Analysis and Optimization of Test Plans for Advanced Exploration Systems Reliability and Supportability

Andrew C. Owens,[1] and William M. Cirillo[2]
NASA Langley Research Center, Hampton, VA 23681, USA

Nicole Piontek,[3] Chel Stromgren[4] and Jason Cho[5]
Binera, Inc., Rockville, MD 20850, USA

Future crewed exploration missions beyond Low Earth Orbit (LEO) will operate farther from Earth and be logistically isolated for longer than any previous human spaceflight mission. Under these conditions, supportability and reliability will be stronger drivers of mission mass and risk than they have been in the past. Items with high failure rates, or uncertain failure rates, can result in high spares mass requirements and/or high risk on deep space missions. Testing is a critical element of system development which provides the opportunity to identify and resolve design issues, defects, or other failure modes before they cause problems during a mission. Reliability growth programs can reduce failure rates by identifying and remove failure modes via design changes, and long-duration life testing can provide valuable data to reduce failure rate estimate uncertainty and verify (to some level of confidence) that components are as reliable as expected. Testing activities take time and resources, however, and must be incorporated into program plans in order to be fully effective. This paper presents an integrated reliability test plan analysis and optimization methodology, which has been used to inform Advanced Exploration Systems (AES) Life Support Systems (LSS) ground test planning for future missions. The methodology determines the optimal number of test units to purchase and allocation of test time – split between reliability growth and uncertainty reduction testing – across a given set of items in order to minimize spares mass for a given mission under constraints on total test cost and schedule. Model outputs also include expected spares mass after testing and the expected number of modifications or refurbishments during testing, both of which can inform program planning. Discussion of the model, conclusions, and future work are also presented.

Nomenclature

$\forall$	For all
$\in$	In (i.e. set membership)
α	Gamma distribution shape parameter
β	Gamma distribution scale parameter
β_D	Discovery rate
$\Gamma(\cdot)$	Gamma function
Δt	Reliability growth test time step size
ζ	Expected number of modifications
λ	Failure rate
$\bar{\lambda}$	Expected failure rate
λ_0	Initial failure rate
λ_A	Type A failure rate

[1] Aerospace Engineer, Space Mission Analysis Branch, 1 N. Dryden St. MS 462
[2] Senior Researcher, Space Mission Analysis Branch, 1 N. Dryden St. MS 462
[3] Junior Reliability Analyst, 77 S. Washington St. (at time of writing)
[4] Vice President / Chief Scientist, 77 S. Washington St.
[5] Aerospace Engineer, 77 S. Washington St.

λ_B	Type B failure rate
λ_D	Discovery scaling parameter
λ_g	Failure rate after reliability growth testing
τ	Maximum total test time
τ_ζ	Modification time
τ_d	Test unit procurement delay time
τ_g	Maximum reliability growth test time
τ_r	Refurbishment time
$\psi(\cdot)$	Approximate spares mass function
c	Maximum total test cost
c_u	Test unit cost
c_ζ	Modification cost
c_r	Refubishment cost
FEF	Fix Effectiveness Factor
g	Number of time steps beyond the transition time used for reliability growth testing
$h(\cdot)$	Discovery rate function
k	Number of test units
MS	Management Strategy
r_G	Expected number of refurbishments during reliability growth testing
r_U	Expected number of refurbishments during uncertainty reduction testing
t	Time
t_h	Reliability growth transition time
x	Binary test indicator variable
x_g	Binary reliability growth test indicator variable
x_u	Binary uncertainty reduction test indicator variable
AES	Advanced Exploration Systems
AMSAA	Army Materiel Systems Analysis Activity
ECLSS	Environmental Control and Life Support Systems
LEO	Low Earth Orbit
LSS	Life Support Systems
MTBF	Mean Time Between Failures
POS	Probability of Sufficiency

I. Introduction

FUTURE crewed exploration missions beyond Low Earth Orbit (LEO) will operate farther from Earth and be logistically isolated for longer than any previous human spaceflight missions. Supportability and reliability will be stronger drivers of mass and risk than they have been in the past, and item failure rates are therefore more important considerations during concept development.[1–3] Unfortunately, an item's failure rate is not a parameter that can be measured directly, because it is not a physical characteristic of the item. Instead, it is a parameter describing the distribution of the amount of time that the item will operate before experiencing a random failure. As a result, failure rates are inherently uncertain estimates. This uncertainty results in higher risk and/or higher spares mass requirements, and is a key challenge for deep space mission supportability.[4,5] Testing and operational experience can be used to identify and remove failure modes to improve reliability, or to reduce uncertainty.[6,7] However, an effective reliability and supportability testing program can require many years of operations and resource investment.[8] Testing must be carefully incorporated into program plans in order to be fully effective.

This paper describes a reliability and supportability test planning and optimization capability that has been developed to inform Advanced Exploration Systems (AES) Life Support Systems (LSS) ground test planning. The model considers two kinds of testing. Reliability growth testing involves operating the system to observe failure modes and make adjustments to the design to try to remove them. Uncertainty reduction, on the other hand, involves operating the system to gather data and refine failure rate estimates without making changes to the design. The amount of each type of testing for each item, as well as the number of test units to purchase for parallel tests, is optimized for each item in order to minimize the total spares mass required for a desired Probability of Sufficiency (POS) on a given reference mission within given test budget and schedule constraints. In addition to this optimal test plan, the model also calculates the expected number of modifications and refurbishments during testing, both of which impact cost and schedule and are important considerations during test planning. The following sections present the methodology in detail, as well as discussion, future work, and conclusions.

