta_{\ell,2}$	$\beta_{\ell,3}$	β_ℓ	$c_{\ell,1}$	$c_{\ell,2}$	$c_{\ell,3}$
1	0.00008	0.000095	0.000085	0.000004	0.7	0.85	0.75	0.8	0.5	0.8	0.4
2	0.000085	0.00009	0.000095	0.0000025	0.75	0.7	0.8	0.65	0.3	0.7	0.5
3	0.00009	0.000075	0.00008	0.000003	0.65	0.75	0.7	0.85	0.3	0.6	0.2
$c_1 = 5$		$c_2 = 7$		$c_3 = 4$	$\lambda = 0.0000055$			$\beta = 0.7$		$c = 20$	$T = 8{,}760$



(a) **(b)**

Figure 4: Results for instances comparing multi-level model with solving the component-level testing allocation and connection testing allocation models iteratively

2.6 Conclusion and Future Research

This study considers the allocation of test times across the components of a series-parallel system. We also propose a model that takes account the possibility of testing at subsystem and system levels. An exact algorithm is developed to solve the problem, and we prove that the algorithm is finite. We perform a detailed study of a numerical example and compare our algorithm computationally with three heuristics from literature. Computational results indicate that the D&RD algorithm provides a tight optimality gap on solutions for instances up to size 5 subsystems with 5 component designs in each subsystem. This method is also effective in finding lower bounds on system reliability, and it quantifies (for the first time in the literature) the quality—with respect to an optimality gap criterion—of existing heuristic solution approaches.

Future work may seek to model and optimally solve the problem when component are connected in a cold standby redundancy structures. Additionally, a follow-on investigation may study the problem when the failures of components are dependent. A subsequent possible investigation may consider the relationship of the reliability growth testing strategies with product warranty strategies and/or maintenance strategies. Future research may also consider this problem for objective function of maximizing a lower percentile of system time to failure. We also assume that the system has a fixed mission length; therefore, future research may seek to model the problem

when the system has an indeterminate mission length.

3 Robust Allocation of Testing Resources in Reliability Growth

3.1 Introduction

In the previous chapter, we address the case where the relationship between component testing and component reliability is known (deterministically), but there is a lack of work addressing what happens when this relationship is subject to uncertainty. This may be problematic because reliability growth models are built upon preliminary failure data that is subject to randomness. In this chapter, we consider a robust allocation of testing resources across the components of the series/series-parallel system to increase system reliability in this environment. We assume that components exhibit reliability growth according to the AMSAA model.

With a few exceptions that we now summarize, all of the resource allocation papers in reliability growth assume known AMSAA parameters, thus yielding deterministic optimization models. However, reliability growth testing is typically applied to new systems with unproven designs [15], and for these systems, there is insufficient data to estimate the AMSAA model parameters with high accuracy, thus introducing uncertainty into the resource allocation problem. Awad [2] and Coit [15] address this limitation to allocate testing times in a series system when each component's failure rate is normally distributed having a mean given by an AMSAA model with known parameters. By contrast, our robust optimization approach needs not assume any underlying probability distribution. Moreover, our method can be extended to consider the allocation of testing times, for the first time in the literature, in a series-parallel system when component reliability growth has AMSAA model structure with unknown parameters.

Robust optimization seeks to generate a solution in which its objective value or its feasibility must be guaranteed for any realization of its parameters within a bounded uncertainty set. Robust linear optimization was first examined by Soyster [50] and later extended (by Ben-Tal and Nemirovski [4] and El-Ghaoui et al. [20]) to derive efficient algorithms for robust convex optimization problem. Bertsimas and Sim [9] characterize a class of robust discrete optimization problems that remain efficiently solvable. Robust optimization has variety of application in decision making en-

vironments such as linear programming, assignment problem, shortest paths, minimum spanning tree, knapsack problem, resource allocation, scheduling, production planning, inventory, layout planning and network design [30]. We refer the interested reader to Ben-Tal and Nemirovski [5] and Bertsimas et al. [8] for comprehensive surveys of robust optimization theory and applications.

This chapter is organized as follows: the background and problem definition is defined in Section 3.2. We present a robust reliability growth model for a series system in Section 3.3 and for a series-parallel system in Section 3.4. Computational results are summarized in Section 3.5, and we conclude in Section 3.6.

3.2 Background and Problem Definition

Following models in [15, 18], we consider the problem of allocating limited testing times to independent components within a system. Let N denote the number of subsystems in the system, and let the N-vector of decision variables τ represent the total testing time (in hours) allocated to each subsystem. One hour of testing on subsystem $\ell = 1,\ldots,N$ is assumed to require c_ℓ^k units of resource $k = 1,\ldots,K$. The N-vector τ^0 denotes the preliminary testing time, where the preliminary testing time τ_ℓ^0 can be interpreted as the amount of testing already completed on subsystem ℓ at the time our optimization model is solved to generate a test plan for the system. This preliminary testing may be used to estimate AMSAA model parameters (and their associated uncertainty) that are used to build the optimization model.

The objective in allocating limited testing time is to maximize system reliability over a mission of length $T > 0$ hours that begins after testing. The deterministic allocation problem is generally stated as

$$\max R(\tau, T), \tag{61a}$$

$$\text{s.t. } \tau \in X \equiv \left\{ \tau \,\middle|\, \sum_{\ell=1}^{N} c_\ell^k \tau_\ell \le b_k,\ k = 1,\ldots,K;\ \tau \ge \tau^0 \right\}, \tag{61b}$$

where $R(\tau, T)$, which is defined mathematically in the following paragraph, denotes the system's

mission reliability resulting when subsystems are tested according to τ. Constraints (61b) ensure that (i) testing time for each subsystem ℓ is at least its initial testing time τ_ℓ^0 and (ii) the total resource consumption across all subsystems is no more than b_k for all resources $k = 1, \ldots, K$. We initially assume (for mathematical convenience) that $\tau_\ell^0 \geq 1$, $\forall \ell = 1, \ldots, N$ but will address (at the end of Section 3.3) recourse in the event this assumption does not hold. In what follows, we extend Model (61) to consider uncertainty in $R(\tau, T)$ via robust optimization. Before doing so, we first summarize assumptions and existing results for Model (61) that are pertinent to our research.

The objective function $R(\tau, T)$ from Model (61) is then related to the growth functions $R(\tau_\ell; T, \lambda_\ell, \beta_\ell)$ according to the system's structure, and we hereafter denote this objective as $R(\tau; \boldsymbol{\lambda}, \boldsymbol{\beta})$ to note its dependence on parameter vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$. (Boldface $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$ are used to represent the associated N-vectors of λ_ℓ- and β_ℓ-values.) Thus, if the N one-component subsystems are connected in series, then

$$R(\tau; \boldsymbol{\lambda}, \boldsymbol{\beta}) = \prod_{\ell=1}^{N} R(\tau_\ell; T, \lambda_\ell, \beta_\ell), \tag{62}$$

but if each subsystem $\ell = 1, \ldots, N$, has M_ℓ identical components that are simultaneously improved by increasing τ_ℓ, then

$$R(\tau; \boldsymbol{\lambda}, \boldsymbol{\beta}) = \prod_{\ell=1}^{N} \left[1 - (1 - R(\tau_\ell; T, \lambda_\ell, \beta_\ell))^{M_\ell} \right]. \tag{63}$$

Given the definitions above, it is now appropriate to describe the robust optimization problem considered in this chapter. Our problem differs from Model (61) because we do not assume $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$ are known with certainty. Rather, we assume $(\boldsymbol{\lambda}, \boldsymbol{\beta})$ lies within an uncertainty set, U, and consider the (robust) optimization problem

$$\max_{\tau \in X} \min_{(\boldsymbol{\lambda}, \boldsymbol{\beta}) \in U} R(\tau; \boldsymbol{\lambda}, \boldsymbol{\beta}). \tag{64}$$

Throughout the reminder of this chapter, we refer to a pair $(\boldsymbol{\lambda}, \boldsymbol{\beta}) \in U$ as a *scenario*, and we refer to $\min_{(\boldsymbol{\lambda}, \boldsymbol{\beta}) \in U} R(\tau; \boldsymbol{\lambda}, \boldsymbol{\beta})$, the reliability at $\tau \in X$ that corresponds to the *worst-case scenario*, as the *robust objective value*. Note that Model (64) maximizes the robust objective value over all

feasible test strategies.

The models throughout the remainder of this chapter utilize a so-called budgeted uncertainty set. (The interested reader may refer to [9, 10, 11] for other examples of budgeted uncertainty and [22, 23] for reliability-related robust optimization models.) We assume each parameter is restricted to fall within an interval, i.e.,

$$\lambda_\ell \in [\bar{\lambda}_\ell, \bar{\lambda}_\ell + \sigma_\ell^L],\ \ell = 1,\ldots,N, \tag{65a}$$

$$\beta_\ell \in [\bar{\beta}_\ell, \bar{\beta}_\ell + \sigma_\ell^B],\ \ell = 1,\ldots,N. \tag{65b}$$

We scale each interval by introducing new parameters γ_ℓ^B and $\gamma_\ell^L \in [0,1]$ such that $\beta_\ell = \bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B$ and $\lambda_\ell = \bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L$, in which $\bar{\lambda}_\ell,\ \sigma_\ell^L \geq 0$ and $\bar{\beta}_\ell,\ \bar{\beta}_\ell + \sigma_\ell^B$ are in $[0,1]$. Using the scaled parameters, we define the uncertainty set U_Φ for a given budget $0 \leq \Phi \leq 2N$ as

$$U_\Phi = \left\{ \left(\gamma^L, \gamma^B\right) \in \mathbb{R}^N : \sum_{\ell=1}^{N} \left(\gamma_\ell^L + \gamma_\ell^B\right) \leq \Phi, 0 \leq \gamma_\ell^L, \gamma_\ell^B \leq 1 \right\}. \tag{66}$$

As we will show in Theorem 9, the reliability function is decreasing in γ_ℓ^L and γ_ℓ^B; therefore, $\Phi = 0$ represents an optimistic view of uncertainty (i.e., all parameters will take their best-case values, which happen to be their smallest values) and $\Phi = 2N$ is pessimistic in that all parameters will take their worst-case/largest values. We restate Model (64) as

$$\max_{\tau \in X} \min_{(\gamma^L, \gamma^B) \in U_\Phi} R\left(\tau; \bar{\lambda} + \sigma^L * \gamma^L, \bar{\beta} + \sigma^B * \gamma^B\right), \tag{67}$$

where $*$ denotes element-wise multiplication of the associated N-vectors. Our reasoning for employing a single uncertainty parameter Φ (e.g., instead of having two parameters to enable separate uncertainty budgets for λ and β) is two-fold: (i) each uncertainty parameter adds an additional dimension to the set of policies that can be generated from our model, thereby increasing the complexity associated with determining how to best implement the model; and (ii) our integrality property (Theorem 10, proven in the following section), which enables efficient solution of the

robust optimization problem for series systems, fails in the case of a multi-parameter uncertainty set.

To summarize, our major assumptions are that the system structure is fixed, subsystems are independent, redundant components are in active standby, and the mission length T is fixed. Moreover, it is assumed that the failures of subsystems occur according to the AMSAA reliability growth model, in which parameters λ_ℓ and β_ℓ lie within a budget-restricted uncertainty set. Although we believe extensions of our models may lead to weakening some of these, all of these assumptions are limiting for the results given in this chapter. In addition to the assumptions listed above, we have also assumed that $\tau_\ell \geq 1,\ \ell = 1,\ldots,N$. This assumption is purely for mathematical convenience in the case of series system, and we have discussed (in the last paragraph of Section 3.3) a means of solving the problem in the event this assumption does not hold.

Before developing the mechanics to solve Model (64), we first summarize its relation to important results from the literature. Coit [15] considered a version of Model (61) in which components are connected in a series, which leads to a convex optimization problem. As we see in Section 3.3, the robust version of this problem turns out to be easily solvable as well. On the other hand, when the system is series-parallel, we propose exact algorithms for the deterministic model in Chapter 2. We extend the deterministic series-parallel model into a robust model (see Section 3.4) and demonstrate that the resulting problem can be solved via extension of the algorithm given in Chapter 2. The notation used in the remainder of the document is summarized below:

Parameters

K	Number of resources
b_k	Amount of resource $k = 1,\ldots,K$ available
c_ℓ^k	Amount of resource $k = 1,\ldots,K$ required per unit time of testing subsystem $\ell = 1,\ldots,N$
$\beta_\ell,\ \lambda_\ell$	AMSAA model parameters for subsystem $\ell = 1,\ldots,N$
$[\bar{\beta}_\ell, \bar{\beta}_\ell + \sigma_\ell^B]$	Interval for parameter β of subsystem $\ell = 1,\ldots,N$

$[\bar{\lambda}_\ell, \bar{\lambda}_\ell + \sigma_\ell^L]$	Interval for parameter λ of subsystem $\ell = 1, \ldots, N$
$\gamma_\ell^L,\ \gamma_\ell^B$	Realized coefficient for the AMSAA model parameters for subsystem $\ell = 1, \ldots, N$
Φ	Uncertainty budget ($\Phi \in [0,\ 2N]$)
U_Φ	Uncertainty set
τ_ℓ^0	Initial testing time of subsystem $\ell = 1, \ldots, N$ (assumed to be greater or equal to 1)
P_ℓ	Number of discrete points for subsystem $\ell = 1, \ldots, N$
P	Number of discrete points for each subsystem (assumed to be equal for all subsystems)
r_ℓ^p	Reliability of point p in subsystem ℓ ($\ell = 1, \ldots, N$; $p = 1, \ldots, P_\ell$)
z_ℓ^p	The value of $\gamma_\ell^L + \gamma_\ell^B$ for point p in subsystem ℓ ($\ell = 1, \ldots, N$; $p = 1, \ldots, P_\ell$)
τ_ℓ^p	Testing time for point p in subsystem ℓ ($p = 1, \ldots, P$; $\ell = 1, \ldots, N$)
$r_{\ell,j}^p$	Reliability of point p in subsystem ℓ for scenario j ($p = 1, \ldots, P$; $\ell = 1, \ldots, N$; $j = 1, \ldots, \lvert\bar{D}\rvert$)

Decision Variables

τ_ℓ	Testing time of subsystem $\ell = 1, \ldots, N$
x_ℓ^p	$\begin{cases} 1 & \text{if the value of } \gamma_\ell^L + \gamma_\ell^B \text{ in subsystem } \ell \text{ is } z_\ell^p \ \ (\ell = 1, \ldots, N;\ p = 1, \ldots, P_\ell) \\ 0 & \text{otherwise} \end{cases}$
y_ℓ^p	$\begin{cases} 1 & \text{if the testing time for subsystem } \ell \text{ is } \tau_\ell^p \ (p = 1, \ldots, P;\ \ell = 1, \ldots, N) \\ 0 & \text{otherwise} \end{cases}$

In the following section, we specialize Model (64) for the case of series system and develop a cutting-plane algorithm to solve it.

3.3 Solution Method for Series Systems

We now specialize the problem to the case of series structure. Let N define the number of subsystems (each with a single component) in the series system. Based on Equation (62), $R(\tau; \lambda, \beta)$ is

given by

$$R(\tau;\lambda,\beta) = \prod_{\ell=1}^{N} R\left(\tau_\ell; T, \bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L, \bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B\right), \tag{68}$$

in which the uncertainty set U_Φ is defined in Equation (66). For series systems, the robust optimization Model (67) can be stated as

$$R^\star = \max \;\; R, \tag{69a}$$

$$\text{s.t.} \;\; \prod_{\ell=1}^{N} e^{-(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B) \tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T} \geq R, \; \forall \left(\gamma^L, \gamma^B\right) \in U_\Phi, \tag{69b}$$

$$\tau \in X. \tag{69c}$$

Towards solving Model (69), we now propose a cutting-plane algorithm to solve Model (69) optimally. (The interested reader may refer to [6] for seminal work on cutting-plane algorithms for min-max and max-min problems.) Model (69) is a nonlinear/nonconvex model, and Constraints (69b) are uncountably infinite, so Model (69) is difficult to solve directly. By applying a logarithm to Constraint (69b), Model (69) is equivalent to

$$R^\star = \max \;\; R, \tag{70a}$$

$$\text{s.t.} \;\; \sum_{\ell=1}^{N} -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B) \tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T \geq \ln R, \; \forall \left(\gamma^L, \gamma^B\right) \in U_\Phi, \tag{70b}$$

$$\tau \in X. \tag{70c}$$

Model (70) turns out to be convex, but we are still left with the issue that (70b) is indexed over an infinite set of constraints. We hence apply a cutting-plane algorithm to solve Model (70). The separation model, used to compute the robust objective value for given $\tau \in X$, can be stated as

$$G(\tau) = \min\left\{F\left(\gamma^L, \gamma^B\right) \mid \left(\gamma^L, \gamma^B\right) \in U_\Phi\right\}, \tag{71}$$

where $f_\ell(\gamma_\ell^L, \gamma_\ell^B) \equiv -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T$ and $F(\gamma^L, \gamma^B) \equiv \sum_{\ell=1}^N f_\ell(\gamma_\ell^L, \gamma_\ell^B)$. The solution for Model (71) generates a feasible solution for Model (69) with $R = e^{G(\tau)}$; therefore, Model (71) provides a lower bound for the logarithm of the objective value of Model (69) (i.e., $e^{G(\tau)} \leq R^\star$). Model (71) is nonconvex in general, but we now demonstrate that it can be solved efficiently via a dynamic programming algorithm. We begin by establishing that $F(\gamma^L, \gamma^B)$ is monotone and *componentwise concave*, i.e., concave in either γ_ℓ^L or γ_ℓ^B whenever all other γ^L- and γ^B-variables are fixed.

Theorem 9. The function $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ is decreasing and componentwise concave.

Proof. The partial derivatives of $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ are given by

$$\frac{\partial f_\ell(\gamma_\ell^L, \gamma_\ell^B)}{\partial \gamma_\ell^L} = -\sigma_\ell^L \left(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B\right) \tau_\ell^{\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1} T, \tag{72a}$$

$$\frac{\partial f_\ell(\gamma_\ell^L, \gamma_\ell^B)}{\partial \gamma_\ell^B} = -\sigma_\ell^B \tau_\ell^{\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1} (\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L) \left[1 + (\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B) \ln(\tau_\ell)\right] T. \tag{72b}$$

Therefore, $\frac{\partial f_\ell(\gamma_\ell^L, \gamma_\ell^B)}{\partial \gamma_\ell^L} < 0$ and the assumption $\tau_\ell^0 \geq 1$ implies $\frac{\partial f_\ell(\gamma_\ell^L, \gamma_\ell^B)}{\partial \gamma_\ell^B} < 0$ in Equation (72b) as well. As a result, $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ is decreasing regarding γ_ℓ^L and γ_ℓ^B when the other is fixed. The second derivatives of $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ with respect to variables γ_ℓ^L and γ_ℓ^B are equal to

$$\frac{\partial^2 f_\ell(\gamma_\ell^L, \gamma_\ell^B)}{\partial^2 \gamma_\ell^L} = 0, \tag{73a}$$

$$\frac{\partial^2 f_\ell(\gamma_\ell^L, \gamma_\ell^B)}{\partial^2 \gamma_\ell^B} = -(\sigma_\ell^B)^2 \tau_\ell^{\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1} \left(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L\right) \ln(\tau_\ell) \left[2 + \left(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B\right) \ln(\tau_\ell)\right] T. \tag{73b}$$

Note that ${\partial^2 f_\ell(\gamma_\ell^L, \gamma_\ell^B)}/{\partial^2 \gamma_\ell^B} \leq 0$, since $\ln \tau_\ell \geq 0$. Therefore, $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ is also componentwise concave.

□

Remark 2. Because $F(\gamma^L, \gamma^B)$ is a summation of terms that all utilize a different ℓ-index, Theorem 9 also proves that the function $F(\gamma^L, \gamma^B)$ is decreasing and componentwise concave.

As we demonstrate in Theorems 12–13, the previous result allows us to guarantee existence

of an optimal solution to Model (71) among a finite set of candidate solutions. In the following counterexample, we demonstrate that $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ is not concave in general.

Counterexample 1. Define $\bar{\lambda}_\ell = 0.00001$, $\sigma_\ell^L = 0.00006$ $\bar{\beta}_\ell = 0.6$, $\sigma_\ell^B = 0.15$, $\tau_\ell = 100$, $K = 1$, $c_\ell^1 = 1$ and $b_1 = 8{,}760$. Letting $(\bar{\gamma}_\ell^L, \bar{\gamma}_\ell^B) = (1,0)$, $(\bar{\bar{\gamma}}_\ell^L, \bar{\bar{\gamma}}_\ell^B) = (0,1)$, and $\alpha = 0.5$, we have

$$f_\ell(\alpha\bar{\gamma}_\ell^L + (1-\alpha)\bar{\bar{\gamma}}_\ell^L, \alpha\bar{\gamma}_\ell^B + (1-\alpha)\bar{\bar{\gamma}}_\ell^B) = -0.05295, \tag{74a}$$

$$\alpha f_\ell(\bar{\gamma}_\ell^L, \bar{\gamma}_\ell^B) + (1-\alpha) f_\ell(\bar{\bar{\gamma}}_\ell^L, \bar{\bar{\gamma}}_\ell^B) = -0.03954. \tag{74b}$$

Because $f_\ell(\alpha\bar{\gamma}_\ell^L + (1-\alpha)\bar{\bar{\gamma}}_\ell^L, \alpha\bar{\gamma}_\ell^B + (1-\alpha)\bar{\bar{\gamma}}_\ell^B) < \alpha f_\ell(\bar{\gamma}_\ell^L, \bar{\gamma}_\ell^B) + (1-\alpha) f_\ell(\bar{\bar{\gamma}}_\ell^L, \bar{\bar{\gamma}}_\ell^B)$, the function $f_\ell(\gamma_\ell^L, \gamma_\ell^B)$ is not concave in general.

Theorem 10. There exists an optimal solution to Model (71) such that the number of subsystems ℓ with γ_ℓ^L and/or γ_ℓ^B taking on noninteger values is at most 1.

Proof. Let $(\hat{\gamma}^L, \hat{\gamma}^B)$ denote an optimal solution to Model (71). Among all such solutions, let $(\hat{\gamma}^L, \hat{\gamma}^B)$ be an optimal solution with the minimum number of subsystems ℓ such that γ_ℓ^L and/or γ_ℓ^B is fractional. (In the remainder of this proof, we refer to such a subsystem as a fractional subsystem.) Let $k^\star$ denote the number of fractional subsystems in $(\hat{\gamma}^L, \hat{\gamma}^B)$. If $k^\star = 0$ or 1, there is nothing to prove, so suppose $k^\star \geq 2$. We establish a contradiction by constructing an optimal solution to Model (71) with only $k^\star - 1$ fractional subsystems. First, let $\{\ell', \ell''\}$ denote a pair of distinct fractional subsystems with respect to $(\hat{\gamma}^L, \hat{\gamma}^B)$ in which at least one of $\{\hat{\gamma}_\ell^L, \hat{\gamma}_\ell^B\}$ is noninteger for $\ell \in \{\ell', \ell''\}$. At least one of the following three cases will hold.

Case 1: Suppose $\hat{\gamma}_{\ell'}^L$ and $\hat{\gamma}_{\ell''}^L$ are fractional. Define $z = \hat{\gamma}_{\ell'}^L + \hat{\gamma}_{\ell''}^L$. When $\gamma_\ell^L = \hat{\gamma}_\ell^L$, $\ell \in \{1, \ldots, N\} \setminus \{\ell', \ell''\}$ and $\gamma^B = \hat{\gamma}^B$ are fixed in Model (71), it reduces to the linear program

$$\min\ f_{\ell'}(\gamma_{\ell'}^L, \hat{\gamma}_{\ell'}^B) + f_{\ell''}(\gamma_{\ell''}^L, \hat{\gamma}_{\ell''}^B), \tag{75a}$$

$$\text{s.t. } \gamma_{\ell'}^L + \gamma_{\ell''}^L \leq z, \tag{75b}$$

$$0 \leq \gamma_{\ell'}^L,\ \gamma_{\ell''}^L \leq 1. \tag{75c}$$

Because Model (75) is a linear program, it has an extreme point optimal solution. Plotting the feasible region (75b)–(75c) reveals that the extreme points are $(\gamma^L_{\ell'}, \gamma^L_{\ell''}) \in \{(0,0),(0,z),(z,0)\}$ if $z \leq 1$ and $(\gamma^L_{\ell'}, \gamma^L_{\ell''}) \in \{(0,0),(0,1),(1,0),(z-1,1),(1,z-1)\}$ if $z > 1$. Noting that all extreme points have either $\gamma^L_{\ell'}$ or $\gamma^L_{\ell''}$ as an integer. Solving (75) yields a new solution $(\tilde{\gamma}^L, \tilde{\gamma}^B) \in U_\Phi$ by replacing $\hat{\gamma}^L_{\ell'}$ and $\hat{\gamma}^L_{\ell''}$ in $(\hat{\gamma}^L, \hat{\gamma}^B)$ with the obtained solution to the LP (75). For this solution, we have

$$f_\ell(\tilde{\gamma}^L_{\ell'}, \hat{\gamma}^B_{\ell'}) + f_\ell(\tilde{\gamma}^L_{\ell''}, \hat{\gamma}^B_{\ell''}) \leq f_\ell(\hat{\gamma}^L_{\ell'}, \hat{\gamma}^B_{\ell'}) + f_\ell(\hat{\gamma}^L_{\ell''}, \hat{\gamma}^B_{\ell''}), \tag{76a}$$

$$f_\ell(\tilde{\gamma}^L_{\ell'}, \hat{\gamma}^B_{\ell'}) + f_\ell(\tilde{\gamma}^L_{\ell''}, \hat{\gamma}^B_{\ell''}) + \sum_{\ell \in \{1,\ldots,N\} \setminus \{\ell', \ell''\}} f_\ell(\hat{\gamma}^L_\ell, \hat{\gamma}^B_\ell) \leq \sum_{\ell \in \{1,\ldots,N\}} f_\ell(\hat{\gamma}^L_\ell, \hat{\gamma}^B_\ell), \tag{76b}$$

$$F(\tilde{\gamma}^L, \tilde{\gamma}^B) \leq F(\hat{\gamma}^L, \hat{\gamma}^B), \tag{76c}$$

in which Equation (76a) is valid because $(\tilde{\gamma}^L_{\ell'}, \tilde{\gamma}^L_{\ell''})$ and $(\hat{\gamma}^L_{\ell'}, \hat{\gamma}^L_{\ell''})$ are respectively optimal and feasible for Model (75), Equation (76b) comes from adding $\sum_{\ell \in \{1,\ldots,N\} \setminus \{\ell', \ell''\}} f_\ell(\hat{\gamma}^L_\ell, \hat{\gamma}^B_\ell)$ to both sides of (76a), and Equation (76c) applies the definition of $F(\cdot,\cdot)$ to both sides after utilizing the fact that $\tilde{\gamma}^B = \hat{\gamma}^B$ and $\tilde{\gamma}^L_\ell = \hat{\gamma}^L_\ell,\ \forall \ell \in \{1,\ldots,N\} \setminus \{\ell', \ell''\}$. Therefore, $(\tilde{\gamma}^L, \tilde{\gamma}^B)$ has at most one noninteger $\{\tilde{\gamma}^L_{\ell'}, \tilde{\gamma}^L_{\ell''}\}$, and $F(\tilde{\gamma}^L, \tilde{\gamma}^B) \leq F(\hat{\gamma}^L, \hat{\gamma}^B)$. Because this solution is feasible to Model (71) and its objective value no worse than that of $(\hat{\gamma}^L, \hat{\gamma}^B)$, this solution must be optimal to Model (71).

Case 2: Suppose $\hat{\gamma}^L_{\ell'}$ and $\hat{\gamma}^B_{\ell''}$ are fractional. Define $z = \hat{\gamma}^L_{\ell'} + \hat{\gamma}^B_{\ell''}$. In Model (71), when $\gamma^L_\ell = \hat{\gamma}^L_\ell,\ \ell \in \{1,\ldots,N\} \setminus \{\ell'\}$ and $\gamma^B_\ell = \hat{\gamma}^B_\ell,\ \ell \in \{1,\ldots,N\} \setminus \{\ell''\}$ are fixed, the model reduces to

$$\min\ f_{\ell'}(\gamma^L_{\ell'}, \hat{\gamma}^B_{\ell'}) + f_{\ell''}(\hat{\gamma}^L_{\ell''}, \gamma^B_{\ell''}), \tag{77a}$$

$$\text{s.t. } \gamma^L_{\ell'} + \gamma^B_{\ell''} \leq z, \tag{77b}$$

$$0 \leq \gamma^L_{\ell'},\ \gamma^L_{\ell''} \leq 1, \tag{77c}$$

but this also has an extreme point optimal solution (see, e.g., classical nonlinear programming references [3, 46]) because $f_{\ell'}(\gamma^L_{\ell'}, \hat{\gamma}^B_{\ell''})$ is linear in $\gamma^L_{\ell'}$ and $f_{\ell''}(\hat{\gamma}^L_{\ell'}, \gamma^B_{\ell''})$ is concave in $\gamma^B_{\ell''}$ (due to Theorem 9), thus implying objective (77a) is concave. By the same argument as in Case 1, we can

obtain $(\tilde{\gamma}^L, \tilde{\gamma}^B)$ that is optimal for Model (71) such that at most one of $\{\tilde{\gamma}^L_{\ell'}, \tilde{\gamma}^B_{\ell''}\}$ is noninteger.

