A RESILIENCE-ORIENTED EXTRA-TERRESTRIAL HABITAT DESIGN PROCESS

by

Jacqueline Ulmer

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science in Aeronautics and Astronautics

School of Aeronautics and Astronautics

West Lafayette, Indiana

August 2023

THE PURDUE UNIVERSITY GRADUATE SCHOOL

STATEMENT OF COMMITTEE APPROVAL

Dr. Karen Marais, Chair

School of Aeronautics and Astronautics

Dr. Shirley Dyke

School of Mechanical Engineering

Dr. Leifur Leifsson

School of Aeronautics and Astronautics

Approved by:

Dr. Gregory A. Blaisdell

Dedicated to my mom, Michelle, for taking me to mother-daughter Space Camp when I was nine and making sure I knew I could be anything I wanted to be.

ACKNOWLEDGMENTS

This material is based upon work supported in part by NASA under grant or cooperative agreement award number 80NSSC19K1076. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Aeronautics and Space Administration (NASA).

I would like to thank the RETHi team, past and present, without whom none of this research would be possible, starting with the MCVT modelers who spent hours and hours working on the platform so that I could use it. The majority of this research focuses on the MCVT's power system, which is currently being modeled by Leila Chebbo from the University of Connecticut. Leila was instrumental in perfecting the power system and helping me understand it. Seungho Rhee on Interior Environment and ECLSS and Murali Krishnan on the Agent model also contributed greatly to my ability to carry out this research. Mohsen Azimi was an amazing help with every aspect of this project, from coordinating modelers and tracking MCVT requirements to helping me debug my own scripts. Without him, the MCVT would not be functional and this research would not have been possible.

I would also like to thank the VRSS group for their ceaseless willingness to share their time, experience, and opinions to help me better myself as a researcher. Dr. Karen Marais deserves so many thanks for assembling such an impressive group of students and inspiring so much collaboration. I would like to thank Dr. Marais for taking me on and guiding me through this process from start to finish with so much careful attention.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY

Accident State: a system condition that has the potential to cause serious habitat damage or loss of life

CDCM: one of RETHi's two simulation platforms. The CDCM uses primarily conceptual models to output lower fidelity results than the MCVT on the order of years or even decades.

Control Effectiveness: a measurement of how well a safety control addresses its target hazard, developed to help designers select which safety controls and which implementation strategies might best serve their habitat

Disruption: an event that causes a drop in a habitat system's performance, moving it to a hazardous state

Disruption Scenario: One series of simulated events that could stem from an initiating disruption. For the disruption *Meteorite Impact*, possible disruption scenarios include *Meteorite Impact above ECLSS at Intensity Level 5* and *Meteorite Impact above Power Subsystem at Intensity Level 3.*

ECLSS: Environmental Control and Life Support System

FHA: functional hazard analysis. A process of identifying the functions necessary for a system to function, and what could happen if those functions are not present

Hazardous State: a system condition of reduced performance. If a hazardous state is left unattended, it can turn into an accident state

MCVT: one of RETHi's two simulation platforms. The MCVT uses physics-based models to perform high-fidelity simulations to output high-fidelity output for simulations on the order of minutes.

RETHi: the Resilient Extra-Terrestrial Habitat Institute. A NASA Space Technology Research Institute (STRi)

Resilience: the ability of a system to react to, survive, and recover from expected and unexpected hazardous states

Safety: a habitat characteristic of protecting human life

Safety Control: a process that can prevent or return a habitat from a hazardous or accident state.

STA: state-trigger analysis. A process of organizing the disruptions, hazardous and accident states, and safety controls that could affect a system

Trigger: an event that shifts a system from one state to another

ABSTRACT

In the wake of the first Artemis launch, humanity is more focused on space exploration and travel than it has been in the half a century since the Space Race. This time, it's not enough just to touch down on the Moon; we want to build sustainable homes on the Moon and on Mars. The goal of long-term extra-terrestrial habitation begs the question: how do we design habitats that can protect human life so far from Earth?

The Resilient Extra-Terrestrial Habitat Institute (RETHi) has been operating for four years now building a foundation of ideologies and tools to help answer that question. The institute has developed a control-theoretic approach to habitat resilience based on a state-trigger analysis, a database of potential hazards to a habitat, metrics for resilience quantification, and simulation platforms for design verification.

The combination of these developments allows for the proposition of a resilience-oriented habitat design process. The process takes the shape of a typical systems vee and is tailored to the needs of an extra-terrestrial habitat and the tools available through RETHi. The process proposes a way to build resilience into the requirements development and design verification of extra-terrestrial habitats at three system levels. The result of this study is a discussion on how we design, evaluate, and select safety mechanisms for extra-terrestrial habitats.

Safety mechanisms are selected by simulating the habitat's response to a disruption when equipped with one safety mechanism at a time and quantifying the habitat's resilience. Then, the resilience of the habitats with different mechanisms are compared, illuminating the best option. Simulations for each mechanism are performed under a variety of circumstances, changing the time of day and intensity of the disruption as well as the type of repair agent carrying out the mechanism to capture the habitat's behavior as totally as possible.

This analysis shows how different safety mechanisms performances compare and provides a basis for making design decisions.

1. INTRODUCTION

In November of 2022, over 50 years after NASA landed men on the Moon at the height of the Space Race, the launch of Artemis I signified a return to international prioritization of space exploration. This new Space Race may have more to do with economic and defensive pursuits than political ones, but the national prestige that will be awarded to the country to win the race by putting the first long-term habitat on the moon, or the first human beings on Mars, is just as strong an incentive as it was half a century ago (Howell, n.d.).