II. Methodology

The test plan model and optimizer use gamma-distributed, uncertain failure rate estimates, as has been done in previous work.[3,6–8] It incorporates reliability growth modeling, projected Bayesian updating, and spares allocation optimization within an overarching branch-and-bound optimization routine to identify optimal test plans. A "test plan," for the purposes of this model, is defined by the amount of reliability growth testing, the amount of uncertainty reduction testing, and the number of test units purchased, with each variable defined for each item. The test optimization assumes that all reliability growth testing is completed before any uncertainty reduction testing occurs, and that these two test phases are distinct and static. In practice, the distinction between these two types of testing may not be as sharp, and test operations can and should be more responsive to observed data. Data from reliability growth testing could be used to reduce uncertainty, and fixes could be attempted during uncertainty reduction testing. However, for modeling purposes, the two types of testing are treated separately.

Optimal test plans, as determined by this model, are not intended to be a final answer, and should not be set in stone. Instead, the test plan should be adjusted to account for expert knowledge of system designs, such as suspected weak points or historical lessons learned. In addition, the test plan should be a living plan that updates as new information becomes available. Projected reliability growth and uncertainty reduction can and should be updated as more information becomes available, and test resources should be rebalanced accordingly. A dynamic, responsive test plan that makes use of new information as it becomes available will almost certainly outperform a static test plan.

A. Modeling Test Impacts

Reliability growth testing and uncertainty reduction testing are modeled as separate activities which occur in that order. The sections below describe the models used to project the impact of test activities on failure rate estimates, either in terms of the expected failure rate or the uncertainty in the estimate. It is important to note that both of these models are projections, not predictions. In practice, actual test data should be used to constantly update failure rate estimates as the test is executed.

1. Reliability Growth

The Extended Reliability Growth Planning Model[9] is used to model the relationship between reliability growth test time and the projected change in failure rate. This parametric, empirical model is based on the Crow-AMSAA (Army Materiel Systems Analysis Activity) Reliability Growth Model,[9] which is in turn developed from the Duane Postulate,[10] an observation that the cumulative failure rate during reliability growth testing tends to follow a power law relationship relative to test time. This is also known as the learning curve model. Under the Extended Reliability Growth Planning Model, the projected failure rate as a function of reliability growth test time t is

$$\lambda(t) = \lambda_A + (1 - FEF)\lambda_B + FEFh(t) \tag{1}$$

where t is the reliability growth test time, and $\lambda(t)$ is the failure rate at time t, and FEF is the Fix Effectiveness Factor (FEF). λ_A and λ_B are the failure rates associated with Type A and Type B failure modes, respectively, calculated as

$$\lambda_A = (1 - MS)\lambda_0 \tag{2}$$

$$\lambda_B = MS\lambda_0 \tag{3}$$

where λ_0 is the initial (pre-test) failure rate and MS is the Management Strategy (MS). Type A failure modes are failure modes that will not be corrected even if observed during testing, while Type B failure modes are ones that will be corrected if observed during testing. The discovery rate function $h(t)$ describes the instantaneous failure rate associated with Type B failure modes and has the form

$$h(t) = \lambda_D \beta_D t^{1-\beta_D} \tag{4}$$

where λ_D is the discovery scaling parameter, β_D is the discovery rate. The discovery scaling parameter can be calculated from the Type B failure rate and discovery rate:[9]

$$\lambda_D = \left(\lambda_B \Gamma\left(1 + \frac{1}{\beta_D}\right)\right)^{\beta_D} \quad (5)$$

FEF, MS, and discovery rate β_D are parameters governing the shape of the reliability growth curve which are typically set based on historical experience. FEF represents the average fraction of a failure mode's rate that is removed by implementing a fix during the reliability growth program. In the limit, this is the fraction of the Type B failure rate that will be removed during the reliability growth program if it were given infinite time and resources. For example, if an item has a Type B failure rate of 1×10^{-5} h^{-1} and the estimated FEF is 0.7, then in the limit the remaining failure rate resulting from Type B failure modes after infinite testing would be 3×10^{-6} h^{-1} (i.e. 30% of the initial value). This reduction in failure rate does not happen all at once, nor is it all associated with a single failure mode. In practice, different fixes may have different levels of effectiveness, and may even result in an increase in failure rate. On average, however, fixes implemented during a reliability growth program are expected to reduce the failure rate for the associated mode by the FEF. This analysis currently uses an FEF of 0.7 for all items as a starting point, because this value is in line with actual FEF from military reliability growth programs reported by Crow.[9] MS represents the fraction of the total initial failure rate which is associated with Type B failure modes, indicating a management decision regarding what portion of the failure rate to address during testing. The recommended MS value for reliability growth planning (which is used in this analysis) is 0.95. The discovery rate β_D determines the shape of the learning curve representing improvement in failure rate as reliability growth testing is performed. Based on reported experience in military and commercial reliability growth programs, this analysis assumes a discovery rate of 0.7.[9] All of these parameters can and should be adjusted based on data from relevant test and operations experience.

The first term in equation 1 (λ_A) represents the failure rate associated with Type A failure modes, which are not addressed during reliability growth testing and therefore are unchanged. The second term, $(1 - FEF)\lambda_B$, represents the remaining failure rate associated with Type B failure modes after all fixes have been implemented. Finally, the third term, $FEFh(t)$, represents the failure rate associated with Type B failure modes that have not yet been addressed with a fix.