Case 3: Suppose $\hat{\gamma}^B_{\ell'}$ and $\hat{\gamma}^B_{\ell''}$ are fractional and let $z = \hat{\gamma}^B_{\ell'} + \hat{\gamma}^B_{\ell''}$. When $\gamma^L = \hat{\gamma}^L$ and $\gamma^B_\ell = \hat{\gamma}^B_\ell,\ \ell \in \{1,\ldots,N\} \setminus \{\ell', \ell''\}$ are fixed in Model (71), the model becomes

$$\min\ f_{\ell'}(\hat{\gamma}^L_{\ell'}, \gamma^B_{\ell'}) + f_{\ell''}(\hat{\gamma}^L_{\ell''}, \gamma^B_{\ell''}), \tag{78a}$$

$$\text{s.t. } \gamma^B_{\ell'} + \gamma^B_{\ell''} \leq z, \tag{78b}$$

$$0 \leq \gamma^L_{\ell'},\ \gamma^L_{\ell''} \leq 1. \tag{78c}$$

According to Theorem 9, $f_{\ell'}(\hat{\gamma}^L_{\ell'}, \gamma^B_{\ell'})$ and $f_{\ell''}(\hat{\gamma}^L_{\ell''}, \gamma^B_{\ell''})$ are concave in $\gamma^B_{\ell'}$ and $\gamma^B_{\ell''}$, respectively. Therefore, Model (78) has an extreme point optimal solution. Analogous to the argument given in case 1, this implies existence of $(\tilde{\gamma}^L, \tilde{\gamma}^B)$ that is optimal for Model (71) with at most one of $\{\tilde{\gamma}^L_{\ell'}, \tilde{\gamma}^B_{\ell''}\}$ noninteger.

Note that in the case where both $\{\gamma^L_\ell, \gamma^B_\ell\}$ are noninteger for $\ell = \ell'$ and/or $\ell = \ell''$, we may apply Cases 1–3 at most three times to construct an optimal solution to Model (71) with only $k^\star - 1$ fractional subsystems. This establishes a contradiction (as $(\hat{\gamma}^L, \hat{\gamma}^B)$ was selected with minimum number of fractional subsystems) and proves $k^\star \leq 1$. □

Corollary 1. There exists an optimal solution of Model (71) such that $\gamma^L_\ell + \gamma^B_\ell \in \{0, \Phi - \lfloor\Phi\rfloor, 1, 1 + \Phi - \lfloor\Phi\rfloor, 2\},\ \forall \ell = 1,\ldots,N$.

Proof. In Remark 2, it is proven that $F(\gamma^L, \gamma^B)$ is decreasing in γ^L_ℓ and $\gamma^B_\ell,\ \ell = 1,\ldots,N$ whenever all other γ^L- and γ^B-variables are fixed; hence in an optimal solution, $\sum_{\ell=1}^N (\gamma^L_\ell + \gamma^B_\ell) = \Phi$. Due to Theorem 10, there exists an optimal solution in which there are $N-1$ subsystems ℓ such that γ^L_ℓ and γ^B_ℓ are integer. Without loss of generality, suppose this is the case for subsystems $1,\ldots,N-1$. Because $\Phi = \sum_{\ell=1}^N (\gamma^L_\ell + \gamma^B_\ell)$ and $\gamma^L_\ell + \gamma^B_\ell$ is integer for $\ell = 1,\ldots,N-1$, subsystem N satisfies

$$(\gamma^L_N + \gamma^B_N) - \lfloor \gamma^L_N + \gamma^B_N \rfloor = \Phi - \lfloor\Phi\rfloor, \tag{79}$$

which yields the result. □

Remark 3. If Φ is integer, Corollary 1 implies existence of an optimal solution of Model (71) in which $\gamma_\ell^L + \gamma_\ell^B \in \{0,1,2\},\ \forall \ell = 1,\ldots,N$.

In the following, we use the previous result and local optimality conditions to further restrict the candidate set of optimal solutions to the separation problem such that the final candidate set is finite for given τ, giving rise to the dynamic programming approach for solving the problem.

Theorem 11. For $0 < z < 2$ and $\tau_\ell > 1$, define quadratic equation coefficients $b_\ell(z)$ and $c_\ell(z)$ as

$$b_\ell(z) = -\left[2\sigma_\ell^L + (\bar{\beta}_\ell \sigma_\ell^L + \sigma_\ell^B(-\bar{\lambda}_\ell + z\sigma_\ell^L))\ln(\tau_\ell)\right]\sigma_\ell^B, \tag{80a}$$

$$c_\ell(z) = -\left[\sigma_\ell^B\left(\bar{\lambda}_\ell - z\sigma_\ell^L\right) - \bar{\beta}_\ell\sigma_\ell^L + \sigma_\ell^B \ln(\tau_\ell)\bar{\lambda}_\ell\left(z\sigma_\ell^B + \bar{\beta}_\ell\right)\right], \tag{80b}$$

and define $a_\ell = \left(\sigma_\ell^B\right)^2 \sigma_\ell^L \ln(\tau_\ell)$. In this case

$$\min\ \left\{f_\ell\left(\gamma_\ell^L, \gamma_\ell^B\right) \mid \gamma_\ell^L + \gamma_\ell^B = z,\ 0 \le \gamma_\ell^L, \gamma_\ell^B \le 1\right\}, \tag{81}$$

has an optimal solution in the set

$$C(z,\tau_\ell) = \left\{(\lfloor z\rfloor, z - \lfloor z\rfloor), \left(\frac{-b_\ell(z) + \sqrt{b_\ell(z)^2 - 4a_\ell c_\ell(z)}}{2a_\ell}, z - \frac{-b_\ell(z) + \sqrt{b_\ell(z)^2 - 4a_\ell c_\ell(z)}}{2a_\ell}\right),\right.$$
$$\left.\left(\frac{-b_\ell(z) - \sqrt{b_\ell(z)^2 - 4a_\ell c_\ell(z)}}{2a_\ell}, z - \frac{-b_\ell(z) - \sqrt{b_\ell(z)^2 - 4a_\ell c_\ell(z)}}{2a_\ell}\right), (z - \lfloor z\rfloor, \lfloor z\rfloor)\right\}. \tag{82}$$

Proof. When $\gamma_\ell^L + \gamma_\ell^B = z$, Model (81) simplifies (because f_ℓ is decreasing) for subsystem ℓ by making the substitution $\gamma_\ell^B = z - \gamma_\ell^L$ to obtain the one-variable model

$$\min g_\ell\left(z, \gamma_\ell^L\right) = f_\ell(\gamma_\ell^L, z - \gamma_\ell^L), \tag{83a}$$

$$\text{s.t.}\ \ \max\{0, z-1\} \le \gamma_\ell^L \le \max\{1, z\}. \tag{83b}$$

The minimum value of $g_\ell\left(z, \gamma_\ell^L\right)$ over $\max\{0,z-1\} \le \gamma_\ell^L \le \max\{1,z\}$ occurs either at its extreme points $\gamma_\ell^L \in \{\lfloor z\rfloor, z - \lfloor z\rfloor\}$ or when ${}^{\partial g_\ell(z,\gamma_\ell^L)}/_{\partial \gamma_\ell^L} = 0$. Letting $K = -\tau_\ell^{z\sigma_\ell^B + \bar{\beta}_\ell - \sigma_\ell^B\gamma_\ell^L - 1}T$, the derivative

is given as

$$\frac{\partial g_\ell\left(z,\gamma_\ell^L\right)}{\partial \gamma_\ell^L} = K\left(\bar{\beta}_\ell\sigma_\ell^L - \sigma_\ell^B(\bar{\lambda}_\ell + \sigma_\ell^L(2\gamma_\ell^L - z)) - \sigma_\ell^B(\sigma_\ell^B(z-\gamma_\ell^L) + \bar{\beta}_\ell)(\bar{\lambda}_\ell + \gamma_\ell^L\sigma_\ell^L)\ln(\tau_\ell)\right), \quad (84)$$

which yields the roots $\gamma_\ell^L = \frac{-b_\ell(z)\pm\sqrt{b_\ell(z)^2-4a_\ell c_\ell(z)}}{2a_\ell}$ under ${}^{\partial h(z,\gamma_\ell^L)}/_{\partial\gamma_\ell^L} = 0$. □

Remark 4. For $\tau_\ell = 1$, the root of ${}^{\partial g_\ell(z,\gamma_\ell^L)}/_{\partial\gamma_\ell^L} = 0$ is $\gamma_\ell^L = {}^{\bar{\beta}_\ell\sigma_\ell^L+\sigma_\ell^B(z\sigma_\ell^L-\bar{\lambda}_\ell)}/_{2\sigma_\ell^L}$. Therefore, in this case, Model (81) has an optimal solution in

$$C(z,1) = \Bigg\{ (\lfloor z\rfloor, z-\lfloor z\rfloor), (z-\lfloor z\rfloor, \lfloor z\rfloor), \left(\frac{\bar{\beta}_\ell\sigma_\ell^L + \sigma_\ell^B(\sigma_\ell^L-\bar{\lambda}_\ell)}{2\sigma_\ell^L}, z - \frac{\bar{\beta}_\ell\sigma_\ell^L + \sigma_\ell^B(\sigma_\ell^L-\bar{\lambda}_\ell)}{2\sigma_\ell^L}\right)\Bigg\}. \quad (85)$$

Remark 5. Some of the points in $C(z,\tau_\ell)$ may be infeasible because they may be negative or greater that one, so the set can be further restricted.

Throughout the remainder of the document, let $\left(\bar{\gamma}_\ell^L(z), \bar{\gamma}_\ell^B(z)\right)$ denote any element of the set $\arg\min\{f_\ell(\hat{\gamma}_\ell^L,\hat{\gamma}_\ell^B) \mid (\hat{\gamma}_\ell^L,\hat{\gamma}_\ell^B) \in C(z,\tau_\ell);\ 0 \le \hat{\gamma}_\ell^L,\hat{\gamma}_\ell^B \le 1\}$. By using Theorem 11 and Remark 5, $\left(\bar{\gamma}_\ell^L(z), \bar{\gamma}_\ell^B(z)\right)$ provides an optimal solution to Model (81). In Theorems 12 and 13, we extend these results to summarize candidate solutions for Model (71).

Theorem 12. For integer value Φ, define

$$S_{\ell,\Phi} \equiv \left\{\left(\gamma^L,\gamma^B\right) \in U_\Phi \mid \left(\gamma_\ell^L,\gamma_\ell^B\right) \in \left\{(0,0), \left(\bar{\gamma}_\ell^L(1), \bar{\gamma}_\ell^B(1)\right), (1,1)\right\}\right\}, \quad (86)$$

and let $S_\Phi \equiv \bigcap_{\ell=1}^N S_{\ell,\Phi}$. For integer-valued Φ, the set S_Φ contains an optimal solution to Model (71).

Proof. Due to Remark 3, there is an optimal solution of Model (71) that satisfies $\gamma_\ell^L + \gamma_\ell^B \in \{0,1,2\}$ for all subsystems $\ell = 1,\ldots,N$. When $\gamma_\ell^L + \gamma_\ell^B = 0$, we have $\gamma_\ell^L = \gamma_\ell^B = 0$, and when $\gamma_\ell^L + \gamma_\ell^B = 1$, we have $\gamma_\ell^L = \gamma_\ell^B = 1$. Also, $(\bar{\gamma}_\ell^L(1), \bar{\gamma}_\ell^B(1))$ is optimal for Model (71) when $\gamma_\ell^L + \gamma_\ell^B = 1$. □

For noninteger Φ, we also show that there exists an optimal solution to Model (71) among a finite set of candidate solutions in Theorem 13.

Theorem 13. For noninteger Φ, define $\alpha = \Phi - \lfloor \Phi \rfloor$ and $\alpha' = 1 + \Phi - \lfloor \Phi \rfloor$. Define

$$S_{\ell,\Phi} \equiv \Bigg\{ \left(\gamma^L, \gamma^B\right) \in U_\Phi \Bigg| (\gamma_\ell^L, \gamma_\ell^B) \in \Bigg\{ (0,0), \left(\bar{\gamma}_\ell^L(\alpha), \bar{\gamma}_\ell^B(\alpha)\right), \left(\bar{\gamma}_\ell^L(1), \bar{\gamma}_\ell^B(1)\right), \left(\bar{\gamma}_\ell^L(\alpha'), \bar{\gamma}_\ell^B(\alpha')\right), (1,1) \Bigg\} \Bigg\}, \tag{87}$$

and let $S_\Phi \equiv \bigcap_{\ell=1}^N S_{\ell,\Phi}$. When Φ is noninteger, S_Φ contains an optimal solution to Model (71).

Proof. According to Corollary 1, there is an optimal solution of Model (71) satisfying $\gamma_\ell^L + \gamma_\ell^B \in \{0, \Phi - \lfloor \Phi \rfloor, 1, 1 + \Phi - \lfloor \Phi \rfloor, 2\}$ for all subsystems $\ell = 1, \ldots, N$. Hence, based on Theorem 11, there exists an optimal solution to Model (71) in which

$$\left(\gamma_\ell^L, \gamma_\ell^B\right) \in \left\{(0,0), (\bar{\gamma}_\ell^L(\alpha), \bar{\gamma}_\ell^B(\alpha)), (\bar{\gamma}_\ell^L(1), \bar{\gamma}_\ell^B(1)), (\bar{\gamma}_\ell^L(\alpha'), \bar{\gamma}_\ell^B(\alpha')), (1,1)\right\}. \tag{88}$$

□

Theorems 12 and 13 establish a finite candidate set of optimal solutions for Model (71). This enables an efficient dynamic programming approach for solving Model (71), in which the number of stages is equal to the number of subsystems, and each stage ℓ contains a number of possible states corresponding to the amount of the original uncertainty budget that remains at the beginning of that stage. We now describe the dynamic programming algorithm.

For integer Φ, define $Q_\Phi = \{1, \ldots, \Phi\}$ and for noninteger Φ define $Q_\Phi = \{1, \ldots, \lfloor \Phi \rfloor\} \cup \{(\Phi - \lfloor \Phi \rfloor), 1 + (\Phi - \lfloor \Phi \rfloor), \ldots, \lfloor \Phi \rfloor + (\Phi - \lfloor \Phi \rfloor) = \Phi\}$. Initialize the value function as $q_0(\phi) = 0$ for all $\phi \in Q_\Phi$. For $\ell = 1, \ldots, N$ and integer $\phi \in \{Q_\Phi \mid \phi \in \mathbb{Z}\}$, calculate $q_\ell(\phi)$ as

$$q_\ell(\phi) = \min\{q_{\ell-1}(\phi) + f_\ell(0,0), q_{\ell-1}(\phi - 1) + f_\ell\left(\bar{\gamma}_\ell^L(1), \bar{\gamma}_\ell^B(1)\right), q_{\ell-1}(\phi - 2) + f_\ell(1,1)\}. \tag{89}$$

For $\ell = 1, \ldots, N$ and noninteger $\phi \in \{Q_\Phi \mid \phi \notin \mathbb{Z}\}$, calculate $q_\ell(\phi)$ as

$$\begin{aligned} q_\ell(\phi) = \min\{&q_{\ell-1}(\phi) + f_\ell(0,0), q_{\ell-1}(\phi-\alpha) + f_\ell\left(\bar{\gamma}_\ell^L(\alpha), \bar{\gamma}_\ell^B(\alpha)\right), q_{\ell-1}(\phi-1) \\ &+ f_\ell\left(\bar{\gamma}_\ell^L(1), \bar{\gamma}_\ell^B(1)\right), q_{\ell-1}(\phi-\alpha') + f_\ell\left(\bar{\gamma}_\ell^L(\alpha'), \bar{\gamma}_\ell^B(\alpha')\right), q_{\ell-1}(\phi-2) + f_\ell(1,1)\}, \end{aligned} \tag{90}$$

in which α and α' are defined in Theorem 13. The optimal objective value of Model (71) is equal to $G(\tau) = q_N(\Phi)$.

We now describe an exact solution approach for Model (69). By applying the variable substitution $L \equiv \ln R$ to the equivalent Model (70), Model (69) is equivalent (due to monotonicity of the natural log function) to

$$\max \ L, \tag{91a}$$

$$\text{s.t.} \ \sum_{\ell=1}^{N} -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T \geq L, \ \forall (\gamma^L, \gamma^B) \in U_\Phi, \tag{91b}$$

$$\tau \in X. \tag{91c}$$

Replacing U_Φ in (91b) with a subset $\bar{D} \subset U_\phi$ yields the outer approximation

$$E(\bar{D}) = \max \ L, \tag{92a}$$

$$\text{s.t.} \ \sum_{\ell=1}^{N} -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T \geq L, \ \forall (\gamma^L, \gamma^B) \in \bar{D}, \tag{92b}$$

$$\tau \in X. \tag{92c}$$

Because the feasible region of Model (92) contains the feasible region of Model (91), Model (92) provides an upper bound for the logarithm of the objective value of Model (69) (i.e., $e^{E(\bar{D})} \geq R^\star$). Algorithm 1 formally states the cutting-plane algorithm.

Algorithm 1 Cutting plane (calculate $R^\star$ within ε tolerance).

1: **function** CUTTINGPLANE (Φ, ε)
2: Initialize $\bar{D} \subset U_\Phi$. **Set** $LB \leftarrow 0$ and $UB \leftarrow 1$.
3: **Solve** the outer approximation problem with optimal objective $e^{E(\bar{D})}$ and solution $\hat{\tau}$.
4: **Set** $UB \leftarrow e^{E(\bar{D})}$.
5: **Solve** the dynamic programming with optimal objective $e^{G(\hat{\tau})}$ and solution $\left(\hat{\gamma}^L, \hat{\gamma}^B\right)$.
6: **If** $e^{G(\hat{\tau})} > LB$ **then set** $LB \leftarrow e^{G(\hat{\tau})}$ and $\tau^\star \leftarrow \hat{\tau}$.
7: **If** $UB - LB < \varepsilon$ **then return** $\tau^\star$ ▷ $\tau^\star$ is ε-optimal
8: **Set** $\bar{D} = \bar{D} \cup \left(\hat{\gamma}^L, \hat{\gamma}^B\right)$. Go to Line 3.
9: **end function**

In Theorem 14, we prove that Model (92) is a convex optimization problem, so the cutting-plane algorithm can be solved efficiently using general purpose convex optimization software.

Theorem 14. Model (92) is a convex optimization problem.

Proof. For fixed value $\left(\gamma_\ell^L, \gamma_\ell^B\right)$, define $h_\ell(\tau_\ell)$ such that

$$h_\ell(\tau_\ell) = -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T. \tag{93}$$

The first and second derivatives of $h_\ell(\tau_\ell)$ with respect to the parameter τ_ℓ are equal to

$$\frac{\partial h_\ell(\tau_\ell)}{\partial \tau_\ell} = -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 2)} T, \tag{94a}$$

$$\frac{\partial^2 h_\ell(\tau_\ell)}{\partial^2 \tau_\ell} = -(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1) \times (\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 2)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 3)} T < 0. \tag{94b}$$

Therefore, $h_\ell(\tau_\ell)$ is concave in τ_ℓ. Constraint (92b) is equivalent to $\sum_{\ell=1}^N h_\ell(\tau_\ell) \geq L$. Because each $h_\ell(\tau_\ell)$-term is concave, $\sum_{\ell=1}^N h_\ell(\tau_\ell)$ is concave and (92b) thus define a convex region. The remaining constraints are linear; hence (92) has a convex feasible region and a linear objective and is therefore a convex optimization problem. □

We now prove the convergence of the algorithm.

Lemma 10. Define $\tau_\ell^{\min} \equiv \tau_\ell^0$ and $\tau_\ell^{\max} \equiv \min_{k=1}^{K} \left\{ {\left(b_k - \Sigma_{\ell'=1}^{N} c_{\ell'}^k \tau_{\ell'}^0 + c_\ell^k \tau_\ell^0\right)}/{c_\ell^k} \right\}$ as the minimum and maximum possible testing time for subsystem $\ell = 1,\ldots,N$, respectively. For $\delta > 0$, let $J_\ell = \left\lceil {\left(\tau_\ell^{\max} - \tau_\ell^{\min}\right)}/{\delta} \right\rceil$ and $J = \max_{\ell=1}^{N}\{J_\ell\}$. Define $\tau_\ell^j,\ \ell = 1,\ldots,N$ as the obtained optimal solution of the outer approximation in iteration j. Among $J^N + 1$ iterations, there are two iterations j' and j'' such that $|\tau_\ell^{j''} - \tau_\ell^{j'}| \le \delta$ for all $\ell = 1,\ldots,N$.

Proof. For each subsystem $\ell = 1,\ldots,N$, divide interval $[\tau_\ell^{\min}, \tau_\ell^{\max}]$ to J equal and disjoint intervals with lengths equal to ${\left(\tau_\ell^{\max} - \tau_\ell^{\min}\right)}/{J} \le {\left(\tau_\ell^{\max} - \tau_\ell^{\min}\right)}/{J_\ell} \le \delta$. Each subsystem's testing time is in one of J intervals, hence there are no more than J^N combinations of possible intervals for system testing time solution τ^j. Therefore, after $J^N + 1$ iterations, there are two solutions with testing times in same intervals. □

Lemma 11. For solutions $\tau^{j''}$ and $\tau^{j'}$, define $\varepsilon_\ell = |R(\tau_\ell^{j''}; T, \lambda_\ell, \beta_\ell) - R(\tau_\ell^{j'}; T, \lambda_\ell, \beta_\ell)|$, $\varepsilon_{\max} = \max_{\ell=1,\ldots,N}\{\varepsilon_\ell\}$, and $\varepsilon \equiv {N\varepsilon_{\max}}/{[1-(N+1)\varepsilon_{\max}]}$. If $\varepsilon_{\max} \le {1}/{N+1}$, then $|R(\tau^{j''}, \lambda, \beta) - R(\tau^{j'}, \lambda, \beta)| \le \varepsilon$.

Proof. We invoke a special case of Lemma 1. We consider a series system, so $\bar{M}_\ell = 1$. Define ε_ℓ as the absolute value of the difference of $|R(\tau_\ell^{j'}; T, \lambda_\ell, \beta_\ell) - R(\tau_\ell^{j''}; T, \lambda_\ell, \beta_\ell)|$. By setting $N = \bar{N}$, $M_{\max} = 1$, $\varepsilon_{\max} = \bar{\varepsilon}_{\max}$ and $\varepsilon = \bar{\varepsilon}$ in Lemma 1, $\bar{M}_{\max}\bar{\varepsilon}_{\max} \le 1$ and ${\bar{N}\bar{M}_{\max}\bar{\varepsilon}_{\max}}/{(1-\bar{M}_{\max}\bar{\varepsilon}_{\max})} \le 1$ result in $\varepsilon_{\max} \le {1}/{N+1}$, $\bar{\varepsilon} \equiv {\bar{N}\bar{M}_{\max}\bar{\varepsilon}_{\max}}/{[1-(\bar{N}+1)\bar{M}_{\max}\bar{\varepsilon}_{\max}]}$ suggests $\varepsilon \equiv {N\varepsilon_{\max}}/{[1-(N+1)\varepsilon_{\max}]}$. Under $\bar{\bar{\rho}}_{\ell,1} = \max\left\{R(\tau_\ell^{j'}; T, \lambda_\ell, \beta_\ell), R(\tau_\ell^{j''}; T, \lambda_\ell, \beta_\ell)\right\}$ and $\bar{\rho}_{\ell,1} = \min\left\{R(\tau_\ell^{j'}; T, \lambda_\ell, \beta_\ell), R(\tau_\ell^{j''}; T, \lambda_\ell, \beta_\ell)\right\}$, Lemma 1 implies

$$\left|\prod_{\ell=1}^{N} R(\tau_\ell^{j''}; T, \lambda_\ell, \beta_\ell) - \prod_{\ell=1}^{N} R(\tau_\ell^{j'}; T, \lambda_\ell, \beta_\ell)\right| \le \left|\prod_{\ell=1}^{N} \bar{\bar{\rho}}_{\ell,1} - \prod_{\ell=1}^{N} \bar{\rho}_{\ell,i}\right| = \prod_{\ell=1}^{N} \bar{\bar{\rho}}_{\ell,1} - \prod_{\ell=1}^{N} \bar{\rho}_{\ell,i} \le \varepsilon. \quad (95)$$

□

Theorem 15. For a series system, the cutting-plane algorithm identifies an ε-optimal solution in a finite number of iterations.

Proof. In Lemma 7, it is proven that for a given $\varepsilon_{\max} > 0$, there exists $\delta_\ell > 0$ such that $R(\tau_\ell + \delta_\ell; T, \lambda_\ell, \beta_\ell) - R(\tau_\ell; T, \lambda_\ell, \beta_\ell) \le \varepsilon_{\max}$ for $\tau_\ell^0 \le \tau_\ell \le \tau_\ell^{\max}$. Define $\delta = \min_{\ell=1,\ldots,N}\{\delta_\ell\}$. According to Lemma 10, there exist two iterations j' and j'' such that $|\tau_\ell^{j''} - \tau_\ell^{j'}| \le \delta$, which suggests

$R(\tau_\ell^{j''};T,\lambda_\ell,\beta_\ell)-R(\tau_\ell^{j'};T,\lambda_\ell,\beta_\ell)\leq\varepsilon_{\max}$. Set $\varepsilon_{\max}={}^{\varepsilon}/_{[N+\varepsilon(N+1)]}$; therefore, $\varepsilon={}^{N\varepsilon_{\max}}/_{[1-(N+1)\varepsilon_{\max}]}$. However,

$$\varepsilon_{\max}=\frac{\varepsilon}{[N+\varepsilon(N+1)]}\leq\frac{\varepsilon}{[\varepsilon(N+1)]}=\frac{1}{[N+1]}. \tag{96}$$

By applying Lemma 11, $|R(\tau^{j''},\lambda,\beta)-R(\tau^{j'},\lambda,\beta)|\leq\varepsilon$. Without loss of generality, assume $j'<j''$, and consider $(\hat{\lambda},\hat{\beta})=(\bar{\lambda}+\sigma^L*\hat{\gamma}^L,\bar{\beta}+\sigma^B*\hat{\gamma}^B)$ as the optimal solution of the separation problem in iteration j', so $R(\tau^{j'},\hat{\lambda},\hat{\beta})$ is a lower bound for $R^\star$ (the optimal objective value of Model (69)) in iteration j'. Because $j'<j''$, $(\hat{\gamma}^L,\hat{\gamma}^B)\in\bar{D}$ in iteration j'', so

$$R(\tau^{j''},\hat{\lambda},\hat{\beta})\geq e^{E(\bar{D})}\geq R^\star, \tag{97}$$

which suggests $R(\tau^{j''},\hat{\lambda},\hat{\beta})$ is an upper bound for $R^\star$. □

In the following section, we develop the robust optimization models and solution methodology for a series-parallel system. However, before doing so, we first provide some commentary regarding the assumption that $\tau_\ell^0\geq 1,\ \ell=1,\ldots,N$, that was used in this section. This assumption only enables solving the separation model, Model (71), via dynamic programming. Given some other algorithm for solving Model (71), the outer approximation Model (92) remains a convex optimization problem. The convergence result, Theorem 15, extends in this case as well, although the required computational effort per iteration may be increased as a result of having to solve Model (71) by another means. The series-parallel algorithm developed in the following section is somewhat more computationally complex than its series-system counterpart; however, it does not require the assumption that $\tau_\ell^0\geq 1,\ \ell=1,\ldots,N$. As a result, one alternative for solving Model (71) is the special case of Algorithm 2—the series-parallel separation algorithm provided in the following section—specialized to the case of one component per subsystem.

3.4 Solution Method for Series-Parallel Systems

In this section, we develop a model for the robust allocation of testing resources across the subsystems of a series-parallel system, in which the components inside each subsystem are identical. In

this system, there are N subsystems connected in a series, and each subsystem $\ell = 1, \ldots, N$ includes M_ℓ identical components connected in parallel. Based on Equation (63), the robust optimization problem for a series-parallel system is defined as

$$R^\star = \max \; R, \tag{98a}$$

$$\text{s.t.} \; \prod_{\ell=1}^{N} \left[1 - \left(1 - e^{-(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B) \tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T} \right)^{M_\ell} \right] \geq R, \; \forall (\boldsymbol{\gamma}^L, \boldsymbol{\gamma}^B) \in U_\Phi, \tag{98b}$$

$$\boldsymbol{\tau} \in X, \tag{98c}$$

in which U_Φ is defined in Equation (66).