This return to the final frontier prompts a whole slew of discussions, including the question of what happens *after* boots touch down. Housing human beings on the Moon and Mars will be no small feat, which naturally inspires today's aerospace engineers to consider the challenge of extra-terrestrial habitats.

How would we even go about starting such a project? Designing a habitat to exist on another planetary body is not a simple engineering problem. The biggest obstacles standing in the way are the hostile environment and the inaccessibility to Earth. A habitat on the moon or on Mars will have to keep human beings safe from the dangers that come with being on those bodies, and it will have to do so without real-time help from experts on Earth or most of the amenities that are taken for granted here.

To make matters more complex, we don't have anything close to a comprehensive list of what could go wrong in an extra-terrestrial environment, nor do we have an inexpensive, safe way to test out potential designs. Our understanding of the risks we're facing is miniscule compared to what we know about Earth, and solving problems through trial and error is not an option. On Earth, we understand exactly how a building's environment will affect it, what things could go wrong, and the best ways to prevent losses. We can list all the potential problems a building might face, make sure we're designing for them, and call it safe. Because we can't list all the potential faults that could affect a habitat on another planetary body, we can't simply apply the design techniques that work best on Earth to this new challenge.

The bottom line is that the typical approach to constructing safe buildings on Earth, where every potential fault is listed and mitigated, will not be enough on its own for extra-terrestrial habitats. We will have to accept that it is impossible to conceive of everything that could go wrong. So, what *can* we do? Is there a way to take our limited understanding of other planets' environments and develop safe, functional habitats? Can we build habitats that are capable of protecting their inhabitants against unforeseen faults in addition to those we anticipate?

The best approach to designing extra-terrestrial habitats is not to list everything that we think could go wrong and design safety mechanisms to combat each one, so what is it? Is it possible to develop safety mechanisms that protect against more than one potential disaster? Are there certain features a habitat can have that make it impervious to some faults by nature? How do we decide which safety mechanisms a habitat needs, and which would be better left out?

These are the kinds of questions we need to explore in these very beginning stages of extra-terrestrial habitat design.

1.1 Resilient Extra-Terrestrial Habitat Institute (RETHi)

The Resilient Extra-Terrestrial Habitat Institute (RETHi) was created to do research on how to design smart habitats that will react to, survive, and recover from both expected and unexpected disruptions in a deep space environment (Dyke, 2019). Students at four universities across the US are working in three thrusts to achieve this mission.

The research in this thesis falls under RETHi's resilience thrust. The resilience thrust's goals are to define resilience as it applies to extra-terrestrial habitats, learn how to build resilience into extra-terrestrial habitats, and evaluate the resilience of different habitats using RETHi's simulation platforms.

The other two thrusts are robotics and awareness. The goal of the robotics thrust is to develop autonomous robots that can operate independently and collaborate with humans, while the awareness thrust is researching how to build networks of sensors that can actively learn, detect, and diagnose issues (Dyke, 2019).

Together, the thrusts aim to carry out the institute's mission.

1.2 Previous Work

The first obstacle for the resilience thrust is defining resilience in terms of extra-terrestrial habitats. We define resilience by creating a theoretical model that we can use to think about what resilience, safety, and risk mean for this area of research and how to design habitats with these things in mind.

1.2.1 State-and-Trigger Model

We use a state-trigger model to visualize how individual disruptions, hazardous states, and safety controls impact the operation of the habitat, as shown in Figure 1. The operation starts in a nominal state, where each system element is functioning as desired. Then, a disruption occurs. This disruption is a "trigger" that causes a drop in the system's performance, creating a hazardous state. If no action is taken, the system will eventually shift from a hazardous state into an accident state. An accident state is a type of hazardous state defined by loss, whether that be loss of mission objectives, equipment, or human health. To prevent the accident state, we introduce safety controls. A safety control can either prevent the system from moving into a hazardous or accident state, or return the system to a safer state. It is also possible that a safety control will stop the system from worsening without returning it all the way to a nominal state. Such a safety control results in the system moving into a temporary safe state, where further safety controls can be activated to solve the problem.