One drawback of this model is that it can forecast failure rates higher than the current failure rate for short tests. To address this issue, this analysis adjusts the discovery rate function as follows:

$$h(t) = \begin{cases} \lambda_B & t \le t_h \\ \lambda_D \beta_D t^{\beta_D - 1} & t > t_h \end{cases} \quad (6)$$

The transition time t_h is the time at which the original discovery rate function is equal to λ_B.

$$t_h = \left(\frac{\lambda_B}{\lambda_D \beta_D}\right)^{\frac{1}{\beta_D - 1}} \quad (7)$$

This adjustment to the model captures the reality that reliability growth testing has no impact on an item's failure rate until a Type B failure mode is observed and the associated fix is implemented. The transition time can be thought of as the amount of reliability growth testing required to have an appreciable effect on the item's failure rate. During reliability growth testing, the failure rate remains unchanged prior to this time.

2. Uncertainty Reduction

Uncertainty reduction testing is the process of observing the time to failure for a given item in order to reduce the level of epistemic uncertainty in the failure rate estimate. One or more test units are operated, and when a failure occurs the accumulated operating time is recorded, the failed unit is replaced with an identical test unit, and testing continues. In contrast to reliability growth testing, uncertainty reduction testing does not involve design modifications to mitigate observed failure modes. Uncertainty reduction testing changes our knowledge of the failure rate of a particular item after that item's design has been frozen; it does not change the item's failure rate.

This analysis models uncertainty reduction testing using the same mathematical approach used for Bayesian updating, which is described in detail by Dezfuli et al.[11] and in previous work.[7] The model assumes that the expected number of failures occurs during uncertainty reduction testing, based on the failure rate estimate that the start of that test. Under this assumption, the expected failure rate remains unchanged, but the level of uncertainty in the failure rate estimate is reduced as more failures are observed. Under these assumptions, the parameters of the gamma-distributed failure rate estimate as a function of uncertainty reduction test time are:

$$\alpha(t) = \alpha_0 + \bar{\lambda}t \tag{8}$$

$$\beta(t) = \beta_0 + t \tag{9}$$

Note that these are simply the standard equations for Bayesian updating of a gamma-distributed failure rate with the number of failures replaced with the expected number of failures $\bar{\lambda}t$.

B. Modeling Failures and Modifications During Testing

The test model also forecasts the expected number of failures during testing. These are split into two categories: modifications and refurbishments. Modifications are failures associated with Type B failure modes during reliability growth testing – that is, failures that will result in a fix that changes the design of the item. No modifications occur during uncertainty reduction testing. Refurbishments, on the other hand, are failures that do not result in changes to the item design. Instead, the unit is returned to working order and put back on the test stand. During reliability growth testing, failures associated with Type A failure modes or Type B failure modes that have already been addressed with a fix are considered refurbishments. All failures during uncertainty reduction testing are refurbishments. The expected number of modifications and refurbishments provides useful information for test planning. In particular, the time, cost, and labor required for modifications and refurbishments should be factored into test scheduling and budgeting. It is important to note, however, that this is simply the expected value; the actual number of modifications and refurbishments that occur during testing may be higher or lower, and cannot be known *a priori*.

The expected number of modifications during a given reliability growth test period, denoted ζ, is equal to the expected number of failures resulting from unobserved Type B failure modes, which is the integral of the discovery rate function:

$$\zeta(t) = \int_0^t h(x)dx \tag{10}$$

Assuming a reliability growth test longer than the transition time (i.e. $t > t_h$), this yields

$$\zeta(t) = \lambda_B t_h + \lambda_D (t - t_h)^{\beta_D} \tag{11}$$

The expected number of refurbishments during a reliability growth test, denoted r_G, is equal to the expected number of failures from Type A failure modes or Type B failure modes that have already been addressed with a fix. This is equivalent to the total number of failures during reliability growth testing minus the number of refurbishments.

$$r_g(t) = \int_0^t \lambda(x) - h(x)dx \tag{12}$$

Assuming a reliability growth test longer than the transition time, this yields

$$r_g(t) = \lambda_A t + (1 - FEF)\lambda_B t + (1 - FEF)\left(\lambda_B t_h + \lambda_D (t - t_h)^{\beta_D}\right) \tag{13}$$

The expected number of refurbishments during uncertainty reduction, denoted r_U, is more straightforward to calculate because the failure rate is constant during the test.

$$r_u(t) = \bar{\lambda}t \tag{14}$$

C. Test Plan Optimization

The objective of test plan optimization is to identify the test plan that makes the best use of limited schedule and budget. This analysis uses the approximate spares mass required for a reference mission as the objective function to be minimized within constraints on the total calendar time available for reliability growth and uncertainty reduction testing and the total cost of testing. The key decision variables in the optimization are the amount of reliability growth test time, the amount of uncertainty reduction test time, and the number of test units. For the purposes of optimization, test time is discretized and, for each item, uncertainty reduction testing occurs after all reliability growth testing for

Problem 1: Test optimization problem formulation, defining one objective ($O1$) and 17 constraints ($C1$-$C17$).

$$\text{Minimize} \quad \psi(g, r_u) \qquad (O1)$$

$$\text{s.t.} \quad \sum_i \left(k_i c_{ui} + \zeta_i c_{\zeta i} + \left(r_{gi} + r_{ui} - x_{ui}\right) c_{ri}\right) \leq c \qquad (C1)$$

$$x_{gi}\tau_{di} + \frac{1}{k_i}\left(x_{gi}t_{hi} + g_i\Delta t + \zeta_i\tau_{\zeta i} + r_{gi}\tau_{ri}\right) \leq \tau_{gi} \quad \forall i \qquad (C2)$$

$$x_i\tau_{di} + \frac{1}{k_i}\left(x_{gi}t_{hi} + g_i\Delta t + \zeta_i\tau_{\zeta i} + \frac{r_{ui}}{\lambda_{gi}} + \left(r_{gi} + r_{ui} - x_{ui}\right)\tau_{ri}\right) \leq \tau_i \quad \forall i \qquad (C3)$$