Model (98) is nonlinear and nonconvex, so there is no apparent exact solution method. We now describe such a method based on a cutting-plane algorithm. The separation model associated with calculating the robust objective value for $\boldsymbol{\tau} \in X$ is given by

$$G(\boldsymbol{\tau}) = \min \; \prod_{\ell=1}^{N} \left[1 - \left(1 - e^{-(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B) \tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T} \right)^{M_\ell} \right], \tag{99a}$$

$$\text{s.t.} \; (\boldsymbol{\gamma}^L, \boldsymbol{\gamma}^B) \in U_\Phi. \tag{99b}$$

A solution of Model (99) is feasible for Model (98); therefore, Model (99) yields a lower bound for Model (98) (i.e. $G(\boldsymbol{\tau}) \leq R^\star$). Unfortunately, the series-parallel extensions of Theorems 12–13 no longer hold for Model (99). We address this issue by using a discretize-and-refine method developed for the deterministic version of our problem in Chapter 2. We will adapt this method to solve the separation and outer approximation problems for series-parallel systems. This method is based on the idea that lower and upper bounds on the robust reliability solution can be found by solving the discretized model. Then, by successive refinement of the discretized models, the lower and upper bounds converge to an arbitrarily small gap. We begin by discretizing the separation model. For this purpose, for each subsystem $\ell = 1, \ldots, N$, consider P_ℓ discrete points $z_\ell^p, \; p = 1, \ldots, P_\ell$, each of which represents a potential value of $\gamma_\ell^L + \gamma_\ell^B$, and define r_ℓ^p as the reliability associated

with point $p = 1, \ldots, P_\ell$. The discretized version of the separation model (99) is represented as

$$R^{\mathrm{SD}}(\mathbf{r}, T) = \min \prod_{\ell=1}^{N} \prod_{p=1}^{P_\ell} r_\ell^{p x_\ell^p}, \tag{100a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} \sum_{p=1}^{P_\ell} z_\ell^p x_\ell^p \leq \Phi, \tag{100b}$$

$$\sum_{p=1}^{P_\ell} x_\ell^p = 1, \qquad \forall \ell = 1, \ldots, N, \tag{100c}$$

in which $\mathbf{r}$ is a vector of reliabilities r_ℓ^p and

$$0 \equiv z_\ell^1 < z_\ell^2 < \cdots < z_\ell^{P_\ell} \equiv 2 (< z_\ell^{P_\ell+1}), \tag{101}$$

where $z_\ell^{P_\ell+1}$ is defined (for notational convenience) to be a constant greater than 2. We now show how Model (100) can provide lower and upper bounds on subsystem reliability. Due to Theorem 11, when $\gamma_\ell^L + \gamma_\ell^B$ is fixed to equal z, Model (99) has an optimal solution in the set $\arg\min\{f_\ell(\hat{\gamma}_\ell^L, \hat{\gamma}_\ell^B) | (\hat{\gamma}_\ell^L, \hat{\gamma}_\ell^B) \in C(z, \tau_\ell)\}$. Let $(\bar{\gamma}_\ell^L(z), \bar{\gamma}_\ell^B(z))$ denote any such solution. Setting

$$r_\ell^p = 1 - \left(1 - f_\ell(\bar{\gamma}_\ell^L(z_\ell^{p+1}), \bar{\gamma}_\ell^B(z_\ell^{p+1}))\right)^{M_\ell}, \ \forall p = 1, \ldots, P, \ \ell = 1, \ldots, N, \tag{102}$$

in Model (100) provides a lower bound on Model (99), but setting

$$r_\ell^p = 1 - \left(1 - f_\ell(\bar{\gamma}_\ell^L(z_\ell^p), \bar{\gamma}_\ell^B(z_\ell^p))\right)^{M_\ell}, \ \forall p = 1, \ldots, P, \ \ell = 1, \ldots, N, \tag{103}$$

yields an upper bound. Towards proving this formally, define LB-MILP and UB-MILP as the lower- and upper-bound models, respectively, and define $\hat{R}(z, T)$ as the optimal objective value of Model (99) when the value of $\gamma_\ell^L + \gamma_\ell^B$ is equal to z_ℓ for $\ell = 1, \ldots, N$. (Here, z denotes the collection of possibly noninteger z_ℓ-values across all subsystems.)

Theorem 16. Define $\bar{r}_\ell^p = 1 - \left(1 - f_\ell(\bar{\gamma}_\ell^L(z_\ell^{p+1}), \bar{\gamma}_\ell^B(z_\ell^{p+1}))\right)^{M_\ell}$ for all $\ell = 1, \ldots, N, \ p = 1, \ldots, P_\ell$, then $R^{\mathrm{SD}}(\bar{\mathbf{r}}, T) \leq G(\tau)$ (LB-MILP yields a lower bound on the separation problem).

Proof. Let $\dot{\gamma}_\ell^L$ and $\dot{\gamma}_\ell^B$, $\ell = 1,\ldots,N$ denote an optimal solution for Model (99). Then, for each subsystem ℓ, find the index $\hat{P}_\ell \in \{1,\ldots,P_\ell\}$ satisfying $z_\ell^{\hat{P}_\ell} \leq \dot{\gamma}_\ell^L + \dot{\gamma}_\ell^B < z_\ell^{\hat{P}_\ell+1}$, and set $\hat{z}_\ell = z_\ell^{\hat{P}_\ell+1}$. The solution $x_\ell^{\hat{P}_\ell} = 1,\ \ell = 1,\ldots,N$ is a feasible solution to UB-MILP, with objective value equal to $\hat{R}(\hat{z},T)$. According to Theorem 9, $e^{f_\ell(\gamma_\ell^L,\gamma_\ell^B)}$ is nonincreasing over either γ_ℓ^L or γ_ℓ^B when the other is fixed. Thus, the value of $1-\left(1-e^{f_\ell(\gamma_\ell^L,\gamma_\ell^B)}\right)^{M_\ell}$ is nonincreasing over γ_ℓ^L or γ_ℓ^B when the other is fixed, which yields that $\hat{R}(z,T)$ is nonincreasing over the value of $\gamma_\ell^L + \gamma_\ell^B$. Therefore,

$$R^{\mathrm{SD}}(\bar{\mathbf{r}},T) \leq \hat{R}(\hat{z},T) \leq \hat{R}(\dot{z},T) = G(\tau), \tag{104}$$

which completes the proof. □

Theorem 17. Define $\bar{\bar{r}}_\ell^p = 1-\left(1-f_\ell(\bar{\gamma}_\ell^L(z_\ell^p),\bar{\gamma}_\ell^B(z_\ell^p))\right)^{M_\ell}$ for all $\ell = 1,\ldots,N$, $p = 1,\ldots,P_\ell$, then $R^{\mathrm{SD}}(\bar{\bar{\mathbf{r}}},T) \geq G(\tau)$ (UB-MILP yields an upper bound on the separation problem).

Proof. By setting $\bar{\bar{r}}_\ell^p = 1-\left(1-f_\ell(\bar{\gamma}_\ell^L(z_\ell^p),\bar{\gamma}_\ell^B(z_\ell^p))\right)^{M_\ell}$, for each solution $\hat{\mathbf{x}}$ in Model (100), there is a solution $\gamma_\ell^L = \sum_{p=1}^{P_\ell} \bar{\gamma}_\ell^L(z_\ell^p)\hat{x}_\ell^p$ and $\gamma_\ell^B = \sum_{p=1}^{P_\ell} \bar{\gamma}_\ell^B(z_\ell^p)\hat{x}_\ell^p$ with the same objective in Model (99). □

The separation problem is solved by Algorithm 2, where P_{sub} denotes the initial number of evenly spaced discrete points (see Line 5) used to populate Model (100) and $\varepsilon_{\mathrm{sub}}$ denotes the desired optimality tolerance. Define the N-vectors of $(\gamma^{L^\star},\gamma^{B^\star})$ as the incumbent (upper bound) solution and define $\bar{P}_\ell,\ \ell = 1,\ldots,N$, as the index p such that $x_\ell^p = 1$ in the obtained optimal solution to LB-SP. The LB-SP and UB-SP models are refined to generate tighter bounds in the following iteration by defining new points $z_\ell^{P_\ell+1}$ given by

$$z_\ell^{P_\ell+1} = \left(0.5z_\ell^{\bar{P}_\ell} + 0.5z_\ell^{\bar{P}_\ell+1}\right), \tag{105}$$

incrementing $P_\ell \leftarrow P_\ell + 1$ (unless $\bar{P}_\ell = P_\ell$, in which case the issue is addressed by Line 20 via redefining $z_\ell^{P_\ell+1}$ as $0.5z_\ell^{P_\ell} + 0.5z_\ell^{P_\ell+1}$).

Algorithm 2 Series-parallel separation algorithm (calculate a lower bound for $R^\star$ within ε_{sub} tolerance).

1: **function** SEPARATIONALGORITHM (ε_{sub}, P_{sub}, τ)
2: **for** $\ell = 1,\ldots,N$ **do**
3: Set $z_\ell^{\min} \leftarrow 0$ and $z_\ell^{\max} \leftarrow 2 + 2/(P_{\text{sub}} - 1)$
4: **for** $p = 1,\ldots,P_{\text{sub}} + 1$ **do**
5: Set $z_\ell^p \leftarrow z_\ell^{\min} + (p-1)(z_\ell^{\max} - z_\ell^{\min})/P_{\text{sub}}$
6: **end for**
7: **end for**
8: Set $LB \leftarrow 0$ and $UB \leftarrow 1$
9: **for** $\ell = 1,\ldots,N$ **do**
10: Set $P_\ell \leftarrow P_{\text{sub}}$
11: **end for**
12: **If** $UB - LB < \varepsilon_{\text{sub}}$ **then return** $(\boldsymbol{\gamma}^{L\star}, \boldsymbol{\gamma}^{B\star})$ ▷ $(\boldsymbol{\gamma}^{L\star}, \boldsymbol{\gamma}^{B\star})$ is ε-optimal
13: **Solve** the LB-MILP model with optimal solution $\boldsymbol{\gamma}^\star$ and objective value $R^{\text{SD}}(\bar{\mathbf{r}}, T)$
14: **Calculate** $\left(\bar{\gamma}_\ell^L, \bar{\gamma}_\ell^B\right) \in \arg\min\{f_\ell(\hat{\gamma}_\ell^L, \hat{\gamma}_\ell^B) \mid (\hat{\gamma}_\ell^L, \hat{\gamma}_\ell^B) \in C(\gamma_\ell^\star, \tau_\ell)\}, \forall \ell = 1,\ldots,N,$ using Theorem 13
15: **If** $R^{\text{SD}}(\bar{\mathbf{r}}, T) > LB$ **then** set $LB \leftarrow R^{\text{SD}}(\bar{\mathbf{r}}, T)$, $\boldsymbol{\gamma}^{L\star} \leftarrow \bar{\boldsymbol{\gamma}}^L$ and $\boldsymbol{\gamma}^{B\star} \leftarrow \bar{\boldsymbol{\gamma}}^B$
16: **Solve** the UB-MILP model with optimal solution $\bar{P}_\ell \in \{1,\ldots,P_\ell\},\ \ell = 1,\ldots,N,$ and objective value $R^{\text{SD}}(\bar{\bar{\mathbf{r}}}, T)$
17: **If** $R^{\text{SD}}(\bar{\bar{\mathbf{r}}}, T) < UB$ **then set** $UB \leftarrow R^{\text{SD}}(\bar{\bar{\mathbf{r}}}, T)$
18: **for** $\ell = 1,\ldots,N$ **do**
19: **If** $0.5\left(z_\ell^{\bar{P}_\ell} + z_\ell^{\bar{P}_\ell+1}\right) \leq 2$ **then add** a new point $z_\ell^{P_\ell+1} \equiv 0.5\left(z_\ell^{\bar{P}_\ell} + z_\ell^{\bar{P}_\ell+1}\right)$ and **set** $P_\ell \leftarrow P_\ell + 1$
20: **If** $0.5\left(z_\ell^{\bar{P}_\ell} + z_\ell^{\bar{P}_\ell+1}\right) > 2$ **then set** $z_\ell^{P_\ell+1} = 0.5\left(z_\ell^{\bar{P}_\ell} + z_\ell^{\bar{P}_\ell+1}\right)$
21: **end for**
22: **for** $\ell = 1,\ldots,N$ **do**
23: **Renumber** the points such that $0 \equiv z_\ell^1 < z_\ell^2 < \cdots < z_\ell^{P_\ell} \equiv 2 (< z_\ell^{P_\ell+1})$
24: **end for**
25: Go to Line 12
26: **end function**

In the following, we describe an approach for solving the series-parallel robust allocation Model (98). Considering $\bar{D} \subseteq U_\Phi$, the outer approximation problem is given as

$$E(\bar{D}) = \max\ R, \tag{106a}$$

$$\text{s.t.}\ \prod_{\ell=1}^{N}\left[1 - \left(1 - e^{-(\bar{\lambda}_\ell + \sigma_\ell^L \gamma_\ell^L)(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B)\tau_\ell^{(\bar{\beta}_\ell + \sigma_\ell^B \gamma_\ell^B - 1)} T}\right)^{M_\ell}\right] \geq R,$$

$$\forall \left(\boldsymbol{\gamma}^L, \boldsymbol{\gamma}^B\right) \in \bar{D}, \tag{106b}$$

$$\tau \in X. \tag{106c}$$

Model (106) is nonlinear/nonconvex, so we use a discretize-and-refine method similar to the method used for the separation problem to solve the outer approximation problem. The discretized version of Model (106) is given as

$$R^{\mathrm{OA}}(\mathbf{r},T) = \max\ R, \tag{107a}$$

$$\text{s.t.}\ \prod_{\ell=1}^{N}\prod_{p=1}^{P} r_{\ell,j}^{p\ y_\ell^p} \geq R,\ \forall j = 1,\ldots,|\bar{D}|, \tag{107b}$$

$$\sum_{p=1}^{P} \tau_\ell^p y_\ell^p \in X, \tag{107c}$$

$$\sum_{p=1}^{P} y_\ell^p = 1, \tag{107d}$$

$$y \in \{0,1\},\ \forall p = 1,\ldots,P. \tag{107e}$$

in which $\mathbf{r}$ is a vector of reliabilities $r_{\ell,j}^p$. Applying a logarithm to Objective (107a) and Equation (107b) yields the equivalent linear model

$$W^{\mathrm{OA}}(\mathbf{r},T) = \max L, \tag{108a}$$

$$\text{s.t.}\ \sum_{\ell=1}^{N}\sum_{p=1}^{P} \ln(r_{\ell,j}^p) y_\ell^p \geq L,\ \forall j = 1,\ldots,|\bar{D}|, \tag{108b}$$

Constraints (107c)–(107e),

in which $e^{W^{\mathrm{OA}}(\mathbf{r},T)}$ is equal to $R^{\mathrm{OA}}(\mathbf{r},T)$, the optimal objective value of Model (107). Define $(\lambda_{\ell,j},\beta_{\ell,j})$ $\ell = 1,\ldots,N$ as the AMSAA model parameters associated with scenario $j = 1,\ldots,|\bar{D}|$. We now prove that setting

$$r_{\ell,j}^p = 1 - \left(1 - e^{-\lambda_{\ell,j}\beta_{\ell,j}\tau_\ell^{p(\beta_{\ell,j}-1)}T}\right)^{M_\ell},\ \forall \ell = 1,\ldots,N,\ j = 1,\ldots,|\bar{D}|,\ p = 1,\ldots,P, \tag{109}$$

provides a lower bound on the optimal objective value $E(\bar{D})$; while, setting

$$r_{\ell,j}^{p} = 1 - \left(1 - e^{-\lambda_{\ell,j}\beta_{\ell,j}\tau_{\ell}^{p+1-(\beta_{\ell,j}-1)}T}\right)^{M_\ell},\ \forall \ell = 1,\ldots,N,\ j = 1,\ldots,|\bar{D}|,\ p = 1,\ldots,P, \qquad (110)$$

yields an upper bound (Define $r_{\ell,j}^{P+1} \equiv \tau_{\ell}^{\max}, \ell = 1,\ldots,N,\ j = 1,\ldots,|\bar{D}|$, where $\tau_{\ell}^{\max} \equiv \min_{k=1}^{K} \left\{ {\left(b_k - \Sigma_{\ell'=1}^{N} c_{\ell'}^{k}\tau_{\ell'}^{0} + c_{\ell}^{k}\tau_{\ell}^{0}\right)}/{c_{\ell}^{k}} \right\}$). Define LB-MD when Model (107) provides a lower bound on Model (106), and UB-MD when it provides an upper bound on Model (106).

Theorem 18. Define $\bar{r}_{\ell,j}^{p} = 1 - \left(1 - e^{-\lambda_{\ell,j}\beta_{\ell,j}\tau_{\ell}^{p-(\beta_{\ell,j}-1)}T}\right)^{M_\ell}$ for all $\ell = 1,\ldots,N,\ j = 1,\ldots,|\bar{D}|,\ p = 1,\ldots,P$, then $R^{\mathrm{OA}}(\bar{\mathbf{r}},T) \leq E(\bar{D})$. (That is, LB-MD provides a lower bound for the outer approximation problem).

Proof. By setting $\bar{r}_{\ell,j} = 1 - \left(1 - e^{-\lambda_{\ell,j}\beta_{\ell,j}\tau_{\ell}^{p-(\beta_{\ell,j}-1)}T}\right)^{M_\ell}$, for each feasible solution $\hat{\mathbf{y}}$ of Model (107), there is a solution $\tau_\ell = \sum_{p=1}^{P} \tau_\ell^p \hat{y}_\ell^p$ in Model (106) with the same objective value. □

Theorem 19. Define $\bar{\bar{r}}_{\ell,j}^{p} = 1 - \left(1 - e^{-\lambda_{\ell,j}\beta_{\ell,j}\tau_{\ell}^{p+1-(\beta_{\ell,j}-1)}T}\right)^{M_\ell}$ for all $\ell = 1,\ldots,N,\ j = 1,\ldots,|\bar{D}|,\ p = 1,\ldots,P$, then $R^{\mathrm{OA}}(\bar{\bar{\mathbf{r}}},T) \geq E(\bar{D})$. (That is, UB-MD provides an upper bound for the outer approximation problem).

Proof. Let $A(\hat{\tau})$ denote the optimal objective value of Model (106) when $\tau = \hat{\tau}$ is fixed. Define $\tau_\ell^*,\ \ell = 1,\ldots,N$, as an optimal solution for Model (106). Find the index $\hat{P}_\ell \in \{1,\ldots,P\}$ for each subsystem ℓ such that $\tau_\ell^{\hat{P}_\ell} \leq \tau_\ell^* < \tau_\ell^{\hat{P}_\ell+1}$, and set $\hat{\tau}_\ell = \tau_\ell^{\hat{P}_\ell+1}$. The solution $y_\ell^{\hat{P}_\ell} = 1,\ \ell = 1,\ldots,N$ is a feasible solution to UB-MD, with objective value equal to $A(\hat{\tau})$. The value $A(\hat{\tau})$ is no less than $R^\star = A(\tau^\star)$ since $A(\tau)$ is nondecreasing in τ. Hence, $R^\star \leq A(\hat{\tau}) \leq R^{\mathrm{OA}}(\bar{\bar{\mathbf{r}}},T)$. □

Algorithm 3 is used to solve the outer approximation Model (106), where P_{sub} is the initial number of discrete points used to populate Model (107) and $\varepsilon_{\mathrm{sub}}$ is the desired optimality tolerance. In this algorithm, $\tau^\star$ defines the incumbent solution. Given a desired optimality tolerance ε, Algorithm 4 provides the resulting cutting-plane algorithm for robust allocation of testing time in a series-parallel system. For the sake of brevity, we have not provided a convergence proof of Algorithm 4, but we now provide a sketch: Arguments similar to those used in Theorem 15 can

be used to bound the number of iterations of the cutting-plane algorithm, and the number of steps per iteration can be bounded using arguments similar to those in Chapter 2, which establish finite convergence of discretize-and-refine algorithms for a related class of problems.

Algorithm 3 The outer approximation algorithm (calculate an upper bound for $R^\star$ within ε_{sub} tolerance).

function OUTERAPPROXIMATION (ε_{sub}, P_{sub})
  for $\ell = 1,\ldots,N$ **do**
    Set $\tau_\ell^{\min} \leftarrow \tau_\ell^0$ and $\tau_\ell^{\max} \leftarrow \min_{k=1}^{K} \left\{ \left(b_k - \Sigma_{\ell'=1}^{N} c_{\ell'}^k \tau_{\ell'}^0 + c_\ell^k \tau_\ell^0\right) / c_\ell^k \right\}$
    for $p = 1,\ldots,P_{\text{sub}}$ **do**
      Set $\tau_\ell^p \leftarrow \tau_\ell^{\min} + (p-1)(\tau_\ell^{\max} - \tau_\ell^{\min})/P_{\text{sub}}$
    end for
  end for
  Set $LB \leftarrow 0$, $UB \leftarrow 1$, and $P \leftarrow P_{\text{sub}}$
  If $UB - LB < \varepsilon_{\text{sub}}$ **then return** $\tau^\star$ ▷ $\tau^\star$ is ε-optimal
  Solve the LB-MD model with optimal solution $\hat{\tau}$ and objective value $R^{\text{OA}}(\bar{\mathbf{r}}, T)$
  If $R^{\text{OA}}(\bar{\mathbf{r}}, T) > LB$ **then set** $LB \leftarrow R^{\text{OA}}(\bar{\mathbf{r}}, T)$, $\tau^\star \leftarrow \hat{\tau}$
  Solve the UB-MD model with optimal solution $\bar{P}_\ell \in \{1,\ldots,P\}$, $\ell = 1,\ldots,N$ and objective value $R^{\text{OA}}(\bar{\bar{\mathbf{r}}}, T)$
  If $R^{\text{OA}}(\bar{\bar{\mathbf{r}}}) < UB$ **then set** $UB \leftarrow R^{\text{OA}}(\bar{\bar{\mathbf{r}}}, T)$
  for $\ell = 1,\ldots,N$ **do**
    Add a new point $\tau_\ell^{P+1} \equiv 0.5\left(\tau_\ell^{\bar{P}_\ell} + \tau_\ell^{\bar{P}_\ell+1}\right)$
  end for
  Renumber the points such that $\tau_\ell^1 < \tau_\ell^2 < \cdots < \tau_\ell^{P+1}$
  set $P \leftarrow P+1$
  Go to Line 9
end function

Algorithm 4 Cutting plane (calculate $R^\star$ within ε tolerance).

function CUTTINGPLANE (V, b_v, Φ, ε, ε_{sub}, P_{sub})
  Initialize $\bar{D} \subset U_\Phi$. **Set** $LB \leftarrow 0$ and $UB \leftarrow 1$.
  Solve OUTERAPPROXIMATION(ε_{sub}, P_{sub}) with objective (L_O, U_O) and solution $\hat{\tau}$.
  Set $UB \leftarrow U_O$.
  Solve SEPARATIONALGORITHM(ε_{sub}, P_{sub}, $\hat{\tau}$) with objective (L_S, U_S) and solution $\left(\hat{\gamma}^L, \hat{\gamma}^B\right)$.
  If $L_S > LB$ **then set** $LB \leftarrow e^{L_S}$ and $\tau^\star \leftarrow \hat{\tau}$.
  If $UB - LB < \varepsilon$ **then return** $\tau^\star$ ▷ $\tau^\star$ is ε-optimal
  Set $\bar{D} = \bar{D} \cup \left(\hat{\gamma}^L, \hat{\gamma}^B\right)$. Go to Line 3.
end function

3.5 Computational Results

This section provides a study of computational results. For series systems, the cutting-plane algorithm is coded in MATLAB and CVX solver is used to solve each convex optimization problem. For series-parallel systems, C++ is used to code the cutting-plane algorithm and CPLEX 12.4 is invoked to solve each individual integer program. For all instances, a server with an Intel core i12 with 2.9 GHz and 12 GB RAM is used. We compare the robust and deterministic approaches via sensitivity analysis for a set of numerical examples in Section 3.5.1, demonstrate the use of confidence intervals to construct uncertainty sets in Section 3.5.2, study the effect of the system structure on testing time allocation in Section 3.5.3, and analyze the computational efficiency of the cutting-plane algorithm in Section 3.5.4. Although we have neither claimed nor demonstrated that that the robust optimization approach always performs better than available deterministic approaches, Sections 3.5.1–3.5.2 demonstrate that the robust approach is favorable for what we believe to be a realistic set of instances.

Based upon preliminary experiments, we identified that smaller values of Φ (e.g., $\Phi \in \{0.3, 0.5\}$ when $N \leq 6$) tend to provide more competitive results from the robust optimization model. We believe this is because the scenarios that result when one or more parameters are allowed to take on their worst-case values are, in some cases, too significant of a deterrent. For instance, a subsystem ℓ with $\bar{\beta}_\ell + \sigma_\ell^B$ close to 1 has, in the worst scenario, almost no growth at all. This discourages solutions in which subsystem ℓ is tested in any significant amount. Because of this observation, we have utilized small values of Φ in Sections 3.5.1–3.5.2, where we are demonstrating value of the robust approach against available deterministic approaches.

3.5.1 Comparing Robust and Deterministic Solutions via Sensitivity Analysis

To demonstrate the value of incorporating uncertainty into the resource allocation model, we begin by describing a procedure that can be used to compare the results of alternative testing strategies under different combinations of realized parameter values. We first illustrate the approach graphically for two-subsystem examples in which only two of the four parameters $\{\lambda_1, \lambda_2, \beta_1, \beta_2\}$ are

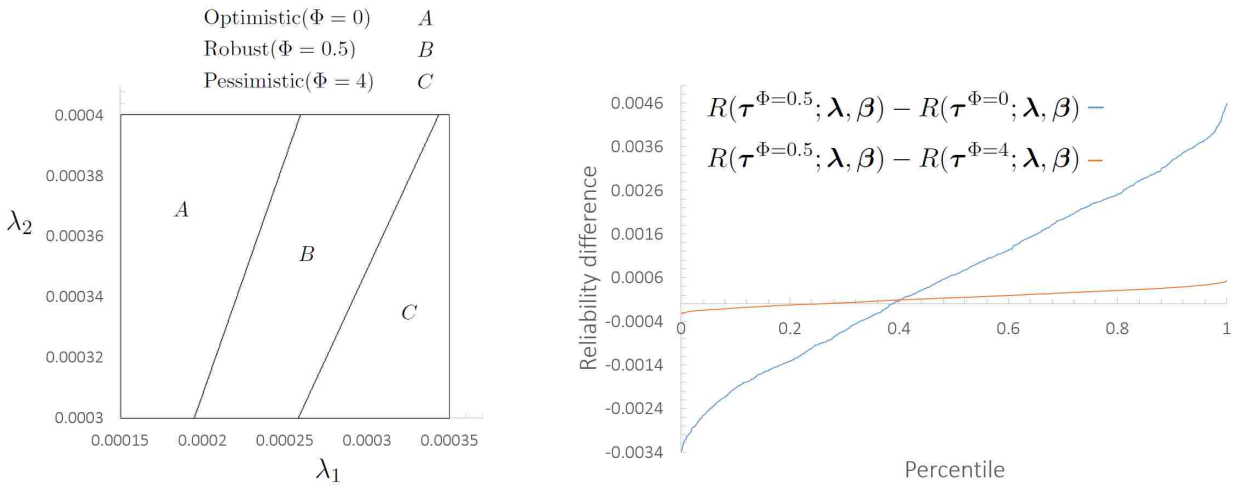
subject to uncertainty. Using this approach, we then summarize results for larger systems with more decision variables and parameters subject to uncertainty.

The approach utilized in this system considers a fixed set of uncertainty intervals $[\bar{\lambda}_\ell, \bar{\lambda}_\ell + \sigma_\ell^L]$ and $[\bar{\beta}_\ell, \bar{\beta}_\ell + \sigma_\ell^B]$ associated with each subsystem $\ell = 1, \ldots, N$. Hereafter, we will refer to the Cartesian product of these $2N$ intervals as the *uncertainty box*. We generate testing plans $\tau^{\Phi=0}$, $\tau^{\Phi=0.5}$, and $\tau^{\Phi=2N}$ by solving the robust optimization model corresponding to $\Phi \in \{0, 0.5, 2N\}$ over the uncertainty box. (Note: The solutions $\tau^{\Phi=0}$ and $\tau^{\Phi=2N}$ are respectively the optimistic and pessimistic solutions obtained by solving the corresponding deterministic model.) We then compare the robust solution $\tau^{\Phi=0.5}$ to the corresponding deterministic solutions over potentially realized parameters in the uncertainty box.