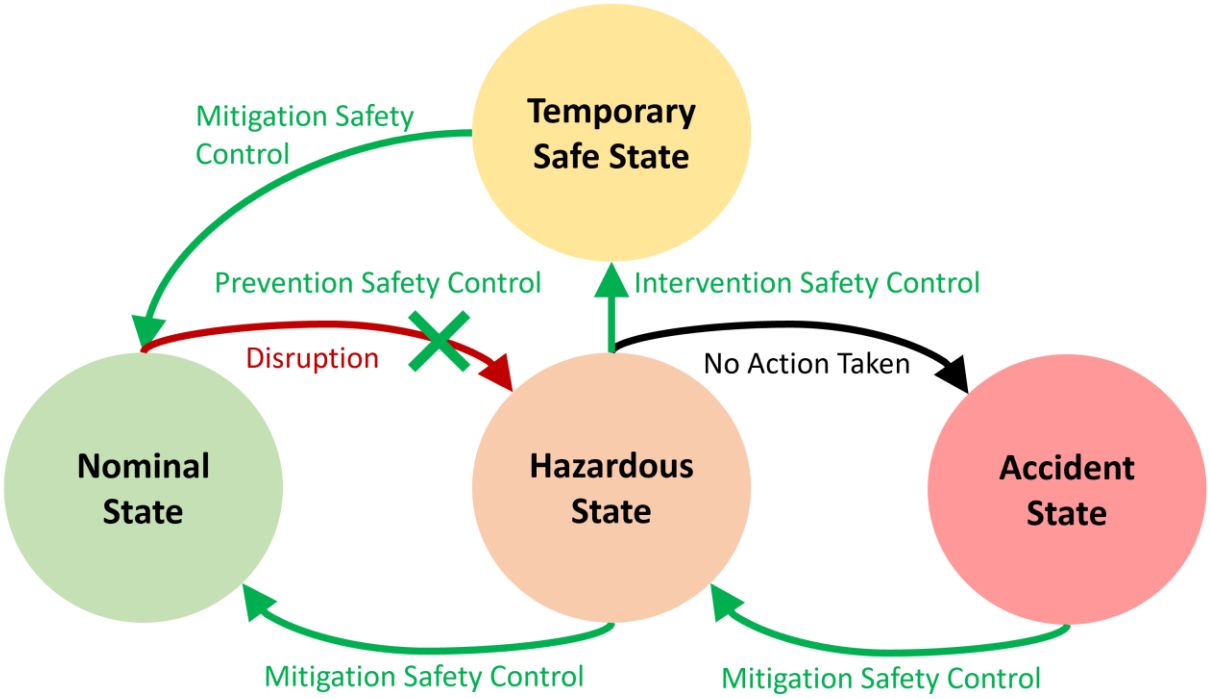


Figure 1: The State-Trigger Model

1.2.2 Control-Theoretic Approach

The performance of each habitat operation falls into one of the above states in the state-trigger model. The combination of all the habitat's individual operations forms an overall habitat-wide state. We visualize the whole-habitat performance using a control-theoretic approach based on the Systems-Theoretic Accident Model, or STAMP (Leveson, 2004). Figure 2 gives a visual representation of the control-theoretic approach. The green region represents a region of safe behavior of a habitat. As conditions change and disruptions happen in subsystems or components, the performance of the habitat might shift toward the regions of unsafe behavior, shown in red. The safety controls on board the habitat can shift the performance back to the center of the safe region.

RETHi takes a control-theoretic approach to modeling the resilience of entire habitats because we need to be able to characterize foreseen disruptions as well as unforeseen disruptions (Kitching, 2020). The control-theoretic approach allows us to discuss the principles of making a habitat resilient without identifying specific disruptions, hazardous states, accident states, or safety controls.

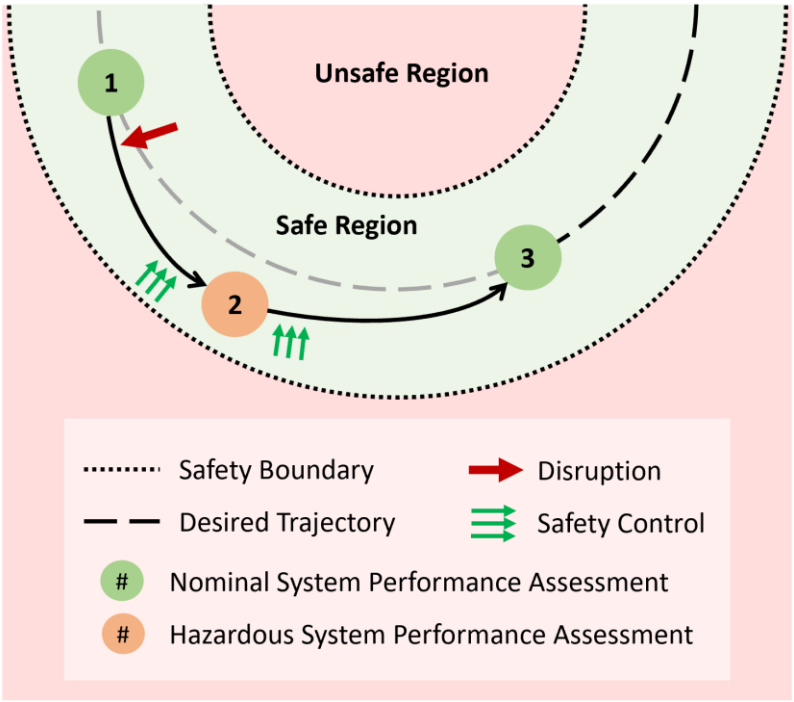


Figure 2: The Control-Theoretic Approach (Dyke et al, 2018)

Using this approach to modeling habitat performance presents 5 steps:

1. Identify Hazards

The first step in characterizing the resilience of a habitat is to identify events that could threaten habitat performance. A disruption is any event that might cause the habitat performance to shift from the safe region toward or into the unsafe region, putting it in a hazardous state. Step 1 is brainstorming disruptions and their resulting hazardous states, with the idea being that any unforeseen disruptions that aren't listed during this step will at least result in one or more of the hazardous states that *are* considered.

2. Assess Hazards

The next step is to analyze the disruptions and hazardous states that were brainstormed. The goal of this step is to evaluate each disruption and hazardous state to assess its criticality. Disruptions that cause many hazardous states are more critical to design for than disruptions that only cause one hazardous state. Hazardous states that have severe or fast-acting consequences are more

critical than hazardous states that have little effect on habitat performance. The highest criticality disruptions and hazardous states will be driving factors for habitat design.

3. Identify Safety controls

The next step is to perform a similar brainstorming session for safety controls. Safety controls are actions that mitigate hazardous states and shift behavior back toward or into the safe region. Safety controls can be tasks performed by crew or by robots, safety protocols, automated habitat features, or anything else that results in safer habitat performance.

4. Assess Safety controls

Like with disruptions and hazardous states, we then assess each safety control to quantify its importance. A safety control that mitigates many hazardous states is more important than one that only mitigates one hazardous state, and a safety control that mitigates more reliably or faster is more important than a slower or less reliable alternative. To complete this step, the Control Effectiveness metric was designed. Control Effectiveness is a metric assigned to each safety control. Its development will be discussed in section 1.2.4.