$$\lambda_{gi} = \lambda_{Ai} + (1 - FEF_i)\lambda_{Bi} + FEF_i\lambda_{Di}\beta_{Di}\left(x_{gi}t_{hi} + g_i\Delta t\right)^{1-\beta_{Di}} \quad \forall i \qquad (C4)$$

$$\zeta_i = x_{gi}\lambda_{Bi}t_{hi} + \lambda_{Di}(g_i\Delta t)^{\beta_{Di}} \quad \forall i \qquad (C5)$$

$$r_{gi} = (\lambda_{Ai} + (1 - FEF_i)\lambda_{Bi})\left(x_{gi}t_{hi} + g_i\Delta t\right) + (FEF_i - 1)\left(x_{gi}\lambda_{Bi}t_{hi} + \lambda_{Di}(g_i\Delta t)^{\beta_{Di}}\right) \quad \forall i \qquad (C6)$$

$$g_i \leq Mx_{gi} \quad \forall i \qquad (C7)$$

$$x_{gi} \leq g_i \quad \forall i \qquad (C8)$$

$$r_{ui} \leq Mx_{ui} \quad \forall i \qquad (C9)$$

$$x_{ui} \leq r_{ui} \quad \forall i \qquad (C10)$$

$$x_{gi} \leq x_i \quad \forall i \qquad (C11)$$

$$x_{ui} \leq x_i \quad \forall i \qquad (C12)$$

$$x_i \leq x_{gi} + x_{ui} \quad \forall i \qquad (C13)$$

$$k_i \leq Mx_i \quad \forall i \qquad (C14)$$

$$x_i \leq k_i \quad \forall i \qquad (C15)$$

$$x_i, x_{gi}, x_{ui} \in \{0, 1\} \quad \forall i \qquad (C16)$$

$$k_i, g_i, r_{ui} \in \mathbb{Z}_{\geq 0} \quad \forall i \qquad (C17)$$

Table 1: Description of objective function, decision variables, and parameters used in test plan optimization. Variables that appear with a subscript represent the value of that variable associated with a particular replaceable item; when that variable appears without a subscript, it represents the vector containing values for all replaceable items. All variables are nonnegative real numbers; some are integers, and some are binary, as described by the domains below.

Symbol & Domain	***Description***
$\psi(g, r_u) \geq 0$	Approximate spares mass as a function of the number of reliability growth test timesteps and number of failures/refurbishments planned for uncertainty reduction testing. *(Objective Function)*
$g \in \mathbb{Z}_{\geq 0}$	Number of timesteps beyond t_h used for reliability growth testing. *(Decision Variable)*
$r_u \in \mathbb{Z}_{\geq 0}$	Number of planned failures/refurbishments during uncertainty reduction testing. *(Decision Variable)*
$k \in \mathbb{Z}_{\geq 0}$	Number of units tested in parallel. *(Decision Variable)*
$x_i \in \{0, 1\}$	Binary indicator of whether or not testing occurs. *(Auxiliary Variable)*
$x_g \in \{0, 1\}$	Binary indicator of whether or not reliability growth testing occurs. *(Auxiliary Variable)*
$x_u \in \{0, 1\}$	Binary indicator of whether or not uncertainty reduction testing occurs. *(Auxiliary Variable)*
$c \geq 0$	Maximum total test cost.
$c_u \geq 0$	Unit cost.
$c_\zeta \geq 0$	Modification cost.
$c_r \geq 0$	Refurbishment cost.
$\tau \geq 0$	Maximum total test time.
$\tau_g \geq 0$	Maximum reliability growth test time.
$\tau_d \geq 0$	Test unit procurement delay time.
$\tau_\zeta \geq 0$	Modification time.
$\tau_r \geq 0$	Refurbishment time.
$\Delta t \geq 0$	Reliability growth testing time step size.
$\zeta \geq 0$	Expected number of modifications during reliability growth testing.
$r_g \geq 0$	Expected number of refurbishments during reliability growth testing.
$\lambda_g \geq 0$	Failure rate after reliability growth testing.
$t_h \geq 0$	Reliability growth discovery rate function transition time.
$\lambda_A \geq 0$	Failure rate associated with Type A failure modes.
$\lambda_B \geq 0$	Failure rate associated with Type B failure modes.
$\lambda_D \geq 0$	Reliability growth discovery rate function scaling parameter.
$0 \leq \beta_{Di} \leq 1$	Reliability growth discovery rate.
$0 \leq FEF \leq 1$	Reliability growth fix effectiveness factor.
$M \gg 0$	Large constant used to enforce constraints.

that item is complete. The result is a single-objective integer optimization problem, defined in Problem 1 on page 7. The objective function and all decision variables and parameters are defined in Table 1 on the same page. The various objectives, constraints, and other characteristics of the problem are described in detail in the following paragraphs.

1. Objective Function

The objective function ($O1$) is the approximate spares mass for a reference mission as a function of the number of time steps used for reliability growth testing g and the number of failures that are planned to be observed during uncertainty reduction testing, represented by the function $\psi(g, r_u)$. The reference mission is defined by a mission endurance and a POS requirement. This function is itself an optimization problem, and approximate spares mass is calculated using the optimization approach described in previous work.[3] The infimum (i.e. largest lower bound[12]) mass from the first round of marginal analysis, a lower bound on the true spares mass, is used as an approximation because it can be calculated quickly and is monotonic with regard to replaceable item failure rates and uncertainties. The objective of the optimization is to identify the assignment to the decision variables (described below) that minimizes this approximate spares mass.