3.5.1.1 Series System with $N = 2$, Uncertainty Only in λ

Consider a series system that contains two subsystems. Suppose the true β-values are known with certainty such that $\beta_1 = \bar{\beta}_1 = 0.65$ and $\beta_2 = \bar{\beta}_2 = 0.6$ and $\sigma_1^B = \sigma_2^B = 0$. Suppose uncertainty in λ is described by $\bar{\lambda}_1 = 0.00015$, $\sigma_1^L = 0.0002$, $\bar{\lambda}_2 = 0.0003$ and $\sigma_2^L = 0.0001$, and the remaining parameters are given by $T = 18{,}250$, $K = 1$, $b_1 = 8{,}760$, $c_\ell^1 = 1$ and $\tau_\ell^0 = 1$, $\ell \in \{1, 2\}$. For this region, we generated $\tau^{\Phi=0}$, $\tau^{\Phi=0.5}$ and $\tau^{\Phi=4}$. (Note, however, that the robust solutions for $2 \leq \Phi \leq 4$ are equal because there is no uncertainty in β.) The testing times are $(\tau_1^{\Phi=0}, \tau_2^{\Phi=0}) = (3860.4, 4899.6)$ for the optimistic solution, $(\tau_1^{\Phi=0.5}, \tau_2^{\Phi=0.5}) = (4671.5, 4088.5)$ for the robust solution, and $(\tau_1^{\Phi=4}, \tau_2^{\Phi=4}) = (4748.7, 4011.3)$ for the pessimistic solution. Using these solutions, Figure 5(a) shows the λ-values in the uncertainty box under which each of the three solutions yields the maximum reliability.

The point $(\lambda_1, \lambda_2) = (0.00015, 0.0003)$ in Figure 5(a), falls in region A, thus indicating that the optimistic model's solution would yield a better reliability for these realized values of λ_1 and λ_2. This is to be expected as the optimization model corresponding to $\Phi = 0$ is deterministic under $\lambda_1 = \bar{\lambda}_1 = 0.00015$ and $\lambda_2 = \bar{\lambda}_2 = 0.0003$. Similarly, it is not surprising that $(\lambda_1, \lambda_2) = (0.00035, 0.0004)$ falls in region C, (indicating that the pessimistic model yields superior performance) because the



(a) **(b)**

Figure 5: (a) Uncertainty box partitioned according to the value of Φ yielding the maximum reliability; and (b) percentile plot of reliability difference between the robust and deterministic solutions for the first instance.

optimization model corresponding to $\Phi = 4$ is deterministic under the assumption that $\lambda_1 = \bar{\lambda}_1 + \sigma_1^L = 0.00035$ and $\lambda_2 = \bar{\lambda}_2 + \sigma_2^L = 0.0004$.

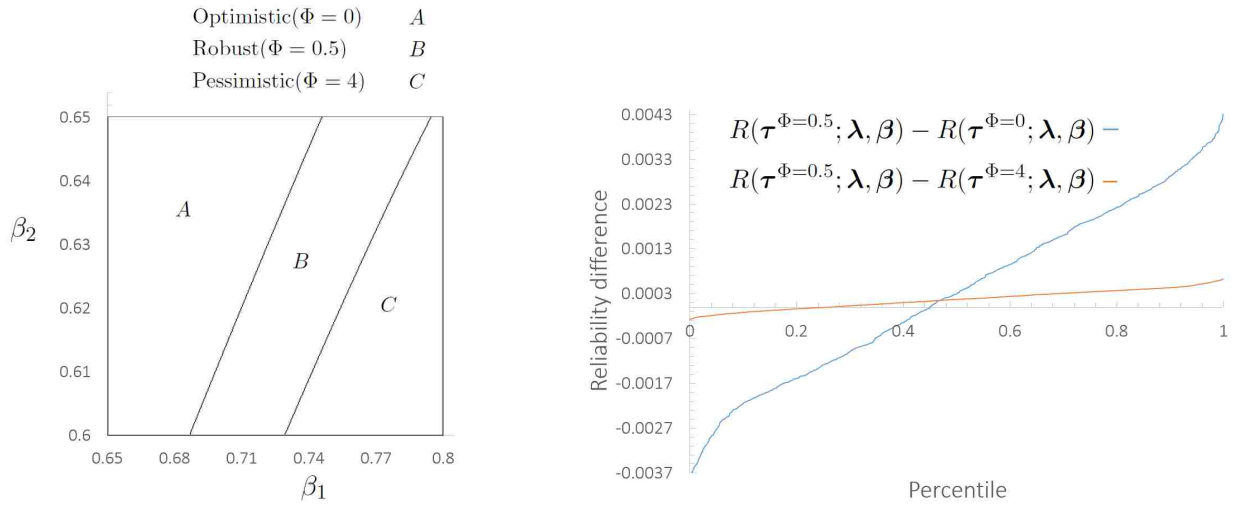
We now summarize the analysis for this example in such a way that can be repeated to determine if the robust optimization model's solutions are a favorable alternative towards using either the optimistic or pessimistic deterministic optimization models. One measure of interest in comparing solutions is the *pairwise hypervolume proportion*, the proportion of the uncertainty box in which the robust solution yields higher reliability than either the optimistic or pessimistic deterministic solution. The pairwise hypervolume proportions for the optimistic and pessimistic solutions are respectively 61.87% (corresponding to Regions B and C in Figure 5(a)) and 74.54% (corresponding to regions A and B). In other words, if the λ-values are selected uniformly from the uncertainty box, the robust solution generates higher reliability than the optimistic (pessimistic) solution 61.87% (74.54%) of the time.

Figure 5(b) depicts a percentile plot of the reliability differences between the robust solution and the optimistic and pessimistic solutions. We sample $1,000$ realizations of λ uniformly within the uncertainty box and calculate the reliability associated with optimistic, robust and pessimistic solutions. We then calculate the differences between the robust solution and each of the optimistic and pessimistic solutions, sort the reliability differences, and plot the differences in Figure 5(b)—

with the smallest difference on the far left, and the largest difference on the far right. The horizontal intercepts occur for the optimistic and pessimistic comparison (approximately) at $(1-0.6187)$ and $(1-0.7454)$, respectively corresponding to the pairwise hypervolume proportions detailed in the previous paragraph. In this instance, the robust solution's reliability may exceed that of the optimistic (pessimistic) solution by as much as 0.00472 (0.00053), but the optimistic (pessimistic) solution stands to exceed the reliability of the robust solution by no more than 0.00358 (0.00023).

3.5.1.2 Series System with $N = 2$, Uncertainty Only in β

We now repeat the analysis of the previous section, but this time in the case where λ is known and β is subject to uncertainty. Specifically, let $\lambda_1 = \bar{\lambda}_1 = 0.00015$ and $\lambda_2 = \bar{\lambda}_2 = 0.0003$ such that their associated uncertainty intervals have a length of zero. The uncertainty underlying parameters are $\bar{\beta}_1 = 0.65$, $\sigma_1^B = 0.15$, $\bar{\beta}_2 = 0.6$ and $\sigma_2^B = 0.05$. Similar to the previous instance, we set $T = 18,250$, $K = 1$, $b_1 = 8,760$, $c_\ell^1 = 1$ and $\tau_\ell^0 = 1,\ \ell \in \{1,2\}$. The testing times are now $(\tau_1^{\Phi=0}, \tau_2^{\Phi=0}) = (3860.4, 4899.6)$ for the optimistic solution, $(\tau_1^{\Phi=0.5}, \tau_2^{\Phi=0.5}) = (4656.5, 4103.3)$ for the robust solution, and $(\tau_1^{\Phi=4}, \tau_2^{\Phi=4}) = (4743.5, 4016.5)$ for the pessimistic solution. Because the underlying parameters of the optimistic solution for this instance and the previous instance are equal, the testing times of both instances for the optimistic solution are also equal. Figure 6(a) specifies the policy that provides the highest reliability when β-values are in the uncertainty box. The pairwise hypervolume proportions for the optimistic and pessimistic solutions are 56.23% (corresponding to regions B and C) and 75.13% (corresponding to regions A and B). Similar to the previous instance, the robust solution never yields the smallest reliability when it is compared with both optimistic and pessimistic solutions in the uncertainty box. From Figure 6(b), for realizations of β in the uncertainty box, the robust solution's reliability exceeds the optimistic (pessimistic) solution's reliability by as much as 0.00439 (0.00065) while the optimistic (pessimistic) solution may exceed the robust solution in reliability by as much as 0.00398 (0.00029).



(a) **(b)**

Figure 6: (a) Uncertainty box partitioned according to the value of Φ yielding the maximum reliability; and (b) percentile plot of reliability difference between robust and deterministic solutions for the second instance.

3.5.1.3 Series System with $N = 2$, Uncertainty in Both λ and β

We now analyze the performance of the robust solution in comparison with the optimistic and pessimistic deterministic solutions when the number of uncertain parameters increases. Figure 7 depicts the approximated percentile plot (now generated based on $10,000$ sampled realizations of the uncertain parameters since the uncertainty box has a higher dimension) for a series system with two subsystems with $T = 18,250$, $K = 1$, $b_1 = 8,760$, $c_\ell^1 = 1$ and $\tau_\ell^0 = 1$, $\ell \in \{1,2\}$. None of the parameter values are known in this instance, and the parameter uncertainty is given by $\bar{\lambda}_1 = 0.00015$, $\sigma_1^L = 0.0002$, $\bar{\lambda}_2 = 0.0003$, $\sigma_2^L = 0.0001$, $\bar{\beta}_1 = 0.65$, $\sigma_1^B = 0.15$, $\bar{\beta}_2 = 0.6$ and $\sigma_2^B = 0.05$. (That is, we now have a hybrid of the instances from Sections 3.5.1.1–3.5.1.2 in which the uncertainty intervals for λ match the uncertainty interval from Section 3.5.1.1, the uncertainty intervals for β match the uncertainty intervals from Section 3.5.1.2, and all other parameters are common across all three instances.) The allocation of testing times across the subsystems for the optimistic solution is $(\tau_1^{\Phi=0}, \tau_2^{\Phi=0}) = (3860.4, 4899.6)$, for the robust solution is $(\tau_1^{\Phi=0.5}, \tau_2^{\Phi=0.5}) = (4656.7, 4103.3)$, and for the pessimistic solution is $(\tau_1^{\Phi=4}, \tau_2^{\Phi=4}) = (5656.9, 3103.1)$.

The pairwise hypervolume proportions for the optimistic and pessimistic solutions are respectively 82.48% and 65.78%, respectively. The robust solution's reliability may now exceed the

optimistic (pessimistic) solution's by as much as 0.00981 (0.01317) and the potential magnitude by which the optimistic (pessimistic) solution's reliability could exceed the robust solution's reliability remains relatively small at 0.00581 (0.00735).

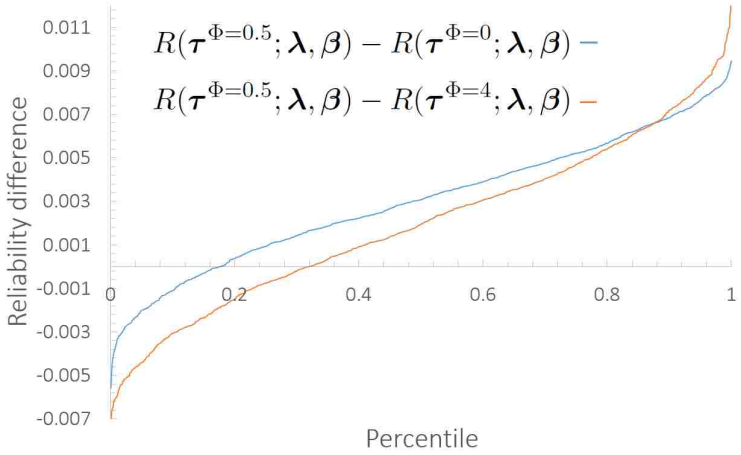


Figure 7: Percentile plot of reliability difference between robust and deterministic solutions for the third instance.

3.5.1.4 Series System with $N = 6$, Uncertainty in Both λ and β

We now consider a series system with 6 subsystems. The parameters of this instance are represented in Table 7 with the exception that $M_\ell = 1,\ \ell = 1,\ldots,N$. We set $T = 4,380$ and $b_1 = 8,760$. The approximate percentile plot is given in Figure 8(a). For this instance, the pairwise hypervolume proportion for the optimistic and pessimistic solutions are 86.29% and 71.61%. The robust solution's reliability may, in this case, exceed that of the optimistic (pessimistic) solution by as much as 0.01163 (0.00656) and may be exceeded by as much as 0.00217 (0.00609). The testing times for each method are presented in Table 8. In this case, the testing times of the robust solution for subsystems 2, 4 and 6 fall between the pessimistic and optimistic solutions.

Table 7: Parameters for the series instance with 6 subsystems

ℓ	$\bar{\lambda}_\ell$	σ_ℓ^L	$\bar{\beta}_\ell$	σ_ℓ^B	M_ℓ	ℓ	$\bar{\lambda}_\ell$	σ_ℓ^L	$\bar{\beta}_\ell$	σ_ℓ^B	M_ℓ
1	0.00008	0.00003	0.65	0.07	3	3	0.0004	0.00007	0.75	0.02	2
3	0.0002	0.0001	0.68	0.05	1	4	0.00025	0.00004	0.7	0.04	3
5	0.00004	0.00003	0.65	0.1	2	6	0.00006	0.0002	0.6	0.15	1
$K = 1$ $\quad c_\ell^1 = 1 \quad \tau_\ell^0 = 1,\ \ell \in \{1,\ldots,6\}$											

Table 8: Testing times for the series instance with 6 subsystems

	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6
Optimistic ($\Phi = 0$)	692.1	3625.4	1554.3	2010.5	414.2	463.5
Robust ($\Phi = 0.5$)	658.3	3434.8	1476.8	1908.7	394.0	887.4
Pessimistic ($\Phi = 12$)	707.5	2707.5	1612.6	1633.0	544.3	1555.1

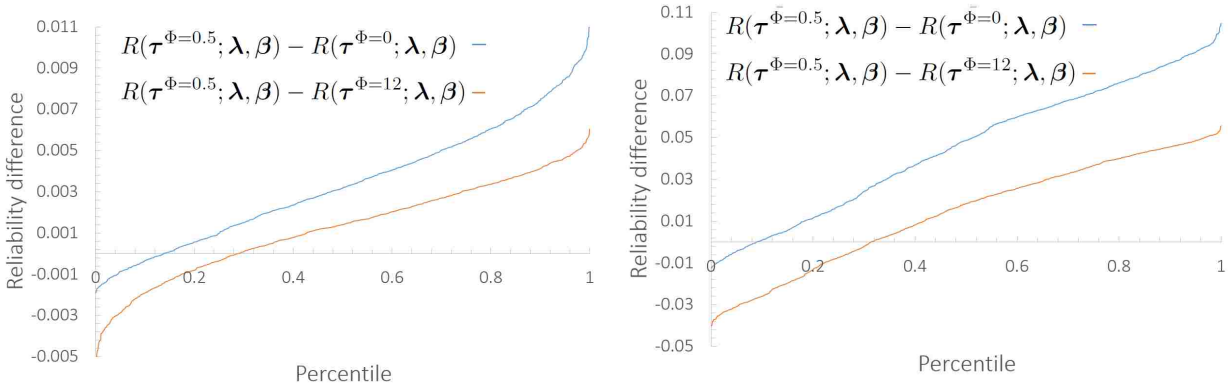


Figure 8: Percentile plot of reliability difference between robust and deterministic solutions (a) for the series instance with 6 subsystems and (b) for the series-parallel instance.

3.5.1.5 Series-Parallel System with $N = 6$, Uncertainty in Both λ and β

We now perform the same set of analyses for a series-parallel system with 6 subsystems. The parameters are presented in Table 7, and we consider $b_1 = 8,760$ and $T = 18,250$ for this instance. The pairwise hypervolume proportions for the optimistic and pessimistic deterministic approaches are 83.59% and 68.28%, respectively. We also present the percentile plot of the reliability difference between the robust solution and the optimistic and pessimistic solutions in Figure 8(b). In this case, the robust solution's reliability may now exceed that of the optimistic (pessimistic) solution by as much as 0.02095 (0.01475) and may not be exceed by more than 0.00563 (0.01101). Thus, in moving from the series to the series-parallel instance, the robust optimization approach seems even more favorable as compared to deterministic methods. The solutions generated by each method are presented in Table 9. The testing times for subsystems 2, 3 and 6 lie between the deterministic solutions.

Table 9: Testing times for the series-parallel instance

	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6
Optimistic ($\Phi = 0$)	135.9	4250.9	2476.3	1025.8	153.4	717.5
Robust ($\Phi = 0.5$)	128.9	3925.0	2276.6	961.0	144.6	1321.9
Pessimistic ($\Phi = 12$)	225.2	2982.6	2136.4	1041.6	304.1	2069.9

3.5.2 Generating Uncertainty Intervals Based on Preliminary Test Data

We now demonstrate, using the three-subsystem instance defined in Table 10, how one may construct uncertainty intervals based on preliminary test data. In this set of results, we simulate the

Table 10: Parameters for three-subsystem series instance

λ_1	β_1	λ_2	β_2	λ_3	β_3	T	K	c_ℓ^1	b_1
0.01	0.65	0.02	0.6	0.025	0.7	120	1	1	30,660

underlying failure/growth process (i.e., the nonhomogenous Poisson process described by Equation (1)) in order to generate initial failure data that may be used in planning future tests. Given this data, we compare the robust approach against deterministic approaches for generating test plans.

Initially, we simulate the AMSAA failure process (using the underlying λ_ℓ- and β_ℓ-values) for $\tau_\ell^0 = \tau^{\text{int}}$ hours for each $\ell = 1,\ldots,N$. (We consider the values $\tau^{\text{int}} \in \{6570, 7300\}$ in our experiments.) Based upon the results of the simulation, for each $\ell = 1,\ldots,N$, we derive maximum likelihood estimates (denoted $\hat{\beta}_\ell$ and $\hat{\lambda}_\ell$) of λ_ℓ and β_ℓ as

$$\hat{\beta}_\ell = \frac{n_\ell}{n_\ell \ln(\tau_\ell^0) - \sum_{i=1}^{n_\ell} \ln(t_{\ell,i})}, \tag{111a}$$

$$\hat{\lambda}_\ell = \frac{n_\ell}{\left(\tau_\ell^0\right)^{\hat{\beta}_\ell}}, \tag{111b}$$

for each $\ell = 1,\ldots,N$ and in which n_ℓ is number of failures for subsystem $\ell = 1,\ldots,N$ in the simulation, and $t_{\ell,i},\ i = 1,\ldots,n_\ell$ is the i^{th} failure time for subsystem $\ell = 1,\ldots,N$. We then solve the deterministic allocation model (i.e., Model (61)) assuming $\lambda = \hat{\lambda}$ and $\beta = \hat{\beta}$ to allocate test times to the three subsystems. We refer to this deterministic approach as the *max-likelihood approach.* Using the test times $\tau_\ell^\star,\ \ell = 1,\ldots,N$, that result from the max-likelihood allocation model, we then evaluate $R(\tau^\star;\lambda,\beta)$—the true reliability after implementing the optimization model's rec-

ommended test times—and report this value for comparison. Each subsystem's failure times are generated based on the homogenous Poisson process by the failure rate given in Equaton (1), and then used to estimate $\hat{\lambda}$ and $\hat{\beta}$. Because the estimates $\hat{\lambda}$ and $\hat{\beta}$ are random, we replicate this procedure 100 times (using the same underlying parameters given in Table 10).

To demonstrate the effect of our robust model, we repeat the above process using the robust model instead of the deterministic model to generate $\tau^\star$. Specifically, given the results of the initial τ^{int} hours of testing completed on each subsystem, we first construct $100(1-\alpha)\%$ two-sided confidence intervals (using the Fisher Matrix bound [42]) on each λ_ℓ and β_ℓ, $\ell = 1,\ldots,N$, as

$$\lambda_\ell \in \left[\hat{\lambda}_\ell e^{z\alpha/2\sqrt{Var(\hat{\lambda}_\ell)}/\hat{\lambda}_\ell}, \hat{\lambda}_\ell e^{-z\alpha/2\sqrt{Var(\hat{\lambda}_\ell)}/\hat{\lambda}_\ell}\right], \tag{112a}$$

$$\beta_\ell \in \left[\hat{\beta}_\ell e^{z\alpha/2\sqrt{Var(\hat{\beta}_\ell)}/\hat{\beta}_\ell}, \hat{\beta}_\ell e^{-z\alpha/2\sqrt{Var(\hat{\beta}_\ell)}/\hat{\beta}_\ell}\right], \tag{112b}$$

in which z_α is the value of Normal distribution with lower tail probability α, and

$$Var(\hat{\lambda}_\ell) = \frac{\hat{\lambda}_\ell^2\left(n_\ell - \hat{\lambda}_\ell n_\ell^{\hat{\beta}_\ell}\ln(n_\ell)^2\hat{\beta}_\ell^2\right)}{n_\ell^2 - \hat{\lambda}_\ell\hat{\beta}_\ell^2 n_\ell^{\hat{\beta}_\ell+1}\ln(n_\ell)^2 - (\hat{\lambda}_\ell\hat{\beta}_\ell n_\ell^{\hat{\beta}_\ell}\ln(n_\ell))^2}, \tag{113a}$$

$$Var(\hat{\beta}_\ell) = \frac{n_\ell\hat{\beta}_\ell^2}{n_\ell^2 - \hat{\lambda}_\ell\hat{\beta}_\ell^2 n_\ell^{\hat{\beta}_\ell+1}\ln(n_\ell)^2 - (\hat{\lambda}_\ell\hat{\beta}_\ell n_\ell^{\hat{\beta}_\ell}\ln(n_\ell))^2}. \tag{113b}$$

We then define an instance of the robust optimization model (i.e., Model (69)) by letting the confidence intervals (112) play the role of Equation (65) in the construction of the uncertainty set U_Φ. We repeat this for all combinations of the values $\alpha \in \{0.6, 0.8\}$ and $\Phi \in \{0.3, 0.5\}$. For each of these combinations, we run 100 replications (reusing the same values of n_ℓ and $t_{\ell,i}$ generated in replicating the max-likelihood approach) because the uncertainty intervals are also random due to depending on the simulated failure data. Because $0 < \beta_\ell < 1$ in the AMSAA model, we use 0.99 in place of $\bar{\beta}_\ell + \sigma_\ell^B$ in the robust optimization model if $\bar{\beta}_\ell + \sigma_\ell^B \geq 1$. Likewise, we truncate the values of the lower limit $\bar{\beta}_\ell$ and maximum likelihood estimate $\hat{\beta}_\ell$ at 0.99 when they would other-

wise appear in an optimization model with value at least 1. Additionally, we note that (112)–(113) are ill-defined when zero failures are observed in a given subsystem and similarly, the maximum likelihood estimates (111) are not meaningful for the purposes of allocating resources. In our experiments, we handle this by removing replications in which some subsystem experiences zero failures (this happens only once out of a total of 200 runs). In order to enhance the resource allocation approach for the case of limited failure data, it may be possible to extend the results herein to account for censored data or to incorporate accelerated testing. We have left this for future research.

The values of α defined in the previous paragraph were chosen based upon a set of preliminary experiments, which revealed that selecting a smaller value of α results in an extremely conservative uncertainty set that incorporates a high frequency of parameter combinations that are unlikely to occur in practice. Furthermore, such an uncertainty set may tend to include values of β_ℓ that are close to 1, and our experience has suggested that this case—see the discussion in the second paragraph of Section 3.5—reduces the overall solution quality as compared to using available deterministic estimates. Utilizing a more conservative uncertainty set (i.e., with a larger α) seems to mitigate this issue and yield higher quality solutions overall.

In order to compare against what would be possible prior to the robust model, we compare against several deterministic optimization approaches. The robust optimization model corresponding to $\Phi = 2N = 6$ allows all parameters to take on their worst-case values (i.e., $\lambda_\ell = \bar{\lambda}_\ell + \sigma_\ell^L$ and $\beta_\ell = \bar{\beta}_\ell + \sigma_\ell^B$, $\forall \ell = 1, \ldots, N$) and thus corresponds to the pessimistic deterministic approach, which we refer to as the *pessimistic approach*. The robust optimization model also becomes deterministic when $\Phi = 0$ as all parameters take their optimistic values (i.e., $\lambda_\ell = \bar{\lambda}_\ell$ and $\beta_\ell = \bar{\beta}_\ell$, $\forall \ell = 1, \ldots, N$). We refer to this as the *optimistic approach*. We also compare against the deterministic model where all γ_ℓ^L and γ_ℓ^B are fixed equal to 0.5, and refer to the resulting approach as *midpoint approach*. For each robust approach (i.e., for $\Phi = 0.3$ and $\Phi = 0.5$) and each deterministic approach (i.e., max-likelihood, pessimistic, optimistic and midpoint), we record the true reliability that results from each of the 100 replications.

The results of our experiments are summarized in Table 11. In Table 11, R>L, R>P, R>O, and R>M represent the number of times from 100 replications in which the robust approach respectively yielded higher reliability than the max-likelihood, pessimistic, optimistic, and midpoint approaches. Regardless of the value of α and τ^{int}, both of the robust treatments are preferable to all of the deterministic treatments in more than half of the replications.

Table 11: Summary of results for the series instance

		Robust ($\Phi = 0.3$)				Robust ($\Phi = 0.5$)			
τ^{int}	α	R>L	R>O	R>M	R>P	R>L	R>O	R>M	R>P
6,570	0.6	54	58	77	75	55	62	71	74
	0.8	62	57	65	75	58	56	62	74
7,300	0.6	67	76	81	87	68	76	79	87
	0.8	65	78	64	76	71	79	65	80

Figure 9 compares the reliability percentiles of the robust and deterministic approaches for $\tau^{\text{int}} \in \{6570, 7300\}$ and $\alpha \in \{0.6, 0.8\}$. Across 100 replications, we obtain 100 different system reliabilities for each method. We sort those reliabilities, and plot them in Figure 9, in which the smallest is in the far left and the largest is in the far right. Figure 9 suggests that the robust approaches are particularly effective in comparison to the deterministic approaches in the lower percentiles although the optimistic and max-likelihood approaches yield similar reliabilities as robust approaches for percentiles above 0.5.

Table 12 compares the average testing times (across the 100 replications) of the robust and the deterministic approaches for $\tau^{\text{int}} \in \{6570, 7300\}$ and $\alpha \in \{0.6, 0.8\}$. The testing times shown in Table 12 are the averages of 100 replications. One observation from Table 12 is that τ_3 is larger for the robust approaches than the deterministic approaches. We conjecture this is related to the fact that λ_3 is larger than either λ_1 or λ_2, and β_3 is larger than either β_1 or β_2, so the third subsystem needs more testing time to reach a particular reliability. The robust solution hedges against potentially larger values of λ_3 and β_3 by allocating more testing time.