5. Assess Residual Risk

The final step is to evaluate the performance of a habitat with a selected set of disruptions, hazardous states, and safety controls. The habitat designer considers the design's performance and determines whether the habitat is resilient enough. If it is not resilient enough, the designer can choose to evaluate a different selection of safety controls. If the habitat is resilient enough, the designer then considers whether enough disruptions and hazardous states were considered for the results to be meaningful.

1.2.3 Disruption Database

To accomplish the tasks of identifying hazards and safety controls, RETHi has built a database. The database is the list of potential hazards and safety controls that we can consider and use as examples as we work out the best way to design habitats. The current database contains 19 triggers,

217 hazardous states, and 786 safety controls that come from previous failures in the space field, accidents in similar fields, and brainstorming.

Some triggers can lead to multiple hazardous states, and some hazardous states can be caused by multiple triggers. The interconnectivity of system states is visualized in the figure below. Red points represent hazardous states, and the lines between them represent triggers moving the system from one state to another.

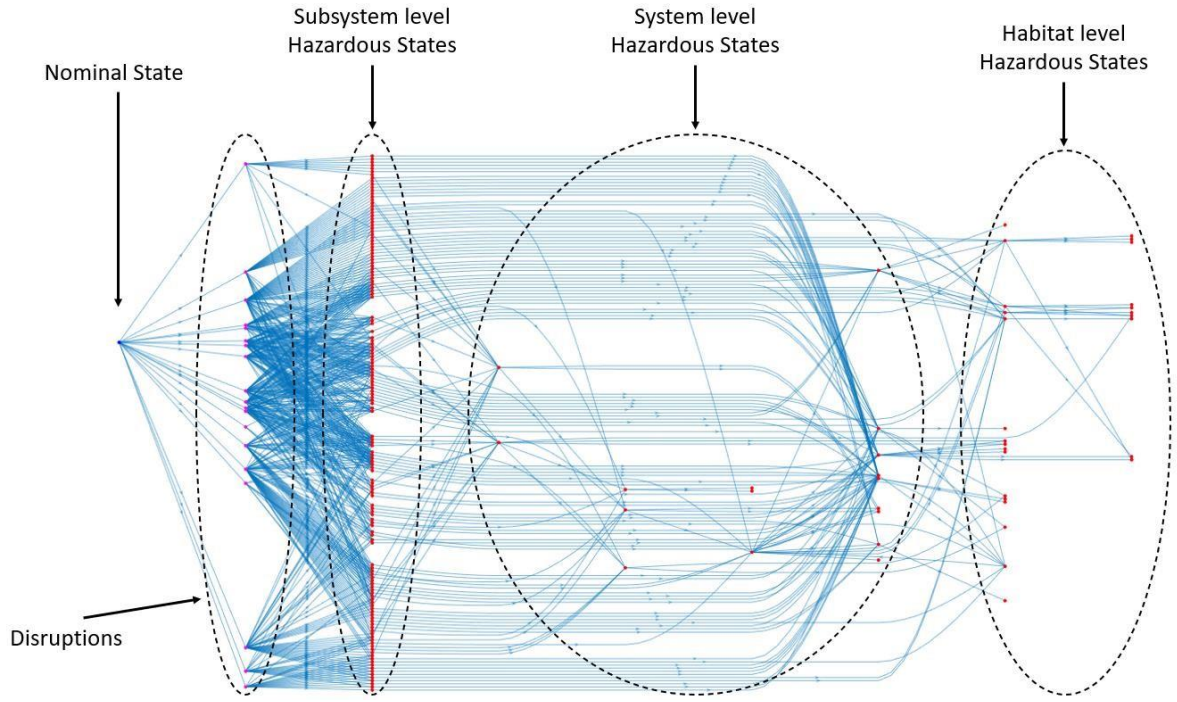


Figure 3: Database Visualization (Kitching, 2020)

The safety controls in the database are actions that could be taken to mitigate a specific hazardous state. For example, one safety control is “Clean Dust off of Solar PV Arrays.” There are multiple methods of cleaning solar PV arrays. Each method is called an implementation strategy. The possible implementation strategies for “Clean Dust off of Solar PV Arrays” could include a person cleaning them with a cloth, a robot cleaning them with a cloth, or an automated function built into the arrays. The specific implementation strategies for each safety control are not listed in the database; they come into play later in the design process.

The safety control option space of a habitat design process refers to the list of all the safety controls being considered for the design and all the potential implementation strategies that could fill each safety control's role.

1.2.4 Control Effectiveness

Once we understand the control-theoretic approach to the challenge of designing resilient extra-terrestrial habitats, and have a list of potential disruptions, hazardous states, and safety controls, we can start to think about how to make individual design choices that will set a habitat up for success. One factor we can use to make those design decisions is called Control Effectiveness.

Control Effectiveness is a measurement of how well a safety control addresses its target hazard, developed to help designers select which safety controls and implementation strategies might best serve their habitat.

Different implementation strategies for the same safety control should all result in mitigation of their target hazardous state, but could theoretically do so to different degrees. A person wiping a solar PV array with a cloth might result in a less clean array than if the cleaning were done by a robot, but it also might be faster.

Control Effectiveness consists of four values that each describe a safety control's effectiveness in a different way. Each value falls on a scale from zero to one with zero being worst, and one being best.

> ***P_available*** is the probability that a safety control is available at the time it is needed. The availability of a safety control can be affected by the number of single-use items it needs, the likelihood that a necessary tool might already be in use, etc. A P_available of 0 refers to a safety control that is not possible on the habitat.
>
> ***P_design*** is the probability that a safety control is competently designed. P_design is a measure of how likely the safety control is to alleviate the hazardous state, assuming it is perfectly implemented by an agent.