2. Decision Variables

The key decision variables are the number of time steps used for reliability growth testing for each replaceable item g, the number of failures planned to be observed in uncertainty reduction testing for each item r_u, and the number of units of each item tested in parallel k. All of these decision variables are positive integers. In addition, these decision variables, as well as the auxiliary variables and many of the parameters, and are vector-valued. When the variable appears without a subscript, it represents the full vector of values for all replaceable items; subscripts are used to indicate the values associated with specific replaceable items. For example, k_i indicates the number of test units for item i, while k represents the vector indicating the number of test units for all items.

Three auxiliary binary variables are also introduced to enforce logical constraints on testing. The overall test indicator variables x_i are equal to 1 if any testing (reliability growth or uncertainty reduction) occurs for item i, and 0 otherwise. The reliability growth testing indicator variable x_g and the uncertainty reduction indicator variable x_u indicate the same for reliability growth and uncertainty reduction testing, respectively. The values of these auxiliary variables are related to the key decision variables by constraints $C7$ through $C15$. Three parameters are also defined in order to simplify the notation, including the failure rate after reliability growth testing λ_g ($C4$), the expected number of modifications during reliability growth testing ζ ($C5$), and the expected number of refurbishments during reliability growth testing r_g ($C6$).

The reliability growth and uncertainty reduction test times are represented using discrete, integer-valued variables for mathematical convenience. The number of reliability growth time steps g indicates the number of time steps beyond the transition time t_h used for reliability growth testing. Each time step represents a duration of length Δt, a parameter which determines the granularity of the analysis. Analyses performed using smaller Δt will be more precise, but will require more computation time to solve. The total reliability growth test time t_g, given an assignment to the decision variables, is

$$t_g = x_g t_h + g\Delta t \tag{15}$$

Uncertainty reduction test planning is parameterized in terms of r_u, the expected number of failures during uncertainty reduction testing. This discretization is less straightforward than the discrete time steps used for reliability growth testing, as it represents a variable amount of time that changes depending on changes to the failure rate resulting from reliability growth. The uncertainty reduction test model described above assumes that the expected number of failures occurs during uncertainty reduction testing. In practice, uncertainty reduction actually occurs when a failure is observed during testing, because that event provides a data point to improve reliability estimates. Therefore, it is natural to parametrize uncertainty reduction testing by the number of failures that test is intended to observe. The time between failures is estimated using the MBTF, which is the inverse of the failure rate after reliability growth testing, and therefore the total uncertainty reduction test time is

$$t_u = \frac{r_u}{\lambda_g} \tag{16}$$

3. Constraints

The cost constraint ($C1$) indicates that the total test cost must be less than the maximum cost allocated for testing, c. The test cost associated with each item is equal to the cost of the test units themselves plus the cost of modifications and the cost of refurbishments, where the unit cost, modification cost, and refurbishment cost are defined by the parameters c_u, c_ζ, and c_r, respectively. The total test cost is the sum of the test cost associated with each item. It is important to note that this formulation of the model does not include the cost of the refurbishment that would be associated with the last failure during uncertainty reduction testing, as indicated by the $-x_{ui}$ term in constraint $C1$. (This term will be equal to 1 when $r_{ui} > 0$, and 0 otherwise.) The rationale for this formulation is that the benefits of uncertainty reduction testing – that is, the reduction of uncertainty in the failure rate estimate – is achieved as soon as the failure occurs, because the required information is the time to failure. Therefore, the test unit that exhibits the last failure does not necessarily have to be refurbished to complete the test. In contrast, the benefits of reliability growth testing are attained by making modifications to the design, and therefore the cost of those modifications must be included. The cost of refurbishing the last failure during uncertainty reduction testing can be easily incorporated into the model by removing the $-x_{ui}$ term. In addition, though this formulation lumps together all test costs under a single budget constraint, additional constraints could be added to impose budget constraints on individual replaceable items, or groups of items, if needed.

The two sets of schedule constraints ($C2$ and $C3$) place limits on the total calendar time available for testing in two ways. The first ($C2$) deals with the total time spent on reliability growth testing, while the second ($C3$) deals with the full test schedule. Both sets of constraints are applied to each individual item, and refer to the total calendar time associated with testing, including time required for unit procurement, modifications, and refurbishments along with test time itself. This is in contrast with test time alone, which refers to the total number of operating hours accumulated by the test units. The schedule constraints drive the number of units required for testing. Each additional test unit is a multiplier on the amount of test time accumulated in a given amount of calendar time. Put another way, the calendar time required to achieve a desired amount of test time is equal to that test time plus the time required for modifications and refurbishments divided by the number of test units. The objective function depends only on the amount of reliability growth and uncertainty reduction testing that can occur, but these values are limited by the amount of test time that can be accumulated within the schedule constraints. When more test units are purchased, more test time can be accumulated.

The reliability growth test schedule constraint ($C2$) indicates that the total calendar time spent on reliability growth for each item must be less than the maximum reliability growth time τ_g. This can be thought of as a constraint on the time until the design must be fixed, and after which no further modifications are allowed. The total reliability growth calendar time is the sum of the unit procurement delay time τ_d and the actual calendar time required for testing, which is the sum of actual test time, time spent on modifications, and time spent on refurbishments, all divided by the number of test units. Note that while the test time and modification/refurbishment time is divided by the number of test units, procurement delay time is not.