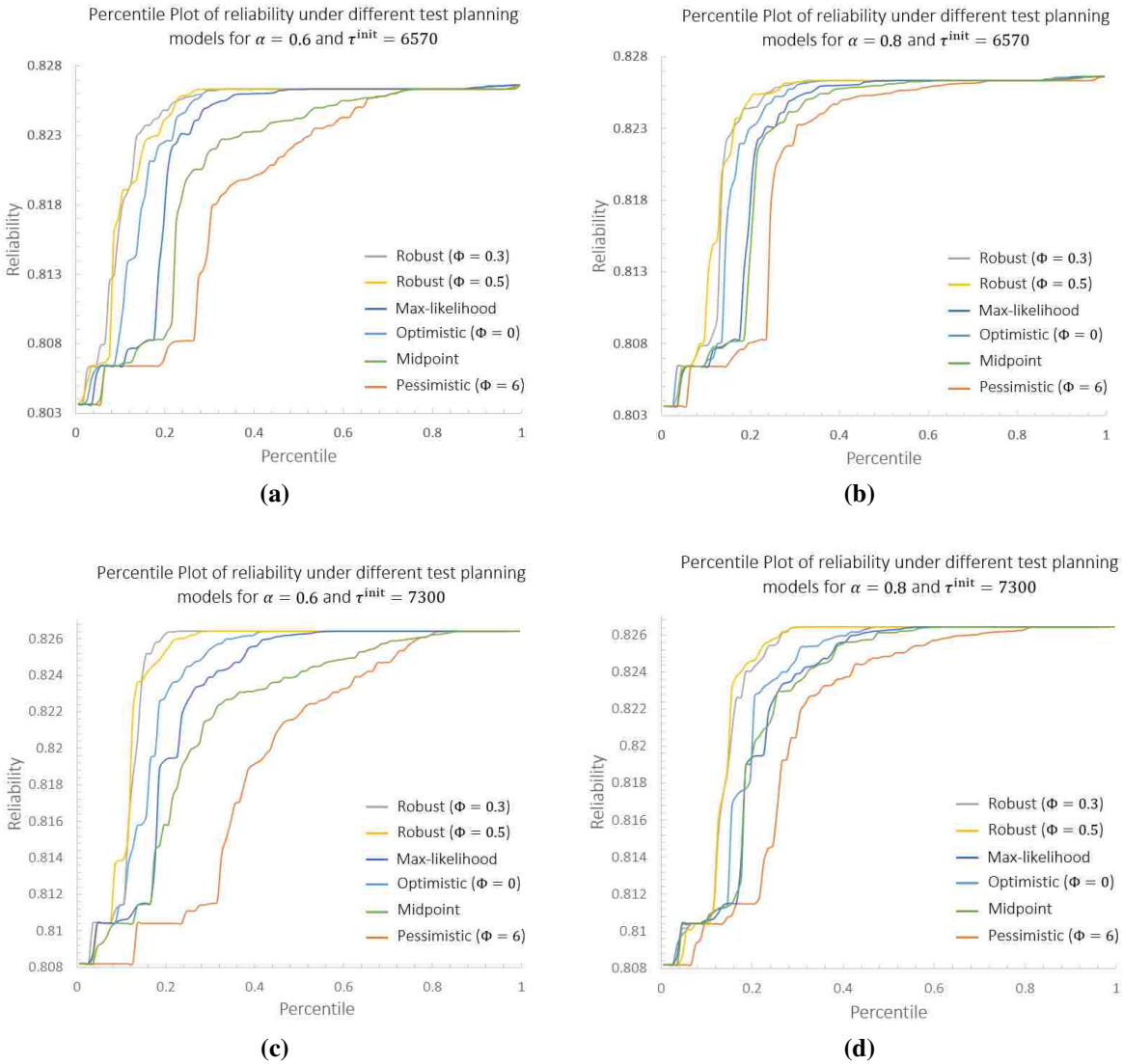


Figure 9: Percentile plot of reliability under different test planning models.

3.5.3 Effect of System Structure on Resource Allocation

In order to compare the effect of the system structure on testing time allocation, we consider two sets of instances. Each instance considers a system with two subsystems. The first set of instances consists of one component in the first subsystem and $s \in \{1,\ldots,6\}$ components in the second subsystem, and the second set of instances consists of $s \in \{1,\ldots,6\}$ components in the first subsystem and one component in the second subsystem. The parameters for the instances are provided in Table 13, and we use $P_{\text{sub}} = 20,000$ and $\varepsilon_{\text{sub}} = 1 \times 10^{-4}$ in the series-parallel algorithm.

Table 12: Average testing time allocation for the series instance

τ^{int}	α		Max-likelihood	Optimistic	Midpoint	Pessimistic	$\Phi = 0.3$	$\Phi = 0.5$
6,570	0.6	τ_1	7783.6	7296.0	8762.9	8643.2	7425.5	7476.3
		τ_2	8656.6	8207.4	9200.9	9903.6	7646.1	7545.7
		τ_3	14219.7	15156.6	12696.2	12113.2	15588.4	15638.0
	0.8	τ_1	7783.6	7372.6	8050.6	8364.5	7362.2	7375.7
		τ_2	8656.6	8347.5	8710.1	9171.0	8009.3	7921.2
		τ_3	14219.7	14939.9	13899.4	13124.4	15288.5	15363.1
7,300	0.6	τ_1	8354.8	8064.5	8824.5	9314.0	7737.1	7824.1
		τ_2	8827.2	8505.5	9539.9	10093.8	8136.9	8147.1
		τ_3	13478.0	14089.9	12295.5	11252.3	14786.0	14688.8
	0.8	τ_1	8354.8	8258.2	8402.2	8811.8	8045.7	8035.3
		τ_2	8827.2	8571.1	8886.9	9432.3	8197.7	8128.5
		τ_3	13478.0	13830.7	13370.9	12416.0	14416.5	14496.2

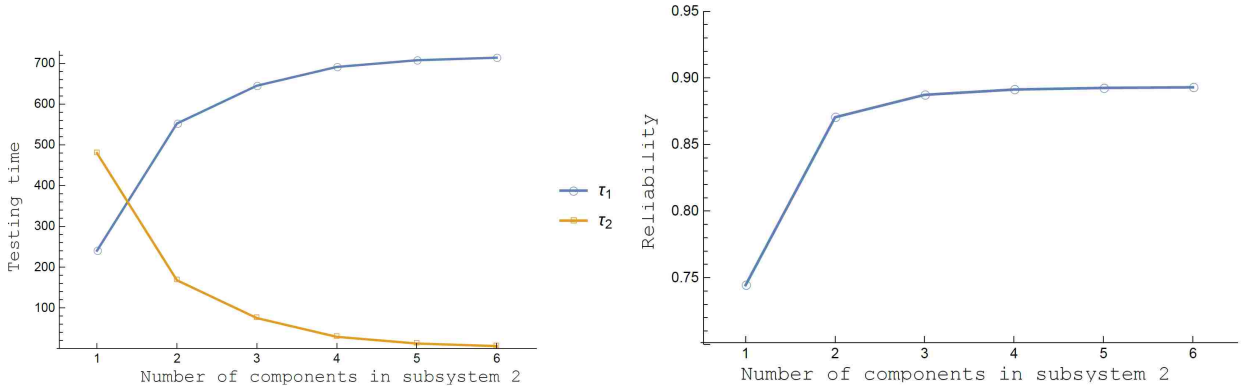
Table 13: Parameters for 11 instances that study the effect of system structure in resource allocation

$\bar{\lambda}_1$	σ_1^L	$\bar{\beta}_1$	σ_1^B	$\bar{\lambda}_2$	σ_2^L	$\bar{\beta}_2$	σ_2^B	Φ	T	K	c_ℓ^1	b_1
0.00005	0.00001	0.7	0.1	0.00006	0.00002	0.8	0.05	2	8,760	1	1	720

The results are summarized in Figures 10–11. The testing time allocation for the first set of instances is presented in Figure 10(a), and for the second set of instances in Figure 11(a). The second subsystem has larger λ- and β-values, so when both subsystems have only one component, the second subsystem needs more testing time to reach a given reliability and its testing time is greater. By increasing redundancies for subsystem $\ell \in \{1,2\}$, the testing time allocated to subsystem ℓ decreases. The system reliabilities for the first and second set of instances are presented in Figure 10(b) and 11(b). Because the second subsystem has larger λ- and β-values, installing a redundant component in the second subsystem lessens the effect of the less favorable parameter values and thereby improves the system reliability more than it would in the first subsystem; therefore, the reliability in the first set of instances is larger than in the second set of instances.

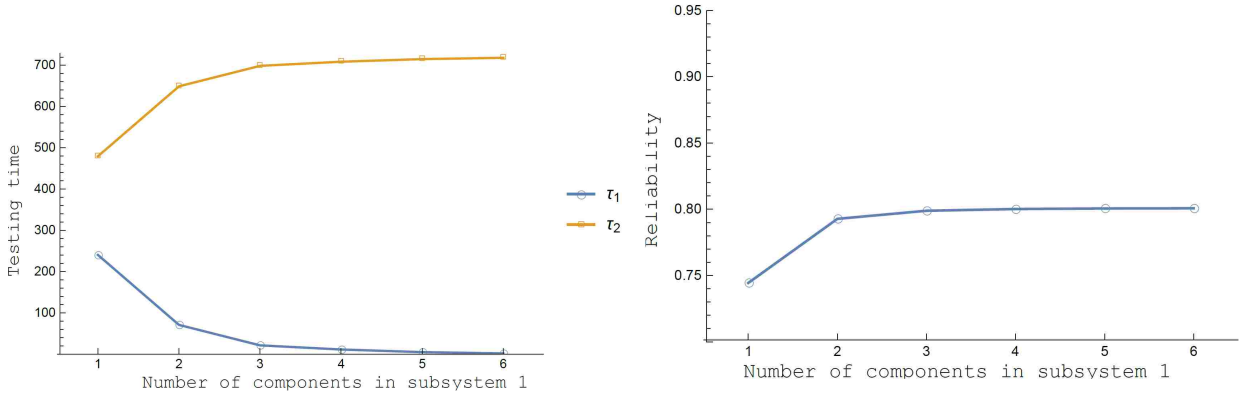
3.5.4 Computational Performance of the Cutting-Plane Algorithm

We now solve five instances to demonstrate the performance of the cutting-plane algorithm for a series system. For each instance, we choose $T = 2190$, $\Phi = 0.5$, $K = 1$, $c_\ell^1 = 1$, $b_1 = 8,760$ and $\tau_\ell^0 = 1$, $\ell \in \{1,2\}$. Moreover, $\bar{\beta}$-values are selected from values 0.6, 0.65, 0.7, 0.75, and 0.8,



(a) **(b)**

Figure 10: (a) Testing time allocation and (b) reliability for the first set of instances



(a) **(b)**

Figure 11: (a) Testing time allocation and (b) reliability for the second set of instances

each of which has an equal probability of being selected, and we select $\bar{\lambda}$-values uniformly from intervals presented in Table 14. We select σ_ℓ^B from the values 0.05 and 0.1, again with an equal chance of selection, and the values of σ_ℓ^L are presented in Table 14. According to Table 14, the cutting-plane algorithm is able to find the optimal solution in no more than a few seconds even for large-size problems.

We also solve five instances to compare the efficiency of the cutting-plane algorithm for series-parallel systems. In the following, we discuss the instance generation procedure for series-parallel instances. Each instance includes 4, 10, 40, 100 or 400 subsystems. We choose $\bar{\beta}$-values from values 0.6, 0.65, 0.7, 0.75, and 0.8, and M_ℓ-values from values 2, 3, and 4, all with equal probability of selection. Each of the N λ-values is selected uniformly from the range described in Table 15.

Table 14: Computational comparison for series instances

Inst.	N	Reliability	Solving time (sec)	$\bar{\lambda}$ range	σ_ℓ^L
1	4	0.958445	2	$[0.00001, 0.00005]$	0.00001
2	10	0.973587	2	$[0.000005, 0.000008]$	0.000002
3	40	0.939991	4	$[0.000001, 0.000004]$	0.000001
4	100	0.967938	8	$[0.0000004, 0.0000007]$	0.0000003
5	400	0.904244	65	$[0.0000001, 0.0000005]$	0.0000001

The value of σ_ℓ^L, $\ell = 1, \ldots, N$, is presented in Table 15, and the value of σ_ℓ^B, $\ell = 1, \ldots, N$, is chosen from values 0.05 and 0.1, each with equal probability of selection. For each instance, we consider initial testing time $\tau_\ell^0 = 1$, $T = 8,760$, $K = 1$, $c_\ell^1 = 1$, $b_1 = 720$, and $\Phi = 0.5$. The time limit is equal to $1,200$ seconds, and we select $\varepsilon = 1 \times 10^{-7}$ and $\varepsilon_{\text{sub}} = 1 \times 10^{-4}$. The value of P_{sub} is equal to $20,000$ for Instances $s \in \{1, \ldots, 3\}$ and $5,000$ for Instances $s \in \{4, 5\}$. As demonstrated in Table 15, the cutting-plane algorithm is consistently able to identify near-optimal solutions for these instances within 20 minutes.

Table 15: Computational comparison for series-parallel instances

Inst.	N	Lower bound	Upper bound	Optimality gap	$\bar{\lambda}$ range	σ_ℓ^L
1	4	0.984892	0.984898	5.43×10^{-6}	$[0.00002, 0.00006]$	0.00001
2	10	0.947247	0.947261	1.43×10^{-5}	$[0.00002, 0.00005]$	0.00001
3	40	0.982389	0.982433	4.42×10^{-5}	$[0.000007, 0.000009]$	0.000003
4	100	0.956155	0.956415	2.61×10^{-4}	$[0.000005, 0.000009]$	0.000002
5	400	0.925610	0.926971	1.36×10^{-3}	$[0.000001, 0.000005]$	0.000001

3.6 Conclusion and Future Research

In this chapter, we consider the robust allocation of testing resources across the subsystems of series and series-parallel systems. We assume that the failures of each component occur according to the AMSAA model with uncertain parameters within a bounded uncertainty set. We propose exact algorithms for both series and series-parallel systems based on a cutting-plane approach, and prove that the algorithm for series systems is convergent. We show how the robust optimization model can be constructed by building confidence intervals from preliminary test data and demonstrate, using a numerical example, that the robust optimization approach is less likely than available deterministic approaches to generate solutions that result in a low reliability. Moreover,

the cutting-plane algorithm for series systems can find the optimal solution in couple of seconds, and the proposed algorithm for series-parallel systems can provides a tight optimality gap for a set of test instances.

Future research may seek to extend our research to consider cold standby redundancy and/or dependent components. The difficulty associated with estimating reliability in a reliability growth environment may also warrant considering resource allocation within a multi-stage reliability growth model, in which the failure information is incorporated at end of each testing stage to have a better estimation of the AMSAA model parameters. Future research may seek to merge the multi-stage concept with the robust resource allocation models utilized herein, yielding a dynamic/sequential test-planning problem in which the budget-restricted uncertainty set becomes narrower during the testing. Furthermore, future research may develop heuristic methods for obtaining high-quality solutions quickly, which would be especially helpful given the computational effort required to solve the series-parallel problem to provable optimality.

4 An Integrated Approach to Redundancy Allocation and Test Planning for Reliability Growth

4.1 Introduction

Two ways to increase a system's reliability are testing its components and installing redundancies. Previous research has considered these two approaches disjointly, but never in combination. In this chapter, we contribute a resource allocation model that merges these two approaches to maximize the reliability of a series-parallel system. For systems in development, it is common to both test components to identify and remove the failure modes of the design, and install redundant components to increase system reliability. This chapter especially has applications for low production volume, high value, and complex systems such as satellites and spacecraft. These types of systems are designed from a mixture of a number of different types of components, some of which are mature in design (i.e., such that no additional testing will be completed) and some of which must be matured by completing testing. For such systems, both development and production costs are significant motivating the use of a resource allocation model to allocate limited resources between these two phases. The model enables weighing the benefits of performing additional testing, which increases development costs but (by increasing component reliability) may allow for reduced redundancy levels (and therefore, reduced production costs).

The redundancy allocation problem (RAP) is the problem of determining component redundancy levels in order to maximize system reliability subject to limited resources. In RAP, there exists a discrete set of component designs, each with a predefined component reliability that is assumed to be known. RAPs have received a significant amount of attention over the years, with specific focus given to series-parallel systems [14, 53, 62, 63], k-out-of-n systems (see, e.g., [48]), or more general network systems [28]. Closely related to RAP is the reliability redundancy allocation problem (RRAP), in which component reliabilities are no longer predefined but can instead modeled as continuous decision variables. RRAPs have also been studied for active standby redundancy (see, e.g., [21, 61]), a combination of cold- and active-standby redundancy strategies [1],

and multi-state systems [55]. Kuo and Prasad [33] and Kuo and Wan [35] provide comprehensive surveys of the redundancy allocation literature.

In this chapter, we consider the problem of maximizing system reliability, subject to limited resources, in a series-parallel system by determining the number and design of components in each subsystem and the amount of testing performed on each component design. We assume that components behave according to the AMSAA reliability growth model with known parameters. All components of the same design in each subsystem are assumed to be identical; hence, testing one design of component equally improves all components of that design. In reliability growth, testing is applied to one or more components of a given design to identify and correct the design's failure modes. Through this process, the reliability is improved for all redundant components of that design. We model this problem as a nonlinear integer program and contribute a decomposition-based exact algorithm to solve the problem.

Our work, which develops exact algorithms, is the first to merge the concepts of redundancy allocation and reliability growth. In comparison to previous research on RRAP, ours is the first to study the relationship between redundancy allocation and reliability growth strategies. In addition, all existing RRAP studies consider only one component design choice in each subsystem; however, we extend RRAP to consider multiple design choices in each subsystem. Moreover, whereas the existing RRAP literature specifies exact algorithms only for the case of convex constraints, we propose an exact algorithm which converges for any set of monotone (and possibly nonconvex) resource constraints.

The remainder of this chapter is organized as follows. Section 4.2 presents the integrated testing/redundancy allocation model. We develop an exact algorithm for the model in Section 4.3. The results are summarized in Section 4.4, and the chapter concludes in Section 4.5.

4.2 Problem Formulation

In this section, we develop a model to solve our problem. We use the reliability function (2) to develop the integrated growth testing and redundancy allocation model. In developing this model,

it will be helpful to define some additional terminology that will be used throughout the remainder of this document. We consider a system containing parallel subsystems connected in a series. We determine the composition of each subsystem. For each subsystem, we have a menu of available component designs, and we determine the number (possibly zero) of components for each design. The following notation will be used in developing the model.

Parameters

E_ℓ	Number of component designs in subsystem $\ell = 1,\ldots,N$
M_ℓ	Number of new designs in subsystem $\ell = 1,\ldots,N$ in which the reliability of the design can be improved through testing
$R_{\ell,i}$	Reliability of mature design $i = M_\ell + 1,\ldots,E_\ell$ components in subsystem $\ell = 1,\ldots,N$
$\lambda_{\ell,i}, \beta_{\ell,i}$	AMSAA model parameters for design $i = 1,\ldots,M_\ell$ components in subsystem $\ell = 1,\ldots,N$
$u_{\ell,i}$	Maximum number of components of design $i = 1,\ldots,E_\ell$ in subsystem $\ell = 1,\ldots,N$
$c^k_{\ell,i}$	Amount of resource $k = 1,\ldots,K$ required per unit time of testing for design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$
$d^k_{\ell,i}$	Amount of resource $k = 1,\ldots,K$ required per unit component of design $i = 1,\ldots,E_\ell$ of subsystem $\ell = 1,\ldots,N$
$\tau^0_{\ell,i}$	Initial testing time of design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$

Decision Variables

$\tau_{\ell,i}$	Testing time for design $i = 1,\ldots,M_\ell$ of subsystem $\ell = 1,\ldots,N$
$n_{\ell,i}$	Number of components of design $i = 1,\ldots,E_\ell$ in subsystem $\ell = 1,\ldots,N$

We now present a formal definition of the problem. To this end, we define N as the number of serially connected subsystems. Each subsystem $\ell = 1,\ldots,N$ includes all components of each design $(\ell,i),\ i = 1,\ldots,E_\ell$ in parallel. We divide the designs into two subsets: new and mature. Designs $i = 1,\ldots,M_\ell$ are considered new, and their reliabilities can be improved through testing. However,

designs $i = M_\ell + 1, \ldots, E_\ell$ are mature, their reliabilities cannot be improved by testing, and the reliabilities are constant. We define variable $n_{\ell,i} \in \{0, \ldots, u_{\ell,i}\}$ as the number of components of design (ℓ, i). Therefore, a system failure takes place if all components in any subsystem fail. In the remainder of this document, we define shorthand $R_{\ell,i}(\tau_{\ell,i}, T) \equiv R(\tau_{\ell,i}; T, \lambda_{\ell,i}, \beta_{\ell,i})$ to refer to component reliability. This problem may be formulated as

$$R^\star = \max \prod_{\ell=1}^{N} \left[1 - \prod_{i=1}^{M_\ell} (1 - R_{\ell,i}(\tau_{\ell,i}, T))^{n_{\ell,i}} \prod_{i=M_\ell+1}^{E_\ell} (1 - R_{\ell,i})^{n_{\ell,i}} \right], \tag{114a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} \sum_{i=1}^{M_\ell} c_{\ell,i}^k \tau_{\ell,i} + \sum_{\ell=1}^{N} \sum_{i=1}^{E_\ell} d_{\ell,i}^k n_{\ell,i} \leq b_k, \qquad \forall k = 1, \ldots, K, \tag{114b}$$

$$\tau_{\ell,i} \geq \tau_{\ell,i}^0, \qquad \forall \ell = 1, \ldots, N,\ i = 1, \ldots, M_\ell, \tag{114c}$$

$$n_{\ell,i} \in \{0, \ldots, u_{\ell,i}\}, \qquad \forall \ell = 1, \ldots, N,\ i = 1, \ldots, E_\ell. \tag{114d}$$

Objective (114a) maximizes the system reliability. Constraint (114b) limits resources (e.g., cost, testing time and weight) expended on testing and redundancy. Constraint (114c) ensures that the testing time for design (ℓ, i) should not be less than initial testing time $\tau_{\ell,i}^0$, and Constraint (114d) establishes that number of components should be a nonnegative integer number less than or equal to $u_{\ell,i}$.

Model (114) generalizes a number of existing models from the literature. Setting $M_\ell = 1,\ \ell = 1, \ldots, N$ and $u_{\ell,i} = 1,\ \ell = 1, \ldots, N,\ i = 1, \ldots, M_\ell$ yields the Coit [15] model. The testing allocation models studied by Dai et al. [18] and Levitin [36] result upon setting $E_\ell = M_\ell$, $u_{\ell,i} = 1$ and $d_{\ell,i}^k = 0,\ \ell = 1, \ldots, N,\ i = 1, ..., M_\ell,\ k = 1, \ldots, K$. The special case corresponding to $M_\ell = 0,\ \ell = 1, \ldots, N$ yields a multi-resource version of the redundancy allocation problem for a series-parallel system. Chern [13] proved that this version of the redundancy allocation problem for a series-parallel system is NP-hard; therefore, our problem which is a generalization of Chern's problem is also NP-hard. We also extend RRAP in several aspects. Previous RRAP models do not formalize the relationship between component reliability and investment in testing using the reliability growth concept. Moreover, only one design choice in each subsystem is considered in the RRAP

problem; however, Model (114) considers multiple design choices for each subsystem. In addition, exact algorithms in the RRAP problem [21, 27, 32, 34, 60] assume that resource consumption is convex in the reliability of each design. Our algorithm, which is discussed in Section 4.3, converges even in the less restrictive case where resource consumption is monotone in the reliability of each design, provided that reliability and resource expenditures are monotone in the amount of testing. In the following section, an optimal algorithm is developed to solve the problem.

In what follows, we use lower-case letters to refer to scalar parameters. Lower-case, bolded variants correspond to vectors (typically in which all of the corresponding scalars from a single subsystem have been aggregated), and upper-case bolded symbols correspond to an aggregation of N subsystem-level vectors.

4.3 Decomposition Algorithm

In this section, we develop an exact algorithm to solve Model (114) within a given optimality gap. Reliability function (2) is nonlinear and nonconcave in general. We address this by adapting a decomposition method used in Chapter 2 that exploits separable subproblems that result upon fixing the resources allocated to each subsystem. Subproblem $\ell = 1,\ldots,N$ entails determining the subsystem ℓ reliability associated with particular available resources. The subproblem for subsystem $\ell = 1,\ldots,N$ and available resources $\mathbf{h}_\ell$ (we use the boldface $\mathbf{h}_\ell$ to represent a K-vector of available resources $h_{\ell,k} \geq h^0_{\ell,k} \equiv \sum_{i=1}^{M_\ell} c^k_{\ell,i}\tau^0_{\ell,i} + \min_{i=1}^{E_\ell}\{d^k_{\ell,i}\}$) is equal to

$$R^{\mathrm{S}}_\ell(\mathbf{h}_\ell, T) = \max\ 1 - \prod_{i=1}^{M_\ell}\left(1 - R_{\ell,i}(\tau_{\ell,i}, T)\right)^{n_{\ell,i}} \prod_{i=M_\ell+1}^{E_\ell}\left(1 - R_{\ell,i}\right)^{n_{\ell,i}}, \tag{115a}$$

$$\text{s.t.} \sum_{i=1}^{M_\ell} c^k_{\ell,i}\tau_{\ell,i} + \sum_{i=1}^{E_\ell} d^k_{\ell,i} n_{\ell,i} \leq h_{\ell,k}, \qquad \forall k = 1,\ldots,K, \tag{115b}$$

$$\tau_{\ell,i} \geq \tau^0_{\ell,i}, \qquad \forall i = 1,\ldots,M_\ell, \tag{115c}$$

$$n_{\ell,i} \in \{0,\ldots,u_{\ell,i}\}, \qquad \forall i = 1,\ldots,E_\ell. \tag{115d}$$

Although Model (115) is simpler in structure than Model (114), it is still nonlinear and nonconcave. Our solution method is based on the idea that solving the discretized version of Model (115) can provide upper and lower bounds on $R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$, the optimal objective value of Model (115). In the discretize-and-refine method, the bounds converge to an arbitrarily small gap through successive refinement of these discrete problems. In the remainder of this section, it will be helpful to define $R_\ell(\tau_\ell, \eta_\ell) \equiv 1 - \prod_{i=1}^{M_\ell}(1 - R_{\ell,i}(\tau_{\ell,i}, T))^{n_{\ell,i}} \prod_{i=M_\ell+1}^{E_\ell}(1 - R_{\ell,i})^{n_{\ell,i}}$ as the reliability of subsystem ℓ given M_ℓ-vectors of test times τ_ℓ and E_ℓ-vectors of number of components η_ℓ. We use the following notation to discretize Model (115).

Parameters

$\overline{u}_{\ell,i}$ Maximum number of components of design (ℓ, i) when the available resources for subproblem are equal to $\mathbf{h}_\ell$ $(i = 1, \ldots, E_{\ell,i})$

P Number of discrete candidate test times for each component (assumed to be equal for all components)

$\tau_{\ell,i}^p$ p-th possible testing time for components of design (ℓ, i) $(p = 1, \ldots, P;\ i = 1, \ldots, M_\ell)$

$r_{\ell,i}^p$ Reliability of candidate test time p in components of design (ℓ, i) $(p = 1, \ldots, P;$ $i = 1, \ldots, M_\ell)$

Decision Variables

$$x_{\ell,i}^p \quad \begin{cases} 1 & \text{if the testing time for design } (\ell, i) \text{ is } \tau_{\ell,i}^p \\ 0 & \text{otherwise} \end{cases}$$

$(p = 1, \ldots, P;\ i = 1, \ldots, M_\ell$

$n_{\ell,i}^p$ Number of components for the candidate test time p of design (ℓ, i) $(p = 1, \ldots, P;$ $i = 1, \ldots, M_\ell)$

For each design $(\ell, i),\ i = 1, \ldots, E_{\ell,i}$, set $\overline{u}_{\ell,i} = \min\{u_{\ell,i}, \min_{k=1}^{K}\{\lfloor {}^{(h_{\ell,k} - \sum_{i=1}^{M_\ell} c_{\ell,i}^k \tau_{\ell,i}^0)}/_{d_{\ell,i}^k} \rfloor\}\}$. To discretize Model (115), consider P testing times $\tau_{\ell,i}^p,\ p = 1, \ldots, P$ for each design $(\ell, i),\ i = 1, \ldots, M_{\ell,i}$

such that $\tau^1_{\ell,i} = \tau^0_{\ell,i}$, and define $r^p_{\ell,i}$ as the reliability associated with candidate test time $p = 1,\ldots,P$. We use Model (116) to calculate the optimal subsystem reliability associated with the discrete candidate test times. We define $\mathbf{r}_\ell$ as a vector of reliabilities $r^p_{\ell,i},\ i = 1,\ldots,M_\ell,\ p = 1,\ldots,P$ and $R_{\ell,i},\ i = M_\ell+1,\ldots,E_\ell$ for subsystem ℓ. Model (116) is given by

$$R^{\mathrm{SD}}_\ell(\mathbf{h}_\ell,\mathbf{r}_\ell) = \max\ 1 - \prod_{i=1}^{M_\ell}\prod_{p=1}^{P}\left(1-r^p_{\ell,i}\right)^{n^p_{\ell,i}} \prod_{i=M_\ell+1}^{E_\ell}\left(1-R_{\ell,i}\right)^{n_{\ell,i}}, \tag{116a}$$

$$\text{s.t.} \sum_{i=1}^{M_\ell}\sum_{p=1}^{P}(c^k_{\ell,i}\tau^p_{\ell,i}x^p_{\ell,i} + d^k_{\ell,i}n^p_{\ell,i}) + \sum_{i=M_\ell+1}^{E_\ell} d^k_{\ell,i}n_{\ell,i} \le h_{\ell,k}, \quad \forall k = 1,\ldots,K, \tag{116b}$$

$$\sum_{p=1}^{P} x^p_{\ell,i} = 1, \qquad \forall i = 1,\ldots,M_\ell, \tag{116c}$$

$$n^p_{\ell,i} \le \overline{u}_{\ell,i}x^p_{\ell,i}, \qquad \forall i = 1,\ldots,M_\ell,\ p = 1,\ldots,P, \tag{116d}$$

$$n^p_{\ell,i} \in \{0,\ldots,\overline{u}_{\ell,i}\}, \qquad \forall i = 1,\ldots,M_\ell,\ p = 1,\ldots,P, \tag{116e}$$

$$n_{\ell,i} \in \{0,\ldots,\overline{u}_{\ell,i}\}, \qquad \forall i = M_\ell+1,\ldots,E_\ell, \tag{116f}$$

$$x^p_{\ell,i} \in \{0,1\}, \qquad \forall i = 1,\ldots,M_\ell,\ p = 1,\ldots,P, \tag{116g}$$

where $R^{\mathrm{SD}}_\ell(\mathbf{h}_\ell,\mathbf{r}_\ell)$ denotes the optimal objective value of Model (116). By applying a logarithm to $1 - R^{\mathrm{SD}}_\ell(\mathbf{h}_\ell,\mathbf{r}_\ell)$, Model (116) is equivalent to a linear model as follows

$$W^{\mathrm{SD}}_\ell(\mathbf{h}_\ell,\mathbf{r}_\ell) = \min \sum_{i=1}^{M_\ell}\sum_{p=1}^{P} n^p_{\ell,i}\ln\left(1-r^p_{\ell,i}\right) + \sum_{i=M_\ell+1}^{E_\ell} n_{\ell,i}\ln\left(1-R_{\ell,i}\right), \tag{117}$$

$$\text{s.t. Constraints (116b)–(116g)},$$

where $1 - e^{W^{\mathrm{SD}}_\ell(\mathbf{h}_\ell,\mathbf{r}_\ell)}$ is equal to $R^{\mathrm{SD}}_\ell(\mathbf{h}_\ell,\mathbf{r}_\ell)$, the optimal objective value of Model (116). In the following remark, we provide an alternative formulation for the discretized version of Model (115).