P_implementation is the probability of perfect implementation. P_implementaiton value considers the possibility that an agent might fail to use a safety control exactly as intended.

M_response is the response margin. It accounts for the ratio between the time it takes for a disruption to cause cascading effects (t_effect) and the time it takes for the safety control to mitigate the disruption (t_affect).

In the early stages of design, these values can be estimated by the design team based on intuition. As the design progresses and there are fewer decisions to be made, empirical data should be used to inform each value selected.

1.2.5 Resilience Metrics

The last major obstacle we face is evaluating the resilience of a habitat once a set of design decisions have been made, which is a necessary step in designing a habitat.

How can we quantify resilience and measure the impact of different design decisions? Much research has already been done on quantifying resilience. Different papers have developed different metrics that return values based on "performance metrics." A performance metric for a system is the time history of a variable that can represent the system's overall performance. One system can have multiple meaningful performance metrics.

For an extra-terrestrial habitat, performance metrics could be the power system's generation, the interior environment's temperature, the amount of dust on the solar panels, etc. For this research, we will calculate resilience metrics for a series of performance metrics for each habitat design.
To understand how resilience metrics produce values, consider the generic performance curve in Figure 4.

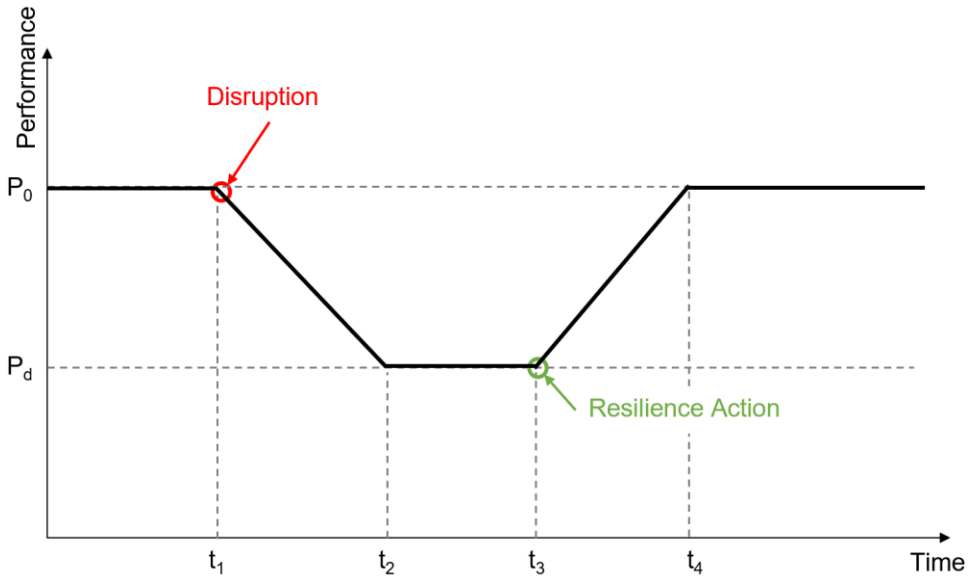


Figure 4: Standard Resilience Curve

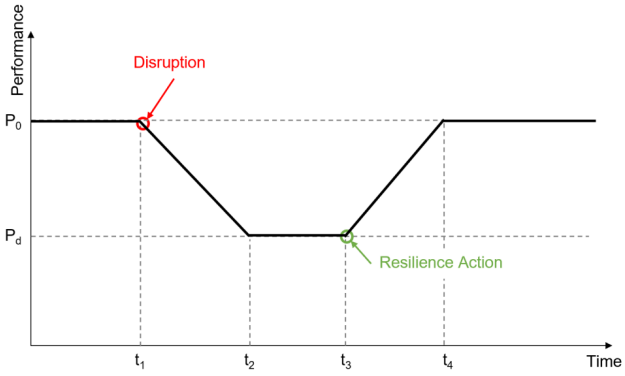
Figure 4 shows the reaction to a disruption of a generic measure of performance for a habitat. The performance maintains its starting value until there is a disruption. The time of the disruption is denoted by t_1. Performance begins to drop until t_2, when it stabilizes at a decreased level of performance. It remains at the decreased performance level until t_3, when a resilience action is implemented to start recovery. Finally, when performance returns to its initial value, it once again levels off at t_4 to reach a new stable operational state.

We want to investigate seven existing resilience metrics to determine which of them will be most appropriate for this research.

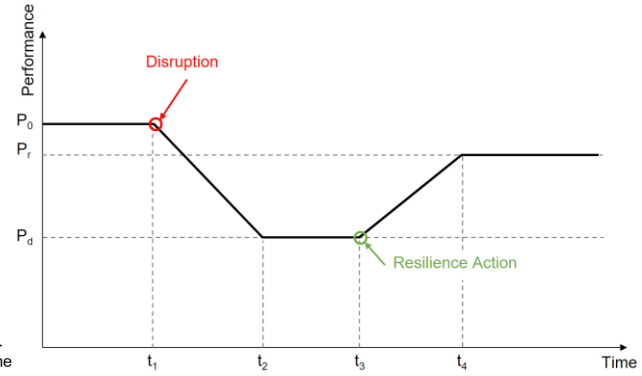
While the standard resilience curve is one way that a performance metric could behave after a disruption, there are also a variety of other ways the system could react. In this section, 10 different potential shapes of performance curves are considered.