The total test schedule constraint ($C3$) indicates that the total calendar time spent on testing for each item must be less than the maximum test time τ. This can be thought of as a constraint on the time until the mission spares allocation must be fixed – that is, before the reliability and uncertainty estimates for all items are as mature as they will get before the spares allocation for the mission is determined. The total test calendar time includes procurement delay, reliability growth testing, and uncertainty reduction testing, with the latter two divided by the number of test units. This model assumes that reliability growth and uncertainty reduction testing occur in sequence, in that order. While the reliability growth test schedule constraint considers only one type of testing, this constraint includes a tradeoff between reliability growth and uncertainty reduction testing. If more of the test schedule is used for reliability growth testing, less will be available for uncertainty reduction testing, and vice versa. Note that uncertainty reduction test time is a function of the failure rate after reliability growth testing. When more reliability growth testing occurs, the item's reliability will be higher, and as a result more test time will be required to achieve the same level of uncertainty reduction. This constraint illustrates the key tradeoff between reliability growth and uncertainty reduction in a schedule-constrained environment.

Constraints $C4$, $C5$, and $C6$ define the relationships between the decision variables and other parameters for each item, and are largely used to simplify mathematical notation by supplying parameters that are used in the previous three constraints. Specifically, $C4$ defines the failure rate after reliability growth λ_g, $C5$ defines the expected number of modifications during reliability growth testing ζ, and $C6$ defines the expected number of refurbishments during reliability growth testing r_g. These values are also useful for test planning and tracking purposes.

Constraints $C7$ through $C15$ define the relationships between the auxiliary test indicator variables and the decision variables. Some of these equations make use of "big-M" notation, a common approach used in integer programming in which a very large constant, denoted M, is used to enforce relationships between binary variables and integer variables. Specifically, M a sufficiently large positive number that, when multiplied by a binary variable, it can be used to either restrict another variable to 0 or produce a value so large that the constraint effectively has no effect.[13] These constraints, respectively, ensure that the following are true for each replaceable item:

- $C7$: If reliability growth testing is performed (i.e. $g > 0$), the reliability growth test indicator variable x_g is 1. In addition, if the reliability growth test indicator variable is 0, no reliability growth testing is performed.
- $C8$: If the reliability growth test indicator variable is 1, at least 1 time step of reliability growth testing is performed. In addition, if no reliability growth testing is performed, the reliability growth test indicator variable is 0.
- $C9$: If uncertainty reduction testing is performed (i.e. $r_u > 0$), the uncertainty reduction test indicator variable x_u is 1. In addition, if the uncertainty reduction test indicator variable is 0, no uncertainty reduction testing is performed.
- $C10$: If the uncertainty reduction test indicator variable is 1, uncertainty reduction testing occurs resulting in at least 1 planned observed failure. In addition, if no uncertainty reduction testing is performed, the uncertainty reduction test indicator variable is 0.
- $C11$: If the reliability growth test indicator variable is 1, the test indicator variable x is also 1. In addition, if the test indicator variable is 0, the reliability growth test indicator variable is also 0.
- $C12$: If the uncertainty reduction test indicator variable is 1, the test indicator variable x is also 1. In addition, if the test indicator variable is 0, the uncertainty reduction test indicator variable is also 0.
- $C13$: If the test indicator variable is 1, at least one of the reliability growth or uncertainty reduction test indicator variables must also be 1. In addition, if the reliability growth and uncertainty reduction test indicator variables are both 0, the test indicator variable is 0.
- $C14$: If units are purchased for testing (i.e. $k > 0$), then the test indicator variable is 1. In addition, if the test indicator variable is 0, no units are purchased for testing.
- $C15$: If the test indicator variable is 1, at least one unit is purchased for testing. In addition, if no units are purchased for testing, the test indicator variable is 0.

Taken together, these constraints ensure that the values of x, x_g, and x_u are consistent with the values of k, g, and r_u, thus ensuring that their impacts on the cost and schedule constraints described above are appropriately accounted for.

Finally, constraints $C16$ and $C17$ define the domains for the auxiliary and decision variables. The auxiliary variables x, x_g, and x_u are all binary, and the decision variables k, g, and r_u are all nonnegative integers.

4. Solution Approach

The test planning optimization problem is solved using a branch-and-bound approach, similar to that used in spares optimization in other work.[3,13,14] The problem is monotonic with regard to the amount of reliability growth test time and uncertainty reduction test time for both the objective function and the constraints. That is, increasing the amount of reliability growth testing can only reduce spares mass or leave it the same, and can only increase cost and test schedule. The same is true for uncertainty reduction testing. One coupling that is worth investigating in detail is the fact that the uncertainty reduction test time depends on the amount of reliability growth testing; however, increased reliability growth testing results in a more reliable items, which in turn increases the amount of uncertainty reduction test time associated with each unit increase in r_u. As a result of this monotonicity, the solution resulting in the lowest spares mass will have some combination that balances maximizing reliability growth test time and uncertainty reduction test time. Therefore, a valid lower bound on spares mass for a given subproblem can be obtained by evaluating the spares mass approximation ψ at the upper bound of the feasible domain of g and r_u. In general, this will be a loose bound, and as a result, this optimization is much less efficient than the one used for spares allocation. Ongoing research, described in Section III, is looking into improvements in this optimization approach.