Remark 6. We also consider an alternative formulation for (116). Define binary variable $y^{p,j}_{\ell,i},\ i = 1,\ldots,M_\ell,\ j = 0,\ldots,\overline{u}_{\ell,i},\ p = 1,\ldots,P$ which is equal to 1 if the design (ℓ,i) has testing time $\tau^p_{\ell,i}$ and redundancy j, and define variable $y^j_{\ell,i},\ i = M_\ell+1,\ldots,E_\ell,\ j = 0,\ldots,\overline{u}_{\ell,i}$ which is equal to 1 if

the design (ℓ, i) has redundancy j. Model (118) is stated as

$$R_\ell^{\mathrm{SD}}(\mathbf{h}_\ell, \mathbf{r}_\ell) = \max\ 1 - \prod_{i=1}^{M_\ell} \prod_{j=0}^{\overline{u}_{\ell,i}} \prod_{p=1}^{P} \left(1 - r_{\ell,i}^p\right)^{j y_{\ell,i}^{p,j}} \prod_{i=M_\ell+1}^{E_\ell} \prod_{j=0}^{\overline{u}_{\ell,i}} \left(1 - R_{\ell,i}\right)^{j y_{\ell,i}^{j}}, \tag{118a}$$

$$\text{s.t.} \sum_{i=1}^{M_\ell} \sum_{j=0}^{\overline{u}_{\ell,i}} \sum_{p=1}^{P} (c_{\ell,i}^k \tau_{\ell,i}^p + d_{\ell,i}^k j) y_{\ell,i}^{p,j} + \sum_{i=M_\ell+1}^{E_\ell} \sum_{j=0}^{\overline{u}_{\ell,i}} d_{\ell,i}^k j y_{\ell,i}^{j} \le h_{\ell,k},$$

$$\forall k = 1, \ldots, K, \tag{118b}$$

$$\sum_{j=0}^{\overline{u}_{\ell,i}} \sum_{p=1}^{P} y_{\ell,i}^{p,j} = 1, \qquad \forall i = 1, \ldots, M_\ell, \tag{118c}$$

$$\sum_{j=0}^{\overline{u}_{\ell,i}} y_{\ell,i}^{j} = 1, \qquad \forall i = M_\ell + 1, \ldots, E_\ell, \tag{118d}$$

$$y_{\ell,i}^{p,j} \in \{0,1\}, \qquad \forall i = 1, \ldots, M_\ell,\ j = 0, \ldots, \overline{u}_{\ell,i},\ p = 1, \ldots, P, \tag{118e}$$

$$y_{\ell,i}^{j} \in \{0,1\}, \qquad \forall i = M_\ell + 1, \ldots, E_\ell,\ j = 0, \ldots, \overline{u}_{\ell,i}, \tag{118f}$$

where $R_\ell^{\mathrm{SD}}(\mathbf{h}_\ell, \mathbf{r}_\ell)$ denotes the optimal objective value of Model (118), which is also equal to the optimal objective value of Model (116). As demonstrated for Model (116), Model (118) can also be linearized by applying a logarithm. Because preliminary experiments suggest that Model (118) almost always solves faster than Model (116), we employ the linearized version of Model (118) when solving subproblems as subroutines of the main algorithm described in this document; however, the notation of Model (116) is somewhat less complicated than that of Model (118), and we hence use it in describing the main algorithm. For instance, instead of referring to a solution of Model (118) in which $y_{\ell,i}^{p,j} = 1,\ i = 1, \ldots, M_\ell,\ p = 1, \ldots, P$, we simply write $x_{\ell,i}^p = 1$ and $n_{\ell,i}^p = j$, and for $y_{\ell,i}^{j} = 1,\ i = M_\ell + 1, \ldots, E_\ell$, we write $n_{\ell,i} = j$.□

In Theorems 20 and 21 respectively, we provide a lower and upper bound for subproblem (115), which extend the results in Chapter 2 for multiple resource constraints and when the number of redundancies is a variable. As we mentioned, solving Model (116) can yield upper and lower bounds on $R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$, the optimal value of Model (115). Considering $r_{\ell,i}^p = R_{\ell,i}(\tau_{\ell,i}^p, T),\ i = 1, \ldots, M_\ell,\ p = 1, \ldots, P$ provides a lower bound, and considering $r_{\ell,i}^p = R_{\ell,i}(\tau_{\ell,i}^{p+1}, T),\ i = 1, \ldots, M_\ell,\ p = 1, \ldots, P$

provides an upper bound. Define $\tau_{\ell,i}^{P+1} \equiv \tau_{\ell,i}^{\max}$, where

$$\tau_{\ell,i}^{\max} \equiv \max\{\tau_{\ell,i} \mid \text{Constraints (114b)–(114d)}\}. \tag{119}$$

Let LB-SD and UB-SD denote the lower and upper bound models, respectively. In the following, we use the notation $\bar{\mathbf{r}}_\ell$ and $\bar{\bar{\mathbf{r}}}_\ell$ in Model (116), respectively, when $\mathbf{r}_\ell$ provides a lower bound and an upper bound in Model (116). For ease of exposition in the algorithm, we use the notation $\mathbf{x}_\ell$ to refer to the PM_ℓ-vectors of testing times and η_ℓ to refer to the PE_ℓ-vectors of the number of components in Model (116).

Theorem 20. Define $\bar{r}_{\ell,i}^p = R_{\ell,i}(\tau_{\ell,i}^p, T),\ i = 1,\ldots,M_\ell,\ p = 1,\ldots,P$ and $\bar{r}_{\ell,i} = R_{\ell,i},\ i = M_\ell + 1,\ldots,E_\ell$, then $R_\ell^{\mathrm{SD}}(\mathbf{h}_\ell, \bar{\mathbf{r}}_\ell) \leq R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$. (That is, LB-SD yields a lower bound for $R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$.)

Proof. For each feasible solution $\hat{\mathbf{x}}_\ell$ and $\hat{\eta}_\ell$ in Model (116) corresponding to $r_{\ell,i}^p = \bar{r}_{\ell,i}^p,\ i = 1,\ldots,M_\ell$, $p = 1,\ldots,P$ and $r_{\ell,i}^p = \bar{r}_{\ell,i},\ i = M_\ell + 1,\ldots,E_\ell$, there exists a solution with $\tau_{\ell,i} = \sum_{p=1}^P \tau_{\ell,i}^p \hat{x}_{\ell,i}^p$ and $n_{\ell,i} = \sum_{p=1}^P \hat{n}_{\ell,i}^p$ for $i = 1,\ldots,M_\ell$ and $n_{\ell,i} = \hat{n}_{\ell,i}$ for $i = M_\ell + 1,\ldots,E_\ell$ in Model (115) with the same objective value. □

Theorem 21. Define $\bar{\bar{r}}_{\ell,i}^p = R_{\ell,i}(\tau_{\ell,i}^{p+1}, T),\ i = 1,\ldots,M_\ell,\ p = 1,\ldots,P$ and $\bar{\bar{r}}_{\ell,i} = R_{\ell,i},\ i = M_\ell + 1,\ldots,E_\ell$, then $R_\ell^{\mathrm{SD}}(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell) \geq R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$. (That is, UB-SD yields an upper bound for $R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$.)

Proof. Consider $\tau_{\ell,i}^\star,\ i = 1,\ldots,M_\ell$ and $n_{\ell,i}^\star,\ i = 1,\ldots,E_\ell$ as an optimal solution for Model (115). For each design $(\ell, i),\ i = 1,\ldots,M_\ell$, find the index $\hat{P}_{\ell,i} \in \{1,\ldots,P\}$ such that $\tau_{\ell,i}^{\hat{P}_{\ell,i}} \leq \tau_{\ell,i}^\star < \tau_{\ell,i}^{\hat{P}_{\ell,i}+1}$, and define $\hat{\tau}_{\ell,i} = \tau_{\ell,i}^{\hat{P}_{\ell,i}+1}$. The solution $x_{\ell,i}^{\hat{P}_{\ell,i}} = 1$ and $n_{\ell,i}^{\hat{P}_{\ell,i}} = n_{\ell,i}^\star,\ i = 1,\ldots,M_\ell$ and $n_{\ell,i} = n_{\ell,i}^\star,\ i = M_\ell + 1,\ldots,E_\ell$ is a feasible solution to UB-SD, with objective value equal to $R_\ell(\hat{\tau}_\ell, \eta_\ell^\star)$. The subsystem reliability function $R_\ell(\tau_\ell, \eta_\ell)$ is nondecreasing over component test times τ_ℓ, so $R_\ell(\hat{\tau}_\ell, \eta_\ell^\star)$ is greater than or equal to $R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T) = R_\ell(\tau_\ell^\star, \eta_\ell^\star)$. As a result, $R_\ell^{\mathrm{SD}}(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell) \geq R_\ell(\hat{\tau}_\ell, \eta_\ell^\star) \geq R_\ell^{\mathrm{S}}(\mathbf{h}_\ell, T)$. □

Algorithm 5 describes the subproblem algorithm, which extends an algorithm, originally inspired by the algorithm in Chapter 2, to the case of the model posed in this chapter (i.e., with

multiple resources and variable redundancy levels). In Chapter 2, the algorithm is given for only a single resource constraint when redundancy levels are fixed.

In Algorithm 5, P_{sub} denotes the initial number of candidate test times and ε_{sub} the optimality gap, and LB and UB denote the lower and upper bound on subsystem reliability. The M_ℓ-vectors of test times $\tau_\ell^\star$ and E_ℓ-vectors of number of components $\eta_\ell^\star$ denote the incumbent solution, which are initially null, and $\hat{\mathbf{x}}_\ell$ and $\hat{\eta}_\ell$ denote the optimal solution of the LB-SD model. The notation $\bar{P}_{\ell,i} \in \{1,\ldots,P\}$ denotes the optimal candidate test time for design (ℓ,i), $i = 1,\ldots,M_\ell$ obtained from the UB-SD model. As we show in Theorem 22, this algorithm computes $R_\ell^{\text{S}}(\mathbf{h}_\ell, T)$ within a given $\varepsilon_{\text{sub}} > 0$ in finite iterations. (Note: The number of discrete candidate test times for each design (ℓ,i) is varied during the development of the subproblem algorithm, but still is denoted as P). Again $\tau_{\ell,i}^{P+1}$ refers to $\tau_{\ell,i}^{\max}$. In Line 5, we discretize the test times. We solve the LB-SD and UB-SD models in Lines 10 and 12, respectively. Note that in Lines 10 and 12, $\bar{\mathbf{r}}_\ell$ and $\bar{\bar{\mathbf{r}}}_\ell$ are updated for a new set of candidate test times to solve LB-SD and UB-SD. We also add a new candidate test time for each design in Line 15.

In the following, the convergence properties of the subproblem algorithm are discussed. By using Lemma 1, Lemma 12 places conditions on the testing times allocated to the components of a system such that the difference of the upper bound system reliability and optimal system reliability is smaller than or equal to ε.

Lemma 12. Define $\bar{x}_\ell$ as a binary solution for UB-SD, and $\bar{P}_{\ell,i} \in \{1,\ldots,P\},\ i = 1,\ldots,M_\ell$ and $\bar{n}_{\ell,i} \in \{0,\ldots,\overline{u}_{\ell,i}\},\ i = 1,\ldots,E_\ell$ as the optimal candidate test time and optimal number of components for design (ℓ,i) such that $\bar{x}_{\ell,i}^{\bar{P}_{\ell,i}} = 1,\ i = 1,\ldots,M_\ell$. Moreover, consider $\bar{\tau}_{\ell,i} = \tau_{\ell,i}^{\bar{P}_{\ell,i}}$, $\bar{\bar{\tau}}_{\ell,i} = \tau_{\ell,i}^{\bar{P}_{\ell,i}+1}$ and $\varepsilon_{\ell,i} = R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}, T)$ for $i = 1,\ldots,M_\ell$. Also, consider $\varepsilon_{\ell,i} = 0$ for $i = M_\ell + 1,\ldots,E_\ell$. Define $M_{\max} = \sum_{i=1}^{E_\ell} \overline{u}_{\ell,i}$ as an upper bound for the number of components in subsystem ℓ, $\varepsilon_{\max} = \max_{i=1}^{M_\ell}\{\varepsilon_{\ell,i}\}$ and $\varepsilon_{\text{sub}} \equiv {M_{\max}\varepsilon_{\max}}/{[1-2M_{\max}\varepsilon_{\max}]}$. If $M_{\max}\varepsilon_{\max} \leq 0.5$, then $R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell) - R_\ell^{\text{S}}(\mathbf{h}_\ell, T) \leq \varepsilon_{\text{sub}}$, where $\bar{\bar{\mathbf{r}}}_\ell$ is defined in Theorem 21.

Proof. We are invoking a special case of Lemma 1. Here there exists one subsystem ℓ; therefore, consider $\bar{N} = 1$ in Lemma 1. Set $\bar{\bar{\rho}}_{\ell,i} = R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}, T)$ and $\bar{\rho}_{\ell,i} = R_{\ell,i}(\bar{\tau}_{\ell,i}, T)$ for $i = 1,\ldots,M_\ell$, and set

Algorithm 5 Discretize and Refine (calculate $R_\ell^S(\mathbf{h}_\ell, T)$ within ε_{sub} tolerance)

1: **function** DISCRETIZEANDREFINE ($\mathbf{h}_\ell$, P_{sub}, ε_{sub})
2: **for** $i = 1, \ldots, M_\ell$ **do**
3: Set $\tau_{\ell,i}^{\min} \leftarrow \tau_{\ell,i}^0$ and $\tau_{\ell,i}^{\max} \leftarrow \min_{k=1}^{K}\{(h_{\ell,k} - \sum_{i'=1}^{\ell} c_{\ell,i'}^k \tau_{\ell,i'}^0 + c_{\ell,i}^k \tau_{\ell,i}^0)/c_{\ell,i}^k\}$
4: **for** $p = 1, \ldots, P_{\text{sub}}$ **do**
5: Set $\tau_{\ell,i}^p \leftarrow \tau_{\ell,i}^{\min} + (p-1)(\tau_{\ell,i}^{\max} - \tau_{\ell,i}^{\min})/P_{\text{sub}}$
6: **end for**
7: **end for**
8: Set $LB \leftarrow 0$, $UB \leftarrow 1$, and $P \leftarrow P_{\text{sub}}$
9: **If** $UB - LB < \varepsilon_{\text{sub}}$ **then return** $(\tau_\ell^\star, \eta_\ell^\star)$ ▷ $(\tau_\ell^\star, \eta_\ell^\star)$ is ε_{sub}-optimal
10: **Solve** the LB-SD model with optimal solution $(\hat{\mathbf{x}}_\ell, \hat{\eta}_\ell)$ and objective value $R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\mathbf{r}}_\ell)$
11: **If** $R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\mathbf{r}}_\ell) > LB$ **then** $LB \leftarrow R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\mathbf{r}}_\ell)$, $\tau_\ell^\star \leftarrow \sum_{p=1}^{P} \tau_{\ell,i}^p \hat{x}_{\ell,i}^p$ and $\eta_\ell^\star \leftarrow \hat{\eta}_\ell$
12: **Solve** the UB-SD model with optimal solution $\bar{P}_{\ell,i} \in \{1, \ldots, P\}$ for each (ℓ, i) and objective value $R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell)$
13: **If** $R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell) < UB$ **then** $UB \leftarrow R_\ell^{\text{SD}}(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell)$
14: **for** $i = 1, \ldots, M_\ell$ **do**
15: **Insert** a new candidate test time $\tau_{\ell,i}^{P+1} \equiv 0.5\left(\tau_{\ell,i}^{\bar{P}_{\ell,i}} + \tau_{\ell,i}^{\bar{P}_{\ell,i}+1}\right)$, $i = 1, \ldots, M_\ell$
16: **Set** $P \leftarrow P + 1$
17: **end for**
18: **for** $i = 1, \ldots, M_\ell$ **do**
19: **Renumber** the candidate test times such that $\tau_{\ell,i}^1 < \tau_{\ell,i}^2 < \ldots < \tau_{\ell,i}^P$
20: **end for**
21: Go to Line 9.
22: **end function**

$\bar{\bar{\rho}}_{\ell,i} = \bar{\rho}_{\ell,i} = R_{\ell,i}$ for $i = M_\ell + 1, \ldots, E_\ell$. By considering $\bar{M}_\ell = \sum_{i=1}^{E_\ell} \bar{u}_{\ell,i}$, $\bar{M}_{\max} = M_{\max}$ and $\varepsilon = \varepsilon_{\text{sub}}$ in Lemma 1, the inequality ${}^{\bar{N}\bar{M}_{\max}\bar{\varepsilon}_{\max}}/_{(1-\bar{M}_{\max}\bar{\varepsilon}_{\max})} \leq 1$ suggests that $M_{\max}\varepsilon_{\max} \leq 0.5$ and equality $\varepsilon = {}^{\bar{N}\bar{M}_{\max}\bar{\varepsilon}_{\max}}/_{[1-(\bar{N}+1)\bar{M}_{\max}\varepsilon_{\max}]}$ results in $\varepsilon_{\text{sub}} = {}^{M_{\max}\varepsilon_{\max}}/_{[1-2M_{\max}\varepsilon_{\max}]}$. For each design (ℓ, i), there exist $\bar{n}_{\ell,i}$ components, so

$$\prod_{\ell=1}^{\bar{N}}\left[1 - \prod_{i=1}^{\bar{M}_\ell}\left(1 - \bar{\bar{\rho}}_{\ell,i}\right)\right] = 1 - \prod_{i=1}^{M_\ell}\left(1 - R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}, T)\right)^{\bar{n}_{\ell,i}} \prod_{i=M_\ell+1}^{E_\ell}\left(1 - R_{\ell,i}\right)^{\bar{n}_{\ell,i}} = R_\ell^{\text{SD}}\left(\mathbf{h}_\ell, \bar{\bar{\mathbf{r}}}_\ell\right). \quad (120)$$

Also, $\bar{\tau}_{\ell,i}$ is a feasible solution for Model (115), hence

$$\prod_{\ell=1}^{\bar{N}}\left[1 - \prod_{i=1}^{\bar{M}_\ell}\left(1 - \bar{\rho}_{\ell,i}\right)\right] = 1 - \prod_{i=1}^{M_\ell}\left(1 - R_{\ell,i}(\bar{\tau}_{\ell,i}, T)\right)^{\bar{n}_{\ell,i}} \prod_{i=M_\ell+1}^{E_\ell}\left(1 - R_{\ell,i}\right)^{\bar{n}_{\ell,i}} \leq R_\ell^{\text{S}}\left(\mathbf{h}_\ell, T\right), \quad (121)$$

which completes the proof. □

In Theorem 22, we use Lemmas 9 and 12 to prove that the subproblem algorithm terminates in finite iterations.

Theorem 22. For any arbitrary $\varepsilon_{\text{sub}} > 0$, the subproblem algorithm terminates in finite steps with ε_{sub}-optimal solution.

Proof. Consider $\varepsilon_{\max} = {\varepsilon_{\text{sub}}}/{[(1+2\varepsilon_{\text{sub}})M_{\max}]}$; therefore,

$$\varepsilon_{\text{sub}} = \frac{M_{\max}\varepsilon_{\max}}{1-2M_{\max}\varepsilon_{\max}}. \tag{122}$$

Because ${\varepsilon_{\text{sub}}}/{(1+2\varepsilon_{\text{sub}})} \leq {\varepsilon_{\text{sub}}}/{2\varepsilon_{\text{sub}}} = 0.5$,

$$\varepsilon_{\max} = \frac{\varepsilon_{\text{sub}}}{(1+2\varepsilon_{\text{sub}})M_{\max}} \leq \frac{0.5}{M_{\max}}, \tag{123}$$

which results in $M_{\max}\varepsilon_{\max} \leq 0.5$. According to Lemma 9 and using $\bar{\tau}_{\ell,i}^{k}$ and $\bar{\bar{\tau}}_{\ell,i}^{k}$ as defined in Lemma 12, the number of iterations k that $R_{\ell,i}(\bar{\bar{\tau}}_{\ell,i}^{k}, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}^{k}, T) > \varepsilon_{\max}$ for some component is bounded. Therefore, by applying Lemma 12 and for $\bar{\bar{\mathbf{r}}}_{\ell}$ defined in Theorem 21, there exists an iteration that $R_{\ell}^{\text{SD}}(\mathbf{h}_{\ell}, \bar{\bar{\mathbf{r}}}_{\ell}) - R_{\ell}^{\text{S}}(\mathbf{h}_{\ell}, T) \leq \varepsilon_{\text{sub}}$. □

In the master problem, we allocate the resources to the subsystems by using a branch-and-bound method. The master problem is defined as follows

$$R^{\star} = \max \prod_{\ell=1}^{N} R_{\ell}^{\text{S}}(\mathbf{h}_{\ell}, T), \tag{124a}$$

$$\text{s.t.} \sum_{\ell=1}^{N} h_{\ell,k} \leq b_k, \qquad \forall k = 1,\ldots,K, \tag{124b}$$

$$h_{\ell,k} \geq h_{\ell,k}^{0}, \qquad \forall \ell = 1,\ldots,N,\ k = 1,\ldots,K, \tag{124c}$$

where $\mathbf{h}_{\ell}$ is a K-vector of available resources $h_{\ell,k}$ allocated to subsystem $\ell = 1,\ldots,N$. To solve the master problem, we develop an approach—based on the notion of branching on the master problem's variables—for exploring the master problem's solution space, while using Algorithm 5 to solve subproblems. Define $\mathbf{H}$ as the $N \times K$ matrix of decision variables, whose elements are

$h_{\ell,k},\ \ell = 1,\ldots,N,\ k = 1,\ldots,K$. Each node of branch and bound tree considers a restricted master problem (i.e., as in [7]) in which $\mathbf{H}$ is restricted to fall in an $N \times V$-dimensional box $[\mathbf{E},\mathbf{F}]$ of form

$$[\mathbf{E},\mathbf{F}] \equiv \left\{\mathbf{H} \mid e_{\ell,k} \leq h_{\ell,k} \leq f_{\ell,k},\ \forall \ell = 1,\ldots,N,\ k = 1,\ldots,K\right\}. \tag{125}$$

Let $L_\ell(\mathbf{E},\mathbf{F})$ and $U_\ell(\mathbf{E},\mathbf{F}),\ \ell = 1,\ldots,N$ define the lower and upper bound on the reliability of subsystem ℓ when Model (124) is restricted to a box $[\mathbf{E},\mathbf{F}]$. The subsystem reliability function is nondecreasing over available resources $h_{\ell,k},\ \ell = 1,\ldots,N,\ k = 1,\ldots,K$; therefore, the bounds could be obtained as $L_\ell(\mathbf{E},\mathbf{F}) = R_\ell^{\mathrm{S}}(\mathbf{e}_\ell,T)$ and $U_\ell(\mathbf{E},\mathbf{F}) = R_\ell^{\mathrm{S}}(\mathbf{f}_\ell,T)$, where $\mathbf{e}_\ell$ and $\mathbf{f}_\ell$ respectively define K-vectors of available resources $e_{\ell,k}$ and $f_{\ell,k},\ k = 1,\ldots,K$. (Unfortunately, $R_\ell^{\mathrm{S}}(.,T)$ can only be calculated within an $\varepsilon_{\mathrm{sub}}$ optimality tolerance; hence, we will settle for values $L_\ell(\mathbf{E},\mathbf{F})$ and $U_\ell(\mathbf{E},\mathbf{F})$ such that $L_\ell(\mathbf{E},\mathbf{F}) \leq R_\ell^{\mathrm{S}}(\mathbf{e}_\ell,T) \leq R_\ell^{\mathrm{S}}(\mathbf{f}_\ell,T) \leq U_\ell(\mathbf{E},\mathbf{F})$, $R_\ell^{\mathrm{S}}(\mathbf{e}_\ell,T) - L_\ell(\mathbf{E},\mathbf{F}) \leq \varepsilon_{\mathrm{sub}}$, and $U_\ell(\mathbf{E},\mathbf{F}) - R_\ell^{\mathrm{S}}(\mathbf{f}_\ell,T) \leq \varepsilon_{\mathrm{sub}}$). Because the system reliability is nondecreasing over subsystem reliabilities, $L(\mathbf{E},\mathbf{F}) = \prod_{\ell=1}^{N} L_\ell(\mathbf{E},\mathbf{F})$ and $U(\mathbf{E},\mathbf{F}) = \prod_{\ell=1}^{N} U_\ell(\mathbf{E},\mathbf{F})$ provide lower and upper bounds on the master problem's reliability, when restricted to box $[\mathbf{E},\mathbf{F}]$. In Algorithm 6, we use DISCRETIZEANDREFINE (i.e., Algorithm 5) to get lower and upper bounds on reliability for the restricted box $[\mathbf{E},\mathbf{F}]$. When the feasible region's box is subdivided into smaller sub-boxes, the bounds on system reliability become tighter. We exploit this property by branching with the hope of using these bounds to prune sub-boxes to keep the effort reasonable. To develop the algorithm, define the m-th $N \times K$-dimensional box as follows

$$[\mathbf{E}^m,\mathbf{F}^m] \equiv \left\{\mathbf{H}^m \mid e_{\ell,k}^m \leq h_{\ell,k}^m \leq f_{\ell,k}^m,\ \forall \ell = 1,\ldots,N,\ k = 1,\ldots,K\right\}. \tag{126}$$

Over the course of the algorithm, we keep track of global lower and upper bounds on the optimal objective function value, so naturally we can prune a box if its upper bound is no more than the global lower bound for the optimal objective function. Define LB and UB as the global lower and upper bounds on the optimal system reliability obtained through the algorithm. The conditions under which we can prune a box are outlined below.

Condition 1. If the box upper bound is no more than the global lower bound LB.

To improve computational efficiency, we may also prune certain boxes without solving subproblems.

Condition 2. Constraint (124b) is violated with respect to $\mathbf{E}^m$ at least for one resource $k = 1, \ldots, K$. In this case, $[\mathbf{E}^m, \mathbf{F}^m]$ contains no feasible solutions and hence cannot contain an optimal solution.

Condition 3. Constraint (124b) is satisfied strictly (i.e., not at equality) for all resources $k = 1, \ldots, K$ with respect to the solution $\mathbf{F}^m$. Because the monotone objective ensures there is an optimal solution in which one of the resource constraints is tight, we can guarantee that pruning $[\mathbf{E}^m, \mathbf{F}^m]$ will not remove all optimal solutions (and is therefore allowable).

Algorithm 6 calculates the lower and upper bound on system reliabilities of boxes $\mathbf{E}$ and $\mathbf{F}$. In Algorithm 6, P_{sub} and ε_{sub} denote the initial number of candidate test times for each component design and subsystem optimality tolerance. Moreover, Algorithm 7 prunes a box based on Conditions 2–3. Set Z (A, respectively) as the set of active $N \times K$-dimensional boxes for which bounds have not (have, respectively) been computed. Let L^m and U^m denote the lower and upper bound reliability for box $[\mathbf{E}^m, \mathbf{F}^m]$, and define $\mathbf{H}^\star$ as the incumbent solution. The branch and bound algorithm, which calls the subroutines in Algorithms 6–7, is represented in Algorithm 8. We define the initial $N \times K$-dimensional box in Lines 3–8. Note that in Lines 24–25 of Algorithm 8, we split the box $[\mathbf{E}^{m^\star}, \mathbf{F}^{m^\star}]$ to two smaller boxes, and in Line 20, a binary heap algorithm is used to find box $\left[\mathbf{E}^{m^\star}, \mathbf{F}^{m^\star}\right]$ having maximum value of U^m.