Each of the seven metrics investigated in this thesis will be applied to the ten different performance curve shapes to determine whether the metric returns meaningful values for those curves or not. If a metric does not return meaningful values for one or more of the curve shapes considered, it may not be a good fit for this research, as it cannot help us quantify resilience in every case.

For this investigation, the following shapes of performance curves are considered:

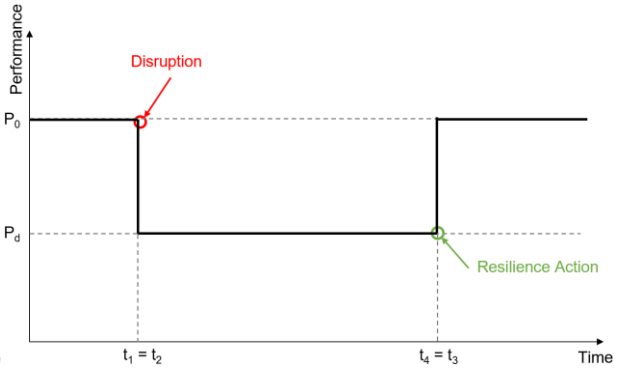


Baseline

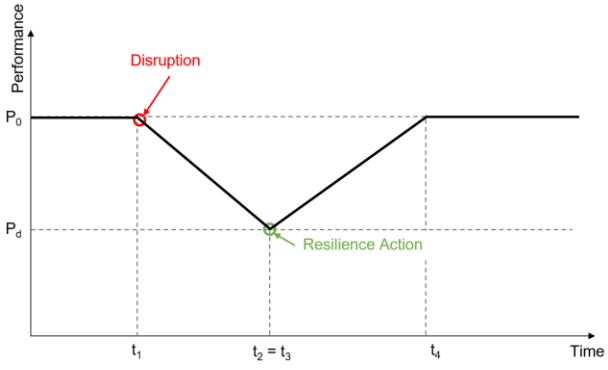


Baseline with Performance Loss

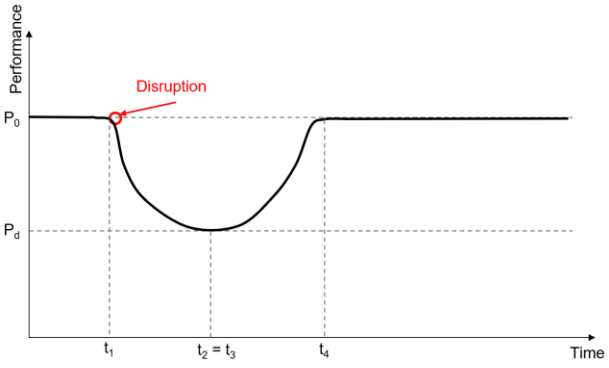
Baseline with Performance Gain



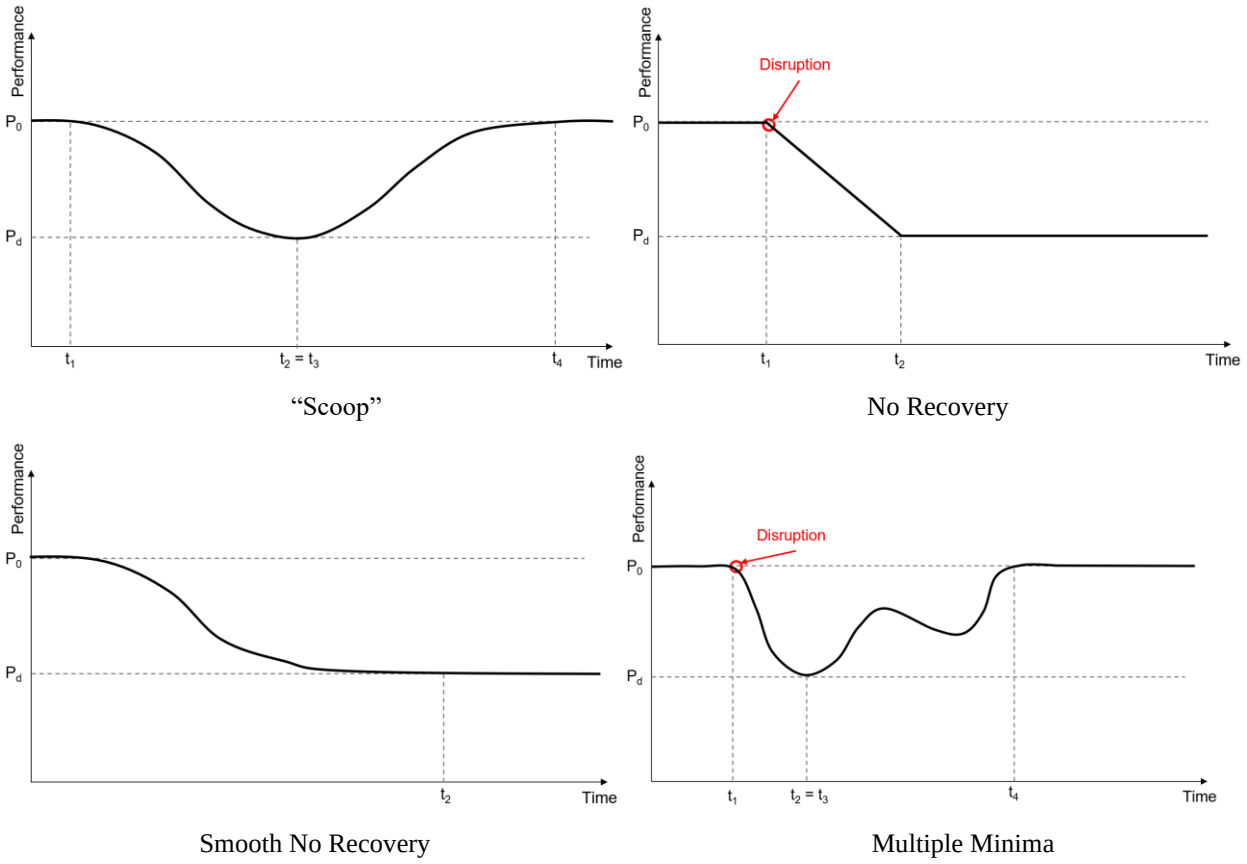
"Bucket"



"V"



"U"



Metric 1: Das et al.