The overall test plan optimization procedure is summarized in Figure 1, which shows how the standard branch-and-bound approach is applied to this particular problem. As with spares optimization, each subproblem is defined using upper and lower bounds for all decision variables and auxiliary variables, and is characterized by a lower bound on spares mass, calculated as described above. An incumbent solution is initialized to one that performs no testing, and has a spares mass equal to the mass that would be required given current failure rate estimates. A queue is initialized with a subproblem with all lower bounds set to 0, and all upper bounds set to 1 for all binary variables and infinity for all others, then tightened using bounds propagation. This queue is maintained as an ordered list that returns the subproblem with the lowest lower bound on spares mass first; put another way, the optimizer uses a best-first search strategy. The iterative search process removes the first subproblem in the queue (the candidate) and examines it. If the lower bound on spares mass for the candidate (i.e. the spares mass associated with the upper bounds of g and r_u) is higher than the incumbent solution, then the incumbent solution is the optimal solution and the process stops. Otherwise, the search continues. If the candidate is a complete assignment – that is, if the domain of every variable has been reduced to a single value – the spares mass associated with that assignment is evaluated. If the result is lower than the spares mass of the incumbent solution, a better solution has been found and the incumbent is replaced with the candidate solution. Otherwise, the candidate is discarded and the process continues to the next subproblem in the

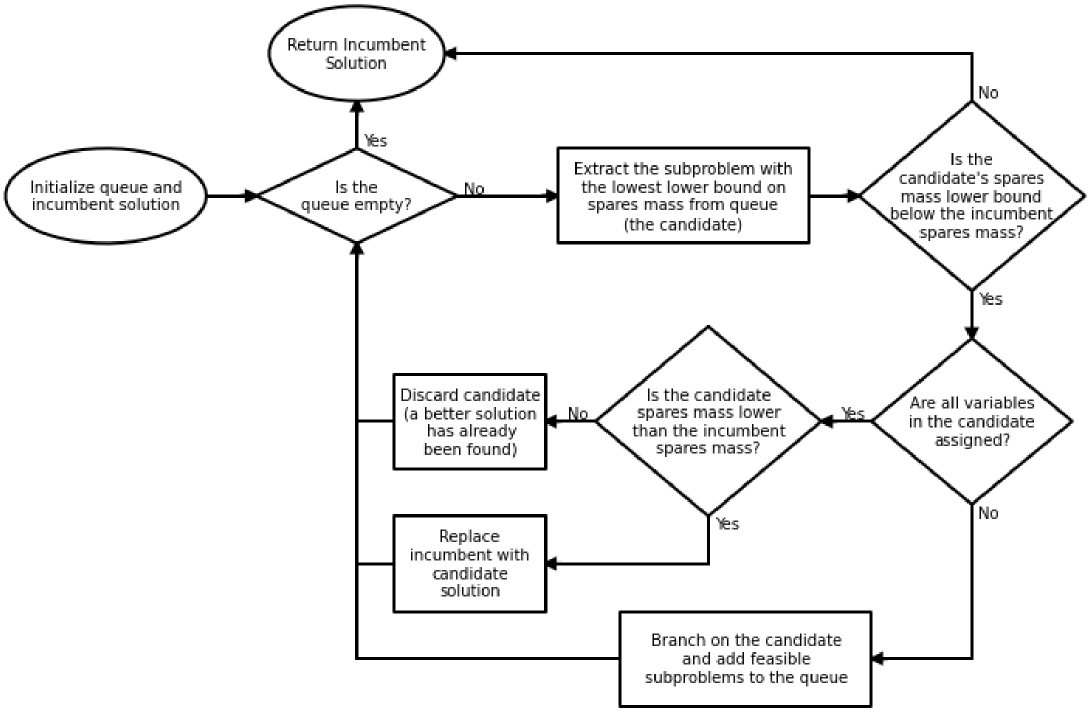

Figure 1: Test plan optimization flow chart, showing a branch-and-bound optimization using best-first search.

queue. If the candidate is a partial assignment, meaning there are unassigned variables in the problem, the search branches on the candidate by selecting a variable, splitting that variable's domain in half, and performing bounds propagation to generate two "child" subproblems. Bounds propagation iteratively checks the upper and lower bounds on each unassigned variable and tightens them based on the problem constraints, using the appropriate upper and lower bounds on all other variables as necessary. This is effectively a simplified version of domain arc consistency that deals with the edges of each variable domain instead of examining every value within the domain, and can be applied here due to the monotonicity of the objective function and constraints. Spares mass bounds are calculated based on the spares mass associated with the upper bound on g and r_u for all variables after bounds propagation. The resulting subproblems are then added to the queue if they are feasible, and the process continues by removing and examining the next subproblem. Once the queue is empty, the incumbent solution is the optimal solution.

III. Discussion and Future Work

The methodology presented above provides a way to evaluate and optimize test campaigns in order to inform test planning and technology investment. This approach improves upon current practices in two ways. First, the above methodology applies formal optimization techniques to balance test budget and schedule resources across all of the items being considered, as well as across reliability growth and uncertainty reduction testing for each item, rather than looking at individual items, systems, or tests in isolation. A broader, integrated view will enable more informed decisions and likely produce better outcomes than a narrow view.[15] Second, the application of formal mathematical optimization techniques enables a systematic exploration of the enormous supportability test tradespace. While a manual trade space exploration could examine dozens or (given sufficient resources) hundreds of options, this problem formulation enables the exploration of tens of thousands of options in order to identify an optimal solution, given the problem definition. The process is also largely automated, and in contrast to a manual trade study can therefore be rerun with relatively little overhead beyond computer time to examine sensitivities and "what if" scenarios. This capability to examine a wider range of options will be further expanded when more efficient optimization strategies are applied, as discussed below.

Results from the optimizer should not be thought of as a final answer; instead, the optimizer itself is an analytical tool that can be used to explore various scenarios, such as changes in budgets, schedules, systems, missions, or combinations thereof. Reliability testing should be integrated into the overall program test plan alongside performance, quality assurance, and acceptance testing. In addition, mass is only one metric that should be considered in informed decision-making. The performance, cost, risk, and schedule implications of test decisions and technology selection must also be considered in a holistic manner. In the end, the model and optimizer are simply useful mathematical abstractions used to project the potential costs and benefits of investments in order to inform decision-making. They make simplifying assumptions and evaluate expected behavior, and are not intended to be a precise “crystal ball” for seeing the future.