Algorithm 6 Returns (L, U), ε_{sub}-tight lower and upper bounds, respectively, on $R_\ell^S(\mathbf{e}_\ell, T)$ and $R_\ell^S(\mathbf{f}_\ell, T)$

```
1: function COMPUTEBOUNDS (E, F, P_sub, ε_sub)
2:     for ℓ = 1, ..., N do
3:         (L_ℓ, Ū_ℓ) ← DISCRETIZEANDREFINE(e_ℓ, P_sub, ε_sub)
4:         (L̄_ℓ, U_ℓ) ← DISCRETIZEANDREFINE(f_ℓ, P_sub, ε_sub)
5:     end for
6:     return (∏_{ℓ=1}^N L_ℓ, ∏_{ℓ=1}^N U_ℓ)
7: end function
```

Algorithm 7 Returns true if $(\mathbf{E}, \mathbf{F})$ can be pruned without knowledge of the objective

function PRUNECHECK $(\mathbf{E}, \mathbf{F})$
  for $k = 1, \ldots, K$ **do**
    If $\sum_{\ell=1}^{N} e_{\ell,k}^{m} > b_k$ **then return** true ▷ Prune based on condition 2
  end for
  for $k = 1, \ldots, K$ **do**
    If $\sum_{\ell=1}^{N} f_{\ell,k}^{m} \geq b_k$ **then return** false
  end for
  return true ▷ Prune based on condition 3
end function

Algorithm 8 Branch and Bound (calculate $R^\star$ within ε tolerance)

function BRANCHANDBOUND (P_{sub}, ε_{sub}, ε)
  Set $Z \leftarrow \{0\}$, $A \leftarrow \emptyset$, $LB \leftarrow 0$, $UB \leftarrow 1$, and $j \leftarrow 0$
  for $\ell = 1, \ldots, N$ **do**
    for $k = 1, \ldots, K$ **do**
      Set $e_{\ell,k}^{0} \leftarrow h_{\ell,k}^{0}$ ▷ Initialize $\mathbf{E}^0$
      Set $f_{\ell,k}^{0} \leftarrow b_k - \sum_{\ell' \neq \ell} h_{\ell',k}^{0}$ ▷ Initialize $\mathbf{F}^0$
    end for
  end for
  while $Z \neq \emptyset$ **do**
    Select $m \in Z$
    if PRUNECHECK($\mathbf{E}^m$,$\mathbf{F}^m$) **then**
      Set $Z \leftarrow Z \setminus \{m\}$ ▷ prune based on conditions 2–3
    else
      $(L^m, U^m) \leftarrow$ COMPUTEBOUNDS($\mathbf{E}$, $\mathbf{F}$, P_{sub}, ε_{sub})
      If $L^m > LB$ **then** set $LB \leftarrow L^m$ and $\mathbf{H}^\star \leftarrow \mathbf{E}^m$ ▷ new incumbent
      If $U^m > LB$ **then** set $A \leftarrow A \cup \{m\}$ ▷ If $U^m \leq LB$ prune based on condition 1
      Set $Z \leftarrow Z \setminus \{m\}$
    end if
  end while
  Select $m^\star \in \arg\max_{m \in A} U^m$
  Set $UB \leftarrow U^{m^\star}$ ▷ update the upper bound UB
  If $UB - LB < \varepsilon$ **then return** $\mathbf{H}^\star$ ▷ $\mathbf{H}^\star$ is ε-optimal
  Select $(\ell^\star, k^\star) \in \arg\max_{\ell=1,\ldots,N,\ k=1,\ldots,K} \{f_{\ell,k}^{m^\star} - e_{\ell,k}^{m^\star}\}$
  Set $\mathbf{E}^{j+1} \leftarrow \mathbf{E}^{m^\star}$, $\mathbf{F}^{j+1} \leftarrow \mathbf{F}^{m^\star}$, $\mathbf{E}^{j+2} \leftarrow \mathbf{E}^{m^\star}$, and $\mathbf{F}^{j+2} \leftarrow \mathbf{F}^{m^\star}$
  Set $f_{\ell^\star,k^\star}^{k+1} \leftarrow 0.5(e_{\ell^\star,k^\star}^{m^\star} + f_{\ell^\star,k^\star}^{m^\star})$ and $e_{\ell^\star,k^\star}^{j+2} \leftarrow 0.5(e_{\ell^\star,k^\star}^{m^\star} + f_{\ell^\star,k^\star}^{m^\star})$
  Set $A \leftarrow A \setminus \{m^\star\}$, $Z \leftarrow Z \cup \{j+1, j+2\}$, $j \leftarrow j+2$
  Go to line 9
end function

As we demonstrate in Remark 7, we can use the solved subproblems in the previous iterations

to find the solution for the current box, without unnecessarily solving the subproblem's algorithm.

Remark 7. In Line 3 of Algorithm 6, let $\hat{\mathbf{x}}_\ell$ and $\hat{\boldsymbol{\eta}}_\ell$ define the lower bound solution obtained from solving DISCRETIZEANDREFINE$(\mathbf{e}_\ell, P_{\text{sub}}, \varepsilon_{\text{sub}})$. Let $\bar{e}_{\ell,k} = \sum_{i=1}^{M_\ell} c_{\ell,i}^k \hat{\tau}_{\ell,i} + d_{\ell,i}^k \hat{n}_{\ell,i}$ and observe that $\bar{\mathbf{e}}_\ell \leq \mathbf{e}_\ell$, and it is possible that $\bar{e}_{\ell,k} < e_{\ell,k}$ for some resource $k = 1,\ldots,K$. Hence, there are distinct vectors $\bar{\bar{\mathbf{e}}}_\ell \neq \{\mathbf{e}_\ell, \bar{\mathbf{e}}_\ell\}$ such that $\bar{\mathbf{e}}_\ell \leq \bar{\bar{\mathbf{e}}}_\ell \leq \mathbf{e}_\ell$ for which we may wish to solve DISCRETIZEANDREFINE$(\bar{\bar{\mathbf{e}}}_\ell, P_{\text{sub}}, \varepsilon_{\text{sub}})$ in future iterations of Algorithm 8. Because $\bar{\bar{\mathbf{e}}}_\ell$ remains feasible to this (restricted) problem, it must be ε_{sub}-optimal for this problem. Hence, by saving a list of historical solutions in memory, it is possible to avoid unnecessary calls to the DISCRETIZEANDREFINE subroutine. (Similarly, upper bound solutions can saved with respect to DISCRETIZEANDREFINE$(\mathbf{f}_\ell, P_{\text{sub}}, \varepsilon_{\text{sub}})$ to avoid unnecessary calls to DISCRETIZEANDREFINE.)□

In Remark 8, we demonstrate that the global lower bound can be improved even after the decomposition algorithm is terminated.

Remark 8. Define $P'_{\text{sub}} > P_{\text{sub}}$ and $\varepsilon'_{\text{sub}} < \varepsilon_{\text{sub}}$ as the initial number of candidate test times and the subsystem gap for each component design, respectively. We can improve the global lower bound *LB* even after Algorithm 8 is terminated. The algorithm terminates with an allocation of resources to the subsystems, which may not be tight in any of the resource constraints. Therefore, we can first scale up resources to have at least one tight resource constraint, and then run DISCRETIZEANDREFINE with a tighter subsystem gap $\varepsilon'_{\text{sub}}$ and a higher initial number of candidate test times P'_{sub} to get a better global lower bound solution.□

We now prove the convergence of Algorithm 8. Lemma 13 defines sufficient conditions for the box $[\mathbf{E}, \mathbf{F}]$ such that the differences between system reliabilities inside the box is less than or equal to an arbitrary gap ε.

Lemma 13. For each $\varepsilon > 0$, there exists $\delta_\ell > 0$ such that for any box $[\mathbf{E}^m, \mathbf{F}^m]$ if $f_{\ell,k}^m - e_{\ell,k}^m \leq \delta_\ell$ for all resources $k = 1,\ldots,K$, then $R_\ell^{\text{S}}(\mathbf{f}_\ell^m, T) - R_\ell^{\text{S}}(\mathbf{e}_\ell^m, T) \leq \varepsilon$, in which $\mathbf{e}_\ell^m$ and $\mathbf{f}_\ell^m$ are K-vectors of available resources $e_{\ell,k}^m$ and $f_{\ell,k}^m$, $k = 1,\ldots,K$, respectively.

Proof. In Lemma 4, it is proven that for $\bar{\varepsilon}_{\max} > 0$, there exists $\bar{\delta}_{\ell,i} > 0$ such that $R_{\ell,i}(\tau_{\ell,i} + \bar{\delta}_{\ell,i}, T) - R_{\ell,i}(\tau_{\ell,i}, T) \leq \bar{\varepsilon}_{\max}$ for $\tau_{\ell,i}^{\min} \leq \tau_{\ell,i} \leq \tau_{\ell,i}^{\max}$. Define $\bar{\delta}_\ell \equiv \min_{i=1}^{M_\ell}\{\bar{\delta}_{\ell,i}\}$ and $\delta_\ell \equiv \bar{\delta}_\ell \min_{k=1}^{K}\{\sum_{i=1}^{M_\ell} c_{\ell,i}^k\}$. Moreover, define $\tau_{\ell,i}^\star$ and $n_{\ell,i}^\star$ as the solution of Model (115) with available resources $\mathbf{f}_\ell^m$. Define the solution with testing times $\bar{\tau}_{\ell,i} \equiv \tau_{\ell,i}^\star - \bar{\delta}_{\ell,i}$ and number of redundancies $n_{\ell,i}^\star$, so $R_{\ell,i}(\tau_{\ell,i}^\star, T) - R_{\ell,i}(\bar{\tau}_{\ell,i}, T) \leq \bar{\varepsilon}_{\max}$. In Lemma 1, define $\bar{\bar{\rho}}_{\ell,i} = R_{\ell,i}(\tau_{\ell,i}^\star, T)$ and $\bar{\rho}_{\ell,i} = R_{\ell,i}(\bar{\tau}_{\ell,i}, T)$ for $i = 1, \ldots, M_\ell$, and set $\bar{\bar{\rho}}_{\ell,i} = \bar{\rho}_{\ell,i} = R_{\ell,i}$ for $i = M_\ell + 1, \ldots, E_\ell$. Also, consider $\bar{N} = 1$, $\bar{M}_{\max} = \sum_{\ell=1}^{E_\ell} n_{\ell,i}^\star$ and $\bar{\varepsilon}_{\max} = {\varepsilon}/{(\bar{N}\bar{M}_{\max} + \varepsilon(\bar{N}+1)\bar{M}_{\max})}$; therefore, $\varepsilon = {\bar{N}\bar{M}_{\max}\bar{\varepsilon}_{\max}}/{(1-(\bar{N}+1)\bar{M}_{\max}\bar{\varepsilon}_{\max})}$. By considering $\bar{N} = 1$, ${\bar{N}\bar{M}_{\max}\bar{\varepsilon}_{\max}}/{(1-\bar{M}_{\max}\bar{\varepsilon}_{\max})} \leq 1$ results in $\bar{M}_{\max}\bar{\varepsilon}_{\max} \leq 0.5$ and $\bar{\varepsilon}_{\max} = {\varepsilon}/{(\bar{N}\bar{M}_{\max} + \varepsilon(\bar{N}+1)\bar{M}_{\max})}$ implies $\bar{\varepsilon}_{\max} = {\varepsilon}/{(\bar{M}_{\max} + 2\varepsilon\bar{M}_{\max})}$. In Theorem 22, it is shown $\bar{\varepsilon}_{\max} = {\varepsilon}/{(\bar{M}_{\max} + 2\varepsilon\bar{M}_{\max})} \leq {0.5}/{\bar{M}_{\max}}$ or $\bar{\varepsilon}_{\max}\bar{M}_{\max} \leq 0.5$. As a result, by applying Lemma 1,

$$\left[1 - \prod_{i=1}^{M_\ell}(1 - R_{\ell,i}(\tau_{\ell,i}^\star, T))^{n_{\ell,i}^\star} \prod_{i=M_\ell+1}^{E_\ell}(1 - R_{\ell,i})^{n_{\ell,i}^\star}\right] - \left[1 - \prod_{i=1}^{M_\ell}(1 - R_{\ell,i}(\bar{\tau}_{\ell,i}, T))^{n_{\ell,i}^\star} \prod_{i=M_\ell+1}^{E_\ell}(1 - R_{\ell,i})^{n_{\ell,i}^\star}\right] \leq \varepsilon. \quad (127)$$

The solution $\tau_{\ell,i}^\star,\ i = 1, \ldots, M_\ell$ and $n_{\ell,i}^\star,\ i = 1, \ldots, E_\ell$ is feasible for Model (115) with available resources $\mathbf{f}_\ell^m$, so $\sum_{i=1}^{M_\ell} c_{\ell,i}^k \tau_{\ell,i}^\star + \sum_{i=1}^{E_\ell} d_{\ell,i}^k n_{\ell,i}^\star \leq f_{\ell,k}^m,\ k = 1, \ldots, K$. For solution $\bar{\tau}_{\ell,i}$ and $n_{\ell,i}^\star$, we have

$$\sum_{i=1}^{M_\ell} c_{\ell,i}^k \bar{\tau}_{\ell,i} = \sum_{i=1}^{M_\ell} c_{\ell,i}^k \tau_{\ell,i}^\star - \sum_{i=1}^{M_\ell} c_{\ell,i}^k \bar{\delta}_{\ell,i} \leq f_{\ell,k}^m - \sum_{i=1}^{E_\ell} d_{\ell,i}^k n_{\ell,i}^\star - \bar{\delta}_\ell \sum_{i=1}^{M_\ell} c_{\ell,i}^k, \quad (128a)$$

$$\leq f_{\ell,k}^m - \sum_{i=1}^{E_\ell} d_{\ell,i}^k n_{\ell,i}^\star - \delta_\ell \leq e_{\ell,k}^m - \sum_{i=1}^{E_\ell} d_{\ell,i}^k n_{\ell,i}^\star, \quad (128b)$$

where the inequality in (128a) is valid due to replacing $\bar{\delta}_{\ell,i}$ with lower bound $\bar{\delta}_\ell$. Also, because $\delta_\ell \leq \bar{\delta}_\ell \sum_{i=1}^{M_\ell} c_{\ell,i}^k$, δ_ℓ is used instead of $\bar{\delta}_\ell \sum_{i=1}^{M_\ell} c_{\ell,i}^k$ in the first inequality of (128b). Therefore, the solution $\bar{\tau}_{\ell,i},\ i = 1, \ldots, M_\ell$ and $n_{\ell,i}^\star,\ i = 1, \ldots, E_\ell$ is feasible for Model (115) with available resources $\mathbf{e}_\ell^m$, which results in $R_\ell^{\mathrm{S}}(\mathbf{e}_\ell^m, T) \geq \left[1 - \prod_{i=1}^{M_\ell}(1 - R(\bar{\tau}_{\ell,i}, T))^{n_{\ell,i}^\star} \prod_{i=M_\ell+1}^{E_\ell}(1 - R_{\ell,i})^{n_{\ell,i}^\star}\right]$. By definition, $R_\ell^{\mathrm{S}}(\mathbf{f}_\ell^m, T) = \left[1 - \prod_{i=1}^{M_\ell}(1 - R(\tau_{\ell,i}^\star, T))^{n_{\ell,i}^\star} \prod_{i=M_\ell+1}^{E_\ell}(1 - R_{\ell,i})^{n_{\ell,i}^\star}\right]$ and Equation (127) completes the proof. □

Lemma 14 proves that for each $\delta > 0$, there is one iteration of the master problem algorithm such that, for the box selected in that iteration, the difference between the lower and upper bound of available resources in the box is smaller than or equal to δ for all $\ell = 1,\ldots,N,\ k = 1,\ldots,K$.

Lemma 14. For each $\delta > 0$, there exists one iteration of the algorithm in which for box $m^\star \in \arg\max_{m\in A} U^m$ that is selected in Line 20 of Algorithm 8, $f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k} \le \delta$ for all $\ell = 1,\ldots,N,\ k = 1,\ldots,K$.

Proof. Let $Q \equiv \left\{\{\ell,k\} \mid f^0_{\ell,k} - e^0_{\ell,k} \ge {}^{\delta}/_{2}\right\}$ define the set of pairs (ℓ,k) such that initially $f^0_{\ell,k} - e^0_{\ell,k} \ge {}^{\delta}/_{2}$. Also, define $J \equiv \left\lceil {}^{\left(2\max_{\ell=1,\ldots,N,\ k=1,\ldots,K}\left\{f^0_{\ell,v} - e^0_{\ell,v}\right\}\right)}/_{\delta} \right\rceil$. In each iteration, for box $m^\star$, pair $(\ell^\star,k^\star) \in \arg\max_{\ell=1,\ldots,N,\ k=1,\ldots,K}\{f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k}\}$ having maximum value of $f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k}$ is selected; therefore, until $f^m_{\ell,k} - e^m_{\ell,k} \le {}^{\delta}/_{2}$ for at least one pair $(\ell,k) \in Q$ in a box $[\mathbf{E}^m,\mathbf{F}^m]$, pairs $(\ell,k) \notin Q$ is not selected. As a result, after $J^{NK}+1$ iterations, there is at least one box $[\mathbf{E}^m,\mathbf{F}^m],\ m \in A$ such that $f^m_{\ell,k} - e^m_{\ell,k} \le {}^{\delta}/_{2}$ for a pair $(\ell,k) \in Q$. Consider box $[\mathbf{E}^{\bar{m}},\mathbf{F}^{\bar{m}}]$ as the first box that has a pair $\left(\bar{\ell},\bar{k}\right) \in Q$ such that $f^{\bar{m}}_{\bar{\ell},\bar{k}} - e^{\bar{m}}_{\bar{\ell},\bar{k}} \le {}^{\delta}/_{2}$. Assume box $[\mathbf{E}^{\bar{m}},\mathbf{F}^{\bar{m}}]$ is created from box $[\mathbf{E}^{m^\star},\mathbf{F}^{m^\star}]$, so box $[\mathbf{E}^{m^\star},\mathbf{F}^{m^\star}]$ was, in the previous iteration, selected in Line 20 of Algorithm 8. In box $[\mathbf{E}^{\bar{m}},\mathbf{F}^{\bar{m}}]$, for the first time $f^{\bar{m}}_{\bar{\ell},\bar{k}} - e^{\bar{m}}_{\bar{\ell},\bar{k}} \le {}^{\delta}/_{2}$, so the pair $\left(\bar{\ell},\bar{k}\right) \in \arg\max_{\ell=1,\ldots,N,\ k=1,\ldots,K}\{f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k}\}$ has the maximum value of $f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k}$ for box $[\mathbf{E}^{m^\star},\mathbf{F}^{m^\star}]$, which suggests $f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k} \le \delta$ for all pairs (ℓ,k) of box $[\mathbf{E}^{m^\star},\mathbf{F}^{m^\star}]$. □

We now use Lemmas 13 and 14 to prove that the decomposition algorithm terminates in finite iterations.

Theorem 23. For each $\varepsilon > 0$, the decomposition algorithm terminates in finite iterations with ε-optimal solution.

Proof. Define $\varepsilon' = \min\{{}^{\bar{N}}/_{\bar{N}+1}, \varepsilon\}$ and consider $\varepsilon'' = {}^{\varepsilon'}/_{(\bar{N}\bar{M}_{\max}+(\bar{N}+1)\bar{M}_{\max}\varepsilon')}$, in which $\bar{M}_{\max} = 1$ and $\bar{N} = N$. Moreover, consider $\varepsilon_{\max} = \varepsilon_{\text{sub}} = {}^{\varepsilon''}/_{3}$. According to Lemma 13, for every $\varepsilon_{\max} > 0$, there exists $\delta_\ell > 0$ such that for any box $[\mathbf{E}^m,\mathbf{F}^m]$ if $f^m_{\ell,k} - e^m_{\ell,k} \le \delta_\ell$ for all resources $k = 1,\ldots,K$, then $R^{\text{S}}_\ell(\mathbf{f}^m_\ell,T) - R^{\text{S}}_\ell(\mathbf{e}^m_\ell,T) \le \varepsilon_{\max}$. Define $\delta = \min^N_{\ell=1}\{\delta_\ell\}$. By Lemma 14, there exists one iteration of the algorithm in which for box $m^\star \in \arg\max_{m\in A} U^m$ selected in Line 20 of Algorithm 8,

$f^{m^\star}_{\ell,k} - e^{m^\star}_{\ell,k} \leq \delta$ for all $\ell = 1,\ldots,N,\ k = 1,\ldots,K$, which implies for box $[\mathbf{E}^{m^\star}, \mathbf{F}^{m^\star}]$, $R^{\mathrm{S}}_\ell(\mathbf{f}^{m^\star}_\ell, T) - R^{\mathrm{S}}_\ell(\mathbf{e}^{m^\star}_\ell, T) \leq \varepsilon_{\max}$. Therefore, for box $[\mathbf{E}^{m^\star}, \mathbf{F}^{m^\star}]$,

$$U^{m^\star}_\ell - L^{m^\star}_\ell = \left[U^{m^\star}_\ell - R^{\mathrm{S}}_\ell(\mathbf{f}^{m^\star}_\ell, T)\right] + \left[R^{\mathrm{S}}_\ell(\mathbf{f}^{m^\star}_\ell, T) - R^{\mathrm{S}}_\ell(\mathbf{e}^{m^\star}_\ell, T)\right] + \left[R^{\mathrm{S}}_\ell(\mathbf{e}^{m^\star}_\ell, T) - L^{m^\star}_\ell\right], \quad (129a)$$

$$\leq \varepsilon_{\mathrm{sub}} + \varepsilon_{\max} + \varepsilon_{\mathrm{sub}} = \varepsilon''. \quad (129b)$$

Moreover, we have

$$\varepsilon'' = \frac{\varepsilon'}{\bar{N}\bar{M}_{\max} + (\bar{N}+1)\bar{M}_{\max}\varepsilon'} \leq \frac{\varepsilon'}{\bar{N}\bar{M}_{\max}} \leq \frac{1}{(\bar{N}+1)\bar{M}_{\max}}, \quad (130)$$

where the second inequality results because $\varepsilon' \leq {\bar{N}}/{\bar{N}+1}$. By using Equation (130), $\bar{M}_{\max}\varepsilon'' \leq 1$ and ${\bar{N}\bar{M}_{\max}\varepsilon''}/{1-\bar{M}_{\max}\varepsilon''} \leq 1$. In Lemma 1, consider $\bar{N} = N$, $\bar{M}_\ell = 1$, $\bar{\bar{\rho}}_\ell = U^{m^\star}_\ell$ and $\bar{\rho}_\ell = L^{m^\star}_\ell$. Because $\bar{M}_{\max}\varepsilon'' \leq 1$ and ${\bar{N}\bar{M}_{\max}\varepsilon''}/{1-\bar{M}_{\max}\varepsilon''} \leq 1$, by applying Lemma 1,

$$U^{m^\star} - L^{m^\star} \leq \varepsilon' \leq \varepsilon. \quad (131)$$

However, in that iteration $UB = U^{m^\star}$ and $LB \geq L^{m^\star}$, which prove the theorem. □

In the following section, we discuss experiments based on our implementation of the algorithm described in this section.

4.4 Numerical Results

The algorithms from the previous section were coded in C++ and tested on a server with an Intel core i12 with 3.1 GHz and 12 GB RAM, using CPLEX 12.4 to solve all of the integer programming models. Initially, we consider an example adapted from one provided by Sung and Cho [52], and later studied by Ramirez-Marquez et al. [44]. The system includes three subsystems in a series, each of which has two or three designs. We consider $M_\ell = 1$, $E_1 = E_2 = 3$, $E_3 = 2$ and $u_{\ell,i} = 10$ for all $\ell = 1,\ldots,3,\ i = 1,\ldots,E_\ell$. There are two resource constraints, in which $c^1_{\ell,1} = 0,\ \ell = 1,\ldots,3$, for the first constraint. For the second resource constraint and all subsystems $\ell = 1,\ldots,3$,

we set $d^2_{\ell,i} = 0,\ i = 1,\ldots,E_\ell$, and $c^2_{\ell,1} = 1$. The first resource constraint is a cost constraint and limits the redundancy, and the second constraint is a testing constraint which limits the testing times. We choose $\beta_{\ell,i}$-values randomly selected from values 0.65, 0.7, 0.75, 0.8 and 0.85. We select $\lambda_{\ell,1},\ \ell = 1,\ldots,3$, such that the reliability of that design with testing time equal to ${}^{b_2}/_N$ is equal to the reliability given in [52], with the exception that λ-values are then rounded with three significant figures. Thus, the model we consider admits a feasible solution with $\tau_{1,1} = \tau_{2,1} = \tau_{3,1} = {}^{b_2}/_N = {}^{720}/_3 = 240$. Extending this assignment to τ with an optimal solution to the resulting RAP (obtained from [52]) results in a reliability of 0.975934. All other parameter values are specified in [52]. These parameters are presented in Table 16.

Table 16: Parameters for the example with 3 subsystems and 2 or 3 designs in each subsystem

	Design 1			Design 2		Design 3	
ℓ	$\lambda_{\ell,1}$	$\beta_{\ell,1}$	$d^1_{\ell,1}$	$R_{\ell,2}$	$d^1_{\ell,2}$	$R_{\ell,3}$	$d^1_{\ell,3}$
1	1.20×10^{-5}	0.65	4	0.95	13	0.92	7
2	6.17×10^{-6}	0.85	8	0.8	3	0.9	3
3	1.71×10^{-5}	0.7	11	0.92	5		
	$b_1 = 30$		$b_2 = 720$		$T = 8,760$		

For this example, we choose $\varepsilon = 1\times10^{-8}$, $\varepsilon'_{\text{sub}} = 1\times10^{-2}$, $\varepsilon_{\text{sub}} = 5\times10^{-2}$, $P'_{\text{sub}} = 40,000$ and $P_{\text{sub}} = 20,000$ ($\varepsilon'_{\text{sub}}$ and P'_{sub} are defined in Remark 8) and consider the time limit equal to $3,600$ seconds. In Table 17, we summarize the results of solving the example. The obtained optimal testing times $\tau_{\ell,1},\ \ell = 1,\ldots,3$, and the number of redundancies $n_{\ell,i},\ \ell = 1,\ldots,3,\ i = 1,\ldots,E_\ell$, are also presented in Table 17, along with the cumulative solving times at each iteration when using either Model (116) or Model (118). The results suggest that the solving time when Model (118) is used for subproblems is less than the solving times of Model (116), and we have therefore used Model (118) in all additional results.

In the first subsystem, the reliability value of the first design with $\tau_{1,1} = 717.99$ is equal to 0.993, so it has greater reliability and smaller cost than the second and third designs, so its number of redundancies is greater. In the second subsystem, the third design has greater reliability value than the second subsystem and their cost values are similar, so the number of redundancies for the third design is greater. Moreover, the reliability of the first design of first subsystem, which

is equal to 0.993 under $\tau_{1,1} = 717.99$, is much greater than the reliability of the third design of second subsystem, so it needs smaller number of redundancies than the second subsystem. The optimal solution in [52] is $n_{1,1} = 2,\ n_{2,1} = n_{2,2} = n_{3,1} = 1,\ n_{1,2} = n_{1,3} = n_{2,3} = n_{3,2} = 0$ with a reliability of 0.976. By investing testing resources on the first design of the first subsystem, its reliability may be improved to 0.993. Because all testing resources are allocated to the first design of the first subsystem, the other subsystems are configured by using mature designs at whatever level of redundancy can be affordable using remaining resources.

Table 17: Results for the example with 3 subsystems and 2 or 3 designs in each subsystem

Iteration	Upper bound	Lower bound	Gap	Cumulative time for (116) (sec)	Cumulative time for (118) (sec)
1	0.99995894	0.820817897	1.79×10^{-1}	133	4
100	0.99995894	0.919907607	8.00×10^{-2}	175	9
1×10^4	0.999318772	0.986575814	1.27×10^{-2}	215	42
1×10^6	0.993454496	0.993454542	4.64×10^{-8}	698	185
$\tau_{1,1} = 717.99, \tau_{2,1} = \tau_{3,1} = 1, n_{1,1} = 2, n_{2,3} = 4, n_{3,2} = 2, n_{1,2} = n_{1,3} = n_{2,1} = n_{2,2} = n_{3,1} = 0$					

We now consider an additional example in order to compare a situation when reliability growth and redundancy allocation are planned together versus when they are disjointed. As depicted in Table 18, the example consists of a system with three subsystems and three designs in each subsystem. This five example has a single resource (that must be divided between testing and redundancy) with $b_1 = 5{,}500$, and we consider $E_\ell = M_\ell = 3$, $u_{\ell,i} = 5$ and $\tau^0_{\ell,i} = 1$ for all $\ell = 1,\ldots,3,\ i = 1,\ldots,3$.