The first metric considered is defined by Das et al. (2020) as the inverse of the time that a system is in a state of disruption. In this investigation, the time that the system is in a disrupted state is from t_1 to t_4.

$$R_1 = \frac{1}{t_4 - t_1} \tag{1}$$

Table 1: Das et al. Resilience Metric Usage

Shape	Value of R_1	Works?	Explanation
Baseline	$0 < R_1 < \infty$	Yes	
Baseline with Performance Lost/Gained	$0 < R_1 < \infty$	Yes	
Bucket	$0 < R_1 < \infty$	Yes	
V/U/Scoop	$0 < R_1 < \infty$	Yes	
No Recovery, Smooth No Recovery	0	Yes*	t_4 does not exist
Multiple Minima	$0 < R_1 < \infty$	Yes	

The Das et al. metric can be calculated easily for every shape that shows any recovery. When there is no recovery, there is no value for t_4. In this case, the denominator of the metric goes to ∞, making the value of R_1 go to zero. Zero is a meaningful value for this metric for this shape because 0 is the smallest possible value of this metric and a curve with no recovery is the least resilient curve possible. However, because there is no t_4, simply using the equation for the metric in the case of no recovery will result in an error. This is noted by the yellow boxes and asterisk in the No Recovery row.

Metric 2: Henry & Ramirez-Marquez 1

The second metric investigated is defined by Henry & Ramirez-Marquez (2012) as the ratio of the recovered performance to the minimum performance.

$$R_2 = \frac{P(t_4)}{min(P)} \tag{2}$$

Table 2: Henry & Ramirez-Marquez 1 Resilience Metric Usage

Shape	Value of R_2	Works?	Explanation
Baseline	$1 < R_2 < \infty$	Yes	
Baseline with Performance Lost/Gained	$1 < R_2 < \infty$	Yes	
Bucket	$1 < R_2 < \infty$	Yes	
V/U/Scoop	$1 < R_2 < \infty$	Yes	
No Recovery, Smooth No Recovery	1	Yes*	t_4 does not exist
Multiple Minima	$1 < R_2 < \infty$	Yes	

Like the Das et al. metric, this metric can be calculated plainly for any curve that shows recovery. In the case of no recovery, there is no t_4 at which the recovered performance can be evaluated. In this case, the value of the performance at the end of the simulation can be used for the performance at t_4. With no recovery, the final performance will be the same as the minimum performance, making the value of R_1 one. One is the smallest possible value of this metric and signifies that no recovery is the least resilient case, which is true. With this considered, this metric returns meaningful values for all curves considered.

Metric 3: Bruneau et al.

The next metric uses integration to calculate the total performance lost by the system between the time of the disruption and recovery. It is defined by Bruneau et al. (2003).

$$R_3 = \int_{t_1}^{t_4} (P(t_1) - P(t))dt \tag{3}$$

Table 3: Bruneau et al. Resilience Metric Usage

Shape	Value of R_3	Works?	Explanation
Baseline	$0 < R_3 < \infty$	Yes	
Baseline with Performance Lost/Gained	$0 < R_3 < \infty$	Yes	
Bucket	$0 < R_3 < \infty$	Yes	
V/U/Scoop	$0 < R_3 < \infty$	Yes	
No Recovery, Smooth No Recovery	∞	No	t_4 does not exist
Multiple Minima	$0 < R_3 < \infty$	Yes	

This metric works for all shapes that show recovery. When there is no recovery, the metric can not be calculated because there is no t_4. Unlike the previous two metrics, there is no way around this issue. Integrating from t_1 to the end of the simulation time could result in a smaller value for the metric in a no-recovery case than one where there is recovery, which would imply that the no-recovery case was more resilient. For this reason, the Bruneau et al. metric is not meaningful for the No Recovery case. This is signified by the red row.

Metric 4: Ayyub et al.

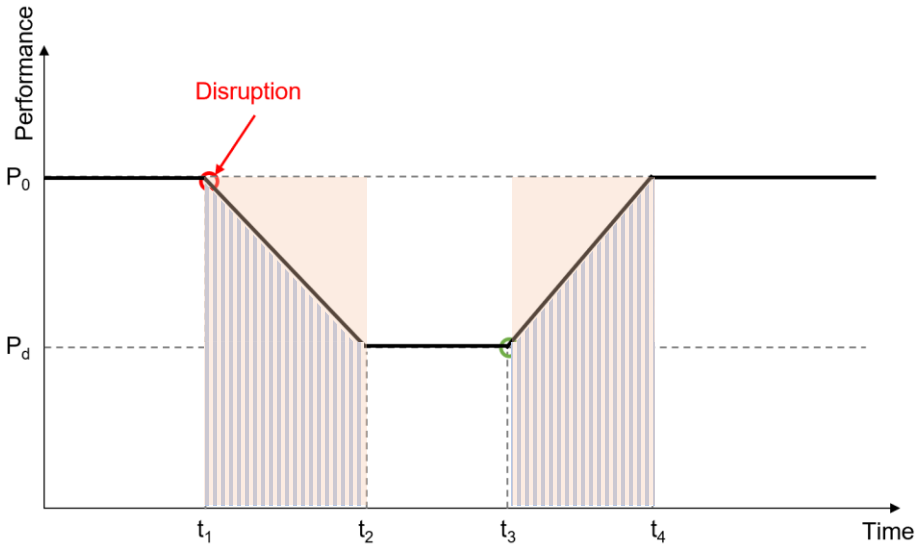
The next metric comes from Ayyub et al. (2014) and is defined as:

$$R_4 = \frac{t_1 + F(t_2 - t_1) + R(t_4 - t_3)}{t_4 - t_1} \tag{4}$$

where F and R are ratios of the actual performance of the system to the non-disrupted performance during the failure and recovery stages. In the figure below, F and R are the ratios of the striped sections to the orange sections.