In addition, model outputs are also only as good as model inputs. Therefore, it is critical that prior failure rate estimates, estimates for procurement, modification, and refurbishment times, and unit cost estimates, along with all other parameters, be reasonable and realistic representations of the system being examined. Many of these parameters will not be known precisely prior to testing, but can and should be estimated based on past experience or analogy to other programs. Sensitivity analysis can also be used to identify key drivers and reasonable bounds on what schedules or costs would need to be to achieve desired outcomes. Data from past experience, as well as data collected during ongoing tests can be used for verification and validation of the model parameters. These data can also provide useful verification and validation of the various models used by the optimizer, particularly the reliability growth and uncertainty reduction models, by comparing projected test impacts to those that are actually achieved.

The current optimization implementation uses variable upper bounds to calculate a spares mass bound for subproblem pruning during the branch-and-bound search. While this approach enables very fast bounds calculations, it provides a loose bound that is not very effective at pruning the search space. As a result, full optimization is relatively slow when budget and schedule are high and the search space is large. Approximate solutions found via the greedy heuristic (i.e. incrementally selecting the most valuable reliability growth or uncertainty reduction step to take) can approximate a Pareto frontier, but these solutions are only approximate. Future work will seek a more effective bounds approximation strategy, or more efficient search strategy, in order to speed up the analysis process.

IV. Conclusions

Reliability and supportability testing will be critical activities for future missions due to the new and more challenging logistical challenges of crewed deep-space operations. However, this kind of long-duration can require careful schedule and budget planning. The AES LSS Ground Test Planning activity is applying a range of test modeling and optimization techniques, described in this paper, to inform test plans. The test plan optimization capability described here enables a systematic and holistic balancing of supportability test resources across all items within the system, as well as across both reliability growth and uncertainty reduction testing, in order to make the most effective use of limited budget and schedules to reduce spares mass requirements. The results from these models will help inform overall ground test planning, which, in conjunction with on-orbit testing on the ISS, will provide the data required to validate and refine failure rate estimates for future exploration missions.

Acknowledgements

This work was supported by the AES LSS Ground Test Planning activity.

References

[1] Owens, A., De Weck, O., Stromgren, C., Goodliff, K. E., and Cirillo, W. Supportability Challenges, Metrics, and Key Decisions for Future Human Spaceflight. 2017.

[2] Cirillo, W., Aaseng, G., Goodliff, K., Stromgren, C., and Maxwell, A. Supportability for Beyond Low Earth Orbit Missions. Presented at the AIAA SPACE 2011 Conference & Exposition, Long Beach, CA, 2011.

[3] Owens, A. C. *Multiobjective Optimization of Crewed Spacecraft Supportability Strategies*. Doctoral Thesis. Massachusetts Institute of Technology, Cambridge, MA, 2019.

[4] Stromgren, C., Goodliff, K. E., Cirillo, W., and Owens, A. The Threat of Uncertainty - Why Using Traditional Approaches for Evaluating Spacecraft Reliability Are Insufficient for Future Human Mars Missions. Presented at the AIAA SPACE 2016, Long Beach, CA, 2016.

[5] Owens, A. C., de Weck, O. L., Stromgren, C., Goodliff, K., and Cirillo, W. Accounting for Epistemic Uncertainty in Mission Supportability Assessment: A Necessary Step in Understanding Risk and Logistics Requirements. Presented at the 47th International Conference on Environmental Systems, Charleston, SC, 2017.

[6] Owens, A., and de Weck, O. International Space Station Operational Experience and Its Impacts on Future Mission Supportability. Presented at the International Conference on Environmental Systems, Albuquerque, NM, 2018.

[7] Owens, A., and de Weck, O. How Much Testing Is Needed to Manage Supportability Risks for Beyond-LEO Missions? Presented at the International Conference on Environmental Systems, Boston, 2019.

[8] Owens, A., Piontek, N., Cirillo, W. M., Stromgren, C., and Cho, J. More Data Needed for Failure Rate Estimation, Validation, and Uncertainty Reduction. Presented at the International Conference on Environmental Systems, 2021.

[9] Crow, L. H. Planning a Reliability Growth Program Utilizing Historical Data. Presented at the Reliability and Maintainability Symposium, Lake Buena Vista, FL, 2011.

[10] Duane, J. T. “Learning Curve Approach to Reliability Monitoring.” *IEEE Transactions on Aerospace*, Vol. 2, No. 2, 1964, pp. 563–566. doi:10.1109/TA.1964.4319640.

[11] Dezfuli, H., Kelly, D., Smith, C., Vedros, K., and Galyean, W. *Bayesian Inference for NASA Probabilistic Risk and Reliability Analysis*. Publication NASA/SP-2009-569. National Aeronautics and Space Administration, Washington, DC, 2009.

[12] Barton, P. I. *Mixed-Integer and Nonconvex Optimization*. Massachusetts Institute of Technology, Cambridge, MA, 2011.

[13] Bradley, S. P., Hax, A. C., and Magnanti, T. L. *Applied Mathematical Programming*. Addison-Wesley Pub. Co, Reading, Mass, 1977.

[14] Hillier, F. S., and Lieberman, G. J. *Introduction to Operations Research*. McGraw-Hill, Boston, 2001.

[15] Owens, A., Jones, C. A., Cirillo, W., Klovstad, J., Judd, E., Chai, P., Merrill, R. G., Piontek, N., Stromgren, C., and Cho, J. Integrated Trajectory, Habitat, and Logistics Analysis and Trade Study for Human Mars Missions. Presented at the ASCEND 2020, Virtual Event, 2020.