Table 18: Parameters for the example with 3 subsystems and 3 designs in each subsystem

	Design 1				Design 2				Design 3			
ℓ	$\lambda_{\ell,1}$	$\beta_{\ell,1}$	$c^1_{\ell,1}$	$d^1_{\ell,1}$	$\lambda_{\ell,2}$	$\beta_{\ell,2}$	$c^1_{\ell,2}$	$d^1_{\ell,2}$	$\lambda_{\ell,3}$	$\beta_{\ell,3}$	$c^1_{\ell,3}$	$d^1_{\ell,3}$
1	0.00005	0.85	15	620	0.00005	0.85	15.5	640	0.00005	0.85	16	660
2	0.00012	0.85	16	220	0.00012	0.85	17	200	0.00012	0.85	18	180
3	0.00012	0.85	15	620	0.00012	0.85	15.5	640	0.00012	0.85	16	660
$T = 8{,}760$												

Table 19 summarizes the example's optimal solution, obtained using Algorithm 8 using a time limit equal to $3{,}600$ seconds with $\varepsilon = 1 \times 10^{-7}$, $\varepsilon'_{\text{sub}} = 1 \times 10^{-2}$, $\varepsilon_{\text{sub}} = 5 \times 10^{-2}$, $P'_{\text{sub}} = 10{,}000$ and $P_{\text{sub}} = 5{,}000$. Both designs $(1,1)$ and $(1,3)$ have the same AMSAA model parameters; however,

the cost of testing and installing components for design $(1,3)$ is more. Therefore, it is more cost effective to have larger τ- and n-values in design $(1,1)$ than design $(1,3)$. The component designs of the first subsystem have smaller λ-values than the third subsystem, and the other parameters are the same. The first subsystem needs less testing to reach a particular reliability, so its τ-values are smaller than the third subsystem. The second subsystem has the same AMSAA model parameters as the third subsystem; however, its cost for installing redundancy is less and cost per unit testing is more. Therefore, the second subsystem has smaller τ-values and larger n-values.

Table 19: Results for the example with 3 subsystems and 3 designs in each subsystem

	Design 1		Design 2		Design 3	
ℓ	$\tau^{\star}_{\ell,1}$	$n^{\star}_{\ell,1}$	$\tau^{\star}_{\ell,2}$	$n^{\star}_{\ell,2}$	$\tau^{\star}_{\ell,3}$	$n^{\star}_{\ell,3}$
1	16.69	2	1	0	1	0
2	1	0	1	0	12.18	5
3	20.96	4	1	0	1	0
	Lower bound: 0.901102		Upper bound: 0.901161		Gap: 5.87×10^{-5}	

Using the instance from Table 18, we now examine what would happen if the redundancy and testing decisions are considered separately. We initially consider 20 scenarios in which the budget of $5,500$ resource units is partitioned into separate budgets for redundancy and testing. For instances $s = 1, \ldots, 20$, we consider $(1,300 + 200s)$ budget for redundancy allocation and $(4,200 - 200s)$ budget for testing of components. We initially fix equal testing time for all designs (such that all designs have the same testing time) and solve the redundancy allocation problem. Then, we solve the reliability growth problem with the solution acquired from redundancy allocation problem. In each iteration, we repeat the algorithm with the solution of the previous iteration (i.e., using the testing times from the previous iteration to initialize solution of the redundancy allocation problem in the current iteration). Then, we compare 20 scenarios with the integrated model in which one resource is considered for both redundancy allocation and testing. We define $\varepsilon = 1 \times 10^{-5}$, $\varepsilon'_{\text{sub}} = 1 \times 10^{-2}$, $\varepsilon_{\text{sub}} = 5 \times 10^{-2}$, $P_{\text{sub}} = 5,000$ and $P'_{\text{sub}} = 15,000$ and set a time limit equal to $3,600$ seconds for all instances. Figure 12 demonstrates a summary of results, and computational times are shown in Figure 13. We display the results when each instance is repeated for one and two iterations, because the solutions of algorithm for more than two iterations are same as its

solutions for two iterations. According to the results, the integrated model increases the system reliability between 0.35% to 172.98% for the instances examined: that is if the project manager decides to allocate $1,700$ budget for redundancy allocation and $1,800$ budget for reliability growth, the system reliability is equal to 0.681322; whereas, the system reliability increases to 0.901102 if we use Model (114) with an integrated constraint for both redundancy allocation and reliability growth. In Figure 12, examining the scenarios where a budget of $4,800$ is allocated to redundancy allocation reveals that taking the maximum objective value over these 20 scenarios results in a system reliability that is close to the one obtained by solving the integrated problem; however, doing this requires much more computation time, as illustrated in Figure 13.

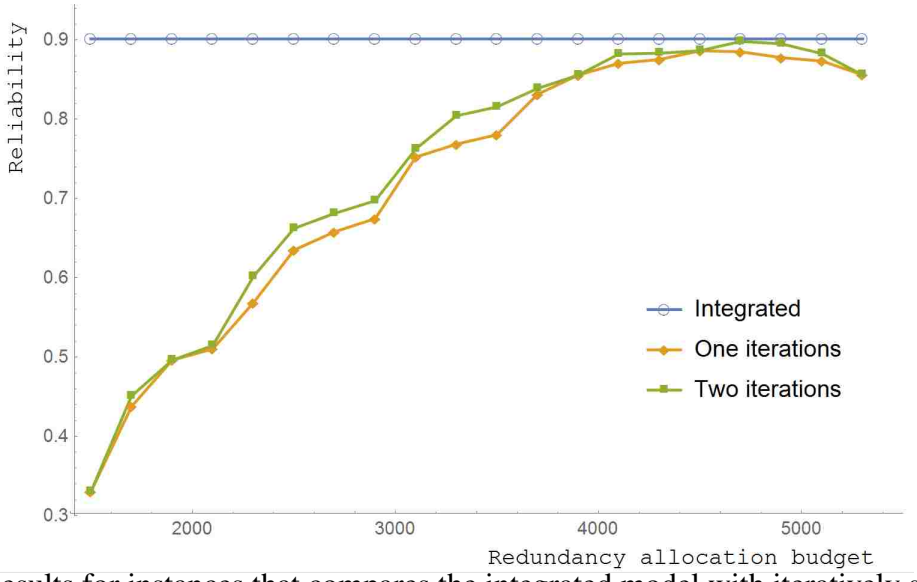


Figure 12: Results for instances that compares the integrated model with iteratively solving reliability growth and redundancy allocation models

Using the parameters from the previous example, we now solve five different instances corresponding to $T \in \{2190, 4380, 6570, 8760, 10950\}$ to obtain optimal testing times $\tau_{\ell,i}$ and the number of redundancies $n_{\ell,i},\ \ell = 1,\ldots,3,\ i = 1,\ldots,3$ for instances $k = 1,\ldots,5$. Using the obtained optimal testing times $\tau_{\ell,i}$ and the optimal number of redundancies $n_{\ell,i}$, we calculate the reliability associated with each value of T. The results are summarized in Figure 14. The obtained optimal number of redundancies when $T \in \{2190, 4380\}$ are equal to $n_{1,1} = 3$, $n_{2,3} = 5$, $n_{3,1} = 4$ and $n_{1,2} = n_{1,3} = n_{2,1} = n_{2,2} = n_{3,2} = n_{3,3} = 0$, and the obtained optimal number of redundancies when $T \in \{6570, 8760, 10950\}$ are equal to $n_{1,1} = 2$, $n_{2,3} = 5$, $n_{3,1} = 4$ and $n_{1,2} = n_{1,3} = n_{2,1} = n_{2,2} =$

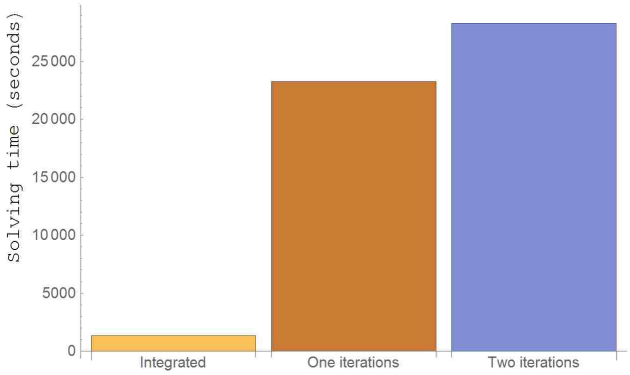


Figure 13: Comparison of computational times between the integrated model and iteratively solving reliability growth and redundancy allocation models

$n_{3,2} = n_{3,3} = 0$. According to the results, the reliabilities corresponding to solutions acquired from $T = 2,190$ and $T = 4,380$ are similar to each other, while the reliabilities of solutions acquired from $T = 6,570$, $T = 8,760$ and $T = 10,950$ are similar. The results suggest that, although optimal redundancy levels are dependent on T, optimal test strategies for two different values of T may be similar if the two values of T result in the same redundancy levels.

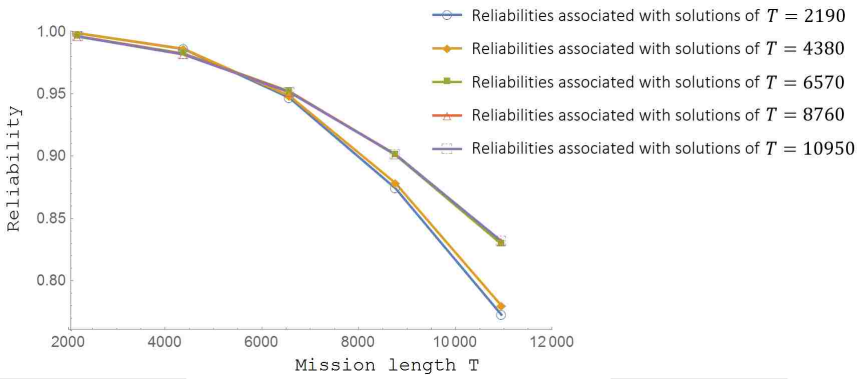


Figure 14: Results of instances that study the effect of mission length T on solutions

To demonstrate the performance of our algorithm, we now report results for 32 instances. In the following, the parameter generation procedure is discussed in detail. For each instance, there is the combination of 2, 3, 4 or 5 subsystems, each with 2, 3, 4 or 5 designs present. We also consider $E_\ell = M_\ell,\ \ell = 1,\ldots,N$ for all instances. The first 16 instances have only a cost constraint, while there exist two resource constraints (cost and testing time) for the next 16 instances. For each component, c- and d-values for the budget constraint are uniformly chosen within intervals $[0.5, 1]$ and $[300, 600]$, and λ-values are selected uniformly within intervals presented in Tables 20–

21. For instances, we consider the available budget $b_1 = 3,000$ and $T = 8,760$. For instances 17 to 32, we define available testing time $b_2 = 360$. In addition, we also select component's U-values from values 4, 5, 6 and 7 and β-values from values 0.65, 0.7, 0.75, 0.8 and 0.85 with the equal probability. We choose $\varepsilon = 1 \times 10^{-7}$, $\varepsilon_{\text{sub}} = 5 \times 10^{-2}$ and $\varepsilon'_{\text{sub}} = 1 \times 10^{-2}$. We consider $P'_{\text{sub}} = 20,000$ for instances 1 to 16 and $P'_{\text{sub}} = 2,000$ for instances 17 to 32. The values of P_{sub} for all instances are represented in Tables 20–21. Table 20 provides a summary of the obtained results for instances 1 to 16, and Table 21 displays a summary of the results for instances 17 to 32. In Tables 20–21, N and M represent number of subsystems and number of designs in each subsystem, respectively. The time limit is set to 3,600 seconds. According to the results, the algorithm provides tight optimality gap for all instances.

Table 20: Computational performance for instances 1–16

Inst.	N	M	λ range	P_{sub}	LB	UB	Gap
1	2	2	[0.0001, 0.00075]	15,000	0.962465	0.962475	1.03×10^{-5}
2	2	3	[0.0001, 0.00025]	15,000	0.985988	0.985994	5.95×10^{-6}
3	2	4	[0.00025, 0.00075]	10,000	0.974904	0.974918	1.35×10^{-5}
4	2	5	[0.00025, 0.00075]	800	0.953065	0.953079	1.43×10^{-5}
5	3	2	[0.00005, 0.00025]	7,000	0.979224	0.979242	1.74×10^{-5}
6	3	3	[0.00005, 0.00015]	7,000	0.949622	0.949650	2.72×10^{-5}
7	3	4	[0.00005, 0.00025]	7,000	0.977787	0.977802	1.54×10^{-5}
8	3	5	[0.0001, 0.00025]	5,000	0.951819	0.951853	3.31×10^{-5}
9	4	2	[0.00001, 0.00005]	5,000	0.971965	0.972007	4.23×10^{-5}
10	4	3	[0.00001, 0.00015]	5,000	0.960397	0.960424	2.67×10^{-5}
11	4	4	[0.00001, 0.0001]	5,000	0.979637	0.979681	4.37×10^{-5}
12	4	5	[0.00001, 0.0001]	3,000	0.941589	0.941641	5.23×10^{-5}
13	5	2	[0.000001, 0.000005]	3,000	0.989030	0.989061	3.01×10^{-5}
14	5	3	[0.000001, 0.00001]	3,000	0.988868	0.988904	3.60×10^{-5}
15	5	4	[0.000005, 0.00001]	3,000	0.979408	0.979489	8.10×10^{-5}
16	5	5	[0.000005, 0.000015]	1,000	0.986053	0.986113	5.97×10^{-5}

4.5 Conclusion and Future Research

This chapter contributes a new model for the problem of maximizing system reliability for a series-parallel system. The system reliability can be improved by both testing components to remove failure modes and installing redundant components. This study merges reliability growth and

Table 21: Computational performance for instances 17–32

Inst.	N	M	λ range	P_{sub}	LB	UB	Gap
17	2	2	[0.0001, 0.00075]	1,000	0.965927	0.965962	3.49×10^{-5}
18	2	3	[0.0001, 0.00025]	500	0.987051	0.987073	2.12×10^{-5}
19	2	4	[0.00025, 0.00075]	600	0.956440	0.956604	1.63×10^{-4}
20	2	5	[0.00025, 0.00075]	500	0.972091	0.972190	9.83×10^{-5}
21	3	2	[0.00005, 0.00025]	500	0.933879	0.934059	1.80×10^{-4}
22	3	3	[0.00005, 0.00015]	500	0.981667	0.981757	8.98×10^{-5}
23	3	4	[0.00005, 0.00025]	600	0.976247	0.976487	2.40×10^{-4}
24	3	5	[0.0001, 0.00025]	500	0.970158	0.970286	1.28×10^{-4}
25	4	2	[0.00001, 0.00005]	400	0.973258	0.973797	5.39×10^{-4}
26	4	3	[0.00001, 0.00015]	400	0.979996	0.980690	6.93×10^{-4}
27	4	4	[0.00001, 0.0001]	500	0.971512	0.972410	8.98×10^{-4}
28	4	5	[0.00001, 0.0001]	600	0.981558	0.981730	1.72×10^{-4}
29	5	2	[0.000001, 0.000005]	200	0.987197	0.987964	7.66×10^{-4}
30	5	3	[0.000001, 0.00001]	400	0.983918	0.984702	7.84×10^{-4}
31	5	4	[0.000005, 0.00001]	300	0.976997	0.977410	4.13×10^{-4}
32	5	5	[0.000005, 0.000015]	300	0.984102	0.984767	6.66×10^{-4}

redundancy allocation concepts. For this problem, we have contributed an exact algorithm which decomposes the problem to a master problem and smaller subproblems. To solve each subproblem, we have extended the algorithm of Chapter 2 into a multi-resource setting, and we have developed a branch and bound method to solve the master problem. We use an example from literature to compare our model with the redundancy allocation problem. Using an example instance, we have demonstrated the benefits in using our model instead of disaggregated redundancy allocation and testing planning models. Numerical experiments show that our proposed method solves (to within $\varepsilon \approx 10^{-4}$) instances up to size 5 subsystems and 5 designs in each subsystem with 2 resources. In addition, the algorithm provides tight optimal gaps for all instances. Our model, which considers the redundancy allocation and reliability growth jointly, improves system reliability over iteratively solving test planning and redundancy allocation problems.

An important point for further investigation is to extend our model for a situation in which the parameters of the AMSAA model are unknown at the beginning of test, and they are learned during the development of the growth test. It also will be important to study the solution methods for large-scale instances. Furthermore, optimization methods incorporating ideas from the redundancy

allocation literature may yields additional efficiency on this class of problems.

5 Conclusion

This dissertation focuses on allocating limited testing resources across the component of a multi-component system to improve system reliability. Complex systems such as weapon systems and aircraft have many components and may develop separately which raises the question of how to allocate the resources within a system to have the most reliable system. In this dissertation, we assume that components are independent and they are connected in an active standby. In addition, each component exhibits reliability growth according to AMSAA reliability growth model. We also consider a fix mission length T for the purpose of evaluating reliability.

In Chapter 2, we consider the resource allocation within a series-parallel system to maximize system reliability, when model's parameters are known. We also extend this model to consider the possibility of testing at different levels (system, subsystem, and component). A class of exact algorithms that decomposes the problem based upon the series-parallel structure is provided. The proposed algorithms quantify the quality of heuristic methods with respect to the optimality gap for the first time in the literature.

In Chapter 3, we extend the first chapter to solve a robust version of this problem, in which the model's parameters are uncertain but assumed to lie within a budget-restricted uncertainty set. We develop and analyze exact solution approaches for this problem based on a cutting plane algorithm for both series and series-parallel systems. A simulation approach is used to compare our solutions from the robust model with those attainable using a deterministic optimization model.

In the forth chapter, we develop a new model that merges two concepts of reliability growth and redundancy allocation to maximize system reliability. We contribute an exact algorithm based on the branch-and-bound method. We also compared the results from our integrated approach with another approach which solves the test planning and redundancy allocation models iteratively.

We have extended this dissertation to consider a multi-stage reliability growth testing for a series-parallel system. The failure information of testing is incorporated to have a better estimation of model's parameters in the next testing stage. In another extension for this dissertation, we

have developed a model for resource allocation in reliability growth within the components of a series-parallel system when components are connected in a cold-standby redundancy. Future research may study the resource allocation when the reliabilities of components are dependent. Another future direction would be to model this problem when the mission reliability growth T is indeterminate.

References

[1] M. Abouei Ardakan, M. Sima, A. Zeinal Hamadani, and D. W. Coit. A novel strategy for redundant components in reliability–redundancy allocation problems. *IIE Transactions*, 48(11):1043–1057, 2016.

[2] M. Awad. Economic allocation of reliability growth testing using Weibull distributions. *Reliability Engineering & System Safety*, 152:273–280, 2016.

[3] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear programming: Theory and algorithms, third edition*. John Wiley & Sons, 2006.

[4] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88(3):411–424, 2000.

[5] A. Ben-Tal and A. Nemirovski. Selected topics in robust convex optimization. *Mathematical Programming*, 112(1):125–158, 2008.

[6] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.

[7] H. P. Benson. An outcome space branch and bound-outer approximation algorithm for convex multiplicative programming. *Journal of Global Optimization*, 15(4):315–342, 1999.

[8] D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011.

[9] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming*, 98(1):49–71, 2003.

[10] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[11] D. Bertsimas and A. Thiele. A robust optimization approach to inventory theory. *Operations Research*, 54(1):150–168, 2006.

[12] J.-E. Byun, H.-M. Noh, and J. Song. Reliability growth analysis of k-out-of-n systems using matrix-based system reliability method. *Reliability Engineering & System Safety*, 165:410–421, 2017.

[13] M. S. Chern. On the computational complexity of reliability redundancy allocation in a series system. *Operations Research Letters*, 11(5):309–315, 1992.

[14] M. S. Chern and R. H. Jan. Reliability optimization problems with multiple constraints. *IEEE Transactions on Reliability*, 35(4):431–436, 1986.

[15] D. W. Coit. Economic allocation of test times for subsystem-level reliability growth testing. *IIE Transactions*, 30(12):1143–1151, 1998.

[16] L. H. Crow. Reliability analysis for complex, repairable systems. In *Reliability and Biometry, Proschan, F. and Serfling, R.J. (Eds.), SIAM, Philadelphia, PA*, pages 379–410, 1974.

[17] L. H. Crow. An extended reliability growth model for managing and assessing corrective actions. In *Reliability and Maintainability Symposium Annual Symposium-RAMS, Los Angeles, CA*, pages 73–80, 2004.

[18] Y. S. Dai, M. Xie, K. L. Poh, and B. Yang. Optimal testing-resource allocation with genetic algorithm for modular software systems. *Journal of Systems and Software*, 66(1):47–55, 2003.

[19] J. T. Duane. Learning curve approach to reliability monitoring. *IEEE Transactions on Aerospace*, 2(2):563–566, 1964.

[20] L. El-Ghaoui, F. Oustry, and H. Lebret. Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9(1):33–52, 1998.

[21] A. O. C. Elegbede, C. Chu, K. H. Adjallah, and F. Yalaoui. Reliability allocation through cost minimization. *IEEE Transactions on Reliability*, 52(1):106–111, 2003.

[22] M. J. Feizollahi, S. Ahmed, and M. Modarres. The robust redundancy allocation problem in series-parallel systems with budgeted uncertainty. *IEEE Transactions on Reliability*, 63(1):239–250, 2014.

[23] M. J. Feizollahi, R. Soltani, and H. Feyzollahi. The robust cold standby redundancy allocation in series-parallel systems with budgeted uncertainty. *IEEE Transactions on Reliability*, 64(2):799–806, 2015.

[24] D. P. Gaver and P. A. Jacobs. Reliability growth by failure mode removal. *Reliability Engineering & System Safety*, 130:27–32, 2014.

[25] A. L. Goel and K. Okumoto. Time-dependent error-detection rate model for software reliability and other performance measures. *IEEE transactions on Reliability*, R-28(3):206–211, 1979.

[26] J. B. Hall and A. Mosleh. An analytical framework for reliability growth of one-shot systems. *Reliability Engineering & System Safety*, 93(11):1751–1760, 2008.

[27] M. Hikita, Y. Nakagawa, K. Nakashima, and H. Narihisa. Reliability optimization of systems by a surrogate-constraints algorithm. *IEEE Transactions on Reliability*, 41(3):473–480, 1992.

[28] J. H. Kim and B. J. Yum. A heuristic method for solving redundancy optimization problems in complex systems. *IEEE Transactions on Reliability*, 42(4):572–578, 1993.

[29] T. Kim, K. Lee, and J. Baik. An effective approach to estimating the parameters of software reliability growth models using a real-valued genetic algorithm. *Journal of Systems and Software*, 102:134–144, 2015.

[30] P. Kouvelis and G. Yu. *Robust discrete optimization and its applications*, volume 14. Springer Science & Business Media, 2013.

[31] M. Krasich. Accelerated reliability growth testing and data analysis method. *Journal of the IEST*, 50(2):98–117, 2007.

[32] W. Kuo, H. H. Lin, Z. Xu, and W. Zhang. Reliability optimization with the Lagrange-multiplier and branch-and-bound technique. *IEEE Transactions on Reliability*, R-36(5):624–630, 1987.

[33] W. Kuo and V. R. Prasad. An annotated overview of system-reliability optimization. *IEEE Transactions on Reliability*, 49(2):176–187, 2000.

[34] W. Kuo, V. R. Prasad, F. A. Tillman, and C. L. Hwang. *Optimal reliability design: fundamentals and applications*. Cambridge university press, 2001.

[35] W. Kuo and R. Wan. Recent advances in optimal reliability allocation. In *Computational Intelligence in Reliability Engineering*, pages 1–36. Springer, 2007.

[36] G. Levitin. Allocation of test times in multi-state systems for reliability growth testing. *IIE Transactions*, 34(6):551–558, 2002.

[37] D. C. Li, F. M. Chang, and K. C. Chen. Building reliability growth model using sequential experiments and the Bayesian theorem for small datasets. *Expert Systems with Applications*, 37(4):3434–3443, 2010.

[38] Z. Li, M. Mobin, and T. Keyser. Multi-objective and multi-stage reliability growth planning in early product-development stage. *IEEE Transactions on Reliability*, 65(2):769–781, 2016.

[39] D. K. Lloyd. Forecasting reliability growth. *Quality and Reliability Engineering International*, 2(1):19–23, 1986.

[40] M. Mobin, Z. Li, and G. Komaki. A multiobjective approach for multistage reliability growth planning by considering the timing of new technologies introduction. *IEEE Transactions on Reliability*, 66(1):97–110, 2017.

[41] J. R. O'Hanley, M. P. Scaparra, and S. García. Probability chains: A general linearization technique for modeling reliability in facility location and related problems. *European Journal of Operational Research*, 230(1):63–75, 2013.

[42] M. Pecht. *Product reliability, maintainability, and supportability handbook*. CRC Press, 2009.

[43] E. A. Pohl and D. L. Dietrich. Optimal stress screening strategies for multi-component systems sold under warranty: The case of phase-type lifetimes. *Annals of Operations Research*, 91:137–161, 1999.

[44] J. E. Ramirez-Marquez, D. W. Coit, and A. Konak. Redundancy allocation for series-parallel systems using a max-min approach. *IIE Transactions*, 36(9):891–898, 2004.

[45] R. Rana, M. Staron, C. Berger, J. Hansson, M. Nilsson, F. Törner, W. Meding, and C. Höglund. Selecting software reliability growth models and improving their predictive accuracy using historical projects data. *Journal of Systems and Software*, 98:59–78, 2014.

[46] S. S. Rao. *Engineering optimization: Theory and practice, fourth edition*. John Wiley & Sons, 2009.

[47] D. G. Robinson and D. Dietrich. A new nonparametric growth model. *IEEE Transactions on Reliability*, R-36(4):411–418, 1987.

[48] R. Romera, J. E. Valdés, and R. I. Zequeira. Active-redundancy allocation in systems. *IEEE Transactions on Reliability*, 53(3):313–318, 2004.

[49] A. Sofer and D. R. Miller. A nonparametric software-reliability growth model. *IEEE Transactions on Reliability*, 40(3):329–337, 1991.

[50] A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5):1154–1157, 1973.

[51] K. M. Sullivan. Mixed-integer linear models for reliable system design. *Technical Report*.

[52] C. S. Sung and Y. K. Cho. Branch-and-bound redundancy optimization for a series system with multiple-choice constraints. *IEEE Transactions on Reliability*, 48(2):108–117, 1999.

[53] T. P. Talafuse and E. A. Pohl. A bat algorithm for the redundancy allocation problem. *Engineering Optimization*, 48(5):900–910, 2016.

[54] T. P. Talafuse and E. A. Pohl. Small sample reliability growth modeling using a grey systems model. *Quality Engineering*, 29(3):455–467, 2017.

[55] Z. Tian, M. J. Zuo, and H. Huang. Reliability-redundancy allocation for multi-state series-parallel systems. *IEEE Transactions on Reliability*, 57(2):303–310, 2008.

[56] Y. Tohma, K. Tokunaga, S. Nagase, and Y. Murata. Structural approach to the estimation of the number of residual software faults based on the hyper-geometric distribution. *IEEE Transactions on Software Engineering*, 15(3):345–355, 1989.

[57] C. R. Torres, M. Heydari, K. M. Sullivan, H. Liao, and E. A. Pohl. Data analysis and resource allocation in bayesian selective accelerated reliability growth. *Technical Report*.

[58] M. Wayne and M. Modarres. A Bayesian model for complex system reliability growth under arbitrary corrective actions. *IEEE Transactions on Reliability*, 64(1):206–220, 2015.

[59] K. J. Wilson and J. Quigley. Allocation of tasks for reliability growth using multi-attribute utility. *European Journal of Operational Research*, 255(1):259–271, 2016.

[60] A. Yalaoui, C. Chu, and E. Chatelet. Reliability allocation problem in a series–parallel system. *Reliability Engineering & System Safety*, 90(1):55–61, 2005.

[61] W. C. Yeh and T. J. Hsieh. Solving reliability redundancy allocation problems using an artificial bee colony algorithm. *Computers & Operations Research*, 38(11):1465–1473, 2011.

[62] P. S. You and T. C. Chen. An efficient heuristic for series–parallel redundant reliability problems. *Computers & Operations Research*, 32(8):2117–2127, 2005.

[63] L. Zia and D. W. Coit. Redundancy allocation for series-parallel systems using a column generation approach. *IEEE Transactions on Reliability*, 59(4):706–717, 2010.