$$F = \frac{\int_{t_1}^{t_2} P(t)dt}{\int_{t_1}^{t_2} P(t_1)dt} \tag{5}$$

$$R = \frac{\int_{t_3}^{t_4} P(t)dt}{\int_{t_3}^{t_4} P(t_1)dt} \tag{6}$$



This metric as a whole is larger when there is less performance loss and when there is less time spent in a state of decreased performance.

Table 4: Ayyub et al. Resilience Metric Usage

Shape	Value of R_4	Works?	Explanation
Baseline	$0 < R_4 < 1$	Yes	
Baseline with Performance Lost/Gained	$0 < R_4 < 1$	Yes	
Bucket	$0 < R_4 < \infty$	No	$t_1 = t_2$ and $t_3 = t_4$
V/U/Scoop	$0 < R_4 < 1$	Yes	
No Recovery, Smooth No Recovery	∞	No	t_4 does not exist
Multiple Minima	$0 < R_4 < 1$	Yes	

This metric does not return meaningful values for the bucket shape or when there is no recovery. In the bucket shape, the values of F and R would both be 0, reducing the metric to $\frac{t_1}{t_4 - t_1}$ which is not meaningful unless t_1 is the same for every simulation. Even when it is, this metric gives the same information as the Das et al. metric. When there is no recovery, R would go to infinity.

Metric 5: Henry & Ramirez-Marquez 2

The next metric investigated is also defined by Henry & Ramirez-Marquez. It improves on their previous metric. This metric is the ratio of the increase in performance during recovery to the loss in performance following the disruption.

$$R_5 = \frac{P(t_4) - min(P)}{P(t_1) - min(P)} \tag{7}$$

Table 5: Henry & Ramirez-Marquez 2 Resilience Metric Usage

Shape	Value of R_5	Works?	Explanation
Baseline	1	Yes	
Baseline with Performance Lost/Gained	$0 < R_5 < \infty$	Yes	
Bucket	$0 < R_5 < \infty$	Yes	
V/U/Scoop	$0 < R_5 < \infty$	Yes	
No Recovery, Smooth No Recovery	0	Yes*	t_4 does not exist
Multiple Minima	$0 < R_5 < \infty$	Yes	

This metric is able to return meaningful values for every shape when a slight modification is made for the case of no recovery to account for the lack of a t_4. If the performance at the end of the simulation is used for P(t_4), the metric returns a value of 0, which correctly expresses that no recovery is the least resilient case.

Metric 6: Yarveisy et al.

Metric 6 is designed by Yarveisy et al. (2020). It is comprised of a combination of three values that describe different aspects of the performance curve.

$$R_6 = Ab + (Ad \cdot Res) - (Ab \cdot Ad \cdot Res) \tag{8}$$

Ab denotes the absorptive capacity of the system, or its ability to limit performance loss after a disruption. The coefficient C_{Ab} accounts for natural degradation of the system. It is assumed that there is no natural degradation expected, so $C_{Ab} = 1$, leaving Ab defined as the ratio of the minimum performance to the starting performance.

$$Ab = C_{Ab} \cdot \left(\frac{min(P)}{P(t_1)}\right) \tag{9}$$

Ad is a measure of the adaptive capacity of the system, or its ability to stabilize performance after a disruption.

$$Ad = 1 - \frac{t_3 - t_2}{t_4 - t_1} \tag{10}$$

Res is the restorative capacity of the system. This is the ability of the system to return to its original performance level. C_R in the equation below is a coefficient that accounts for natural degradation of the system and is again assumed to be 1. C_T is the ratio of time spent not recovering to total time spent at a disrupted performance level.

$$Res = \frac{1}{90} \cdot tan^{-1}\left[\frac{P(t_4) - P(t_3)}{\frac{t_4 - t_3}{t_4 - t_1}}\right] \cdot C_T \cdot C_R \tag{11}$$

$$C_T = \frac{t_3 - t_1}{t_4 - t_1} \tag{12}$$

Table 6; Yarveisy et al. Resilience Metric Usage

Shape	Value of R_6	Works?	Explanation
Baseline	$0 < R_6 < 1$	Yes	
Baseline with Performance Lost/Gained	$0 < R_6 < 1$	Yes	
Bucket	$0 < R_6 < 1$	Yes	
V/U/Scoop	$0 < R_6 < 1$	Yes	
No Recovery, Smooth No Recovery	DNE	No	t_3, t_4 do not exist
Multiple Minima	$0 < R_6 < 1$	Yes	

This metric returns meaningful values for every shape except when there is no recovery. In the no recovery case, t_3 and t_4 do not exist, which makes many of the values above nonsense. There is no way around this issue for this metric.

Metric 7: Cheng et al.

The final metric investigated is from Cheng et al. This metric is made up of the sum of two values: one that describes the absorptive capacity of the system and one that describes its restorative capacity. Absorptive capacity refers to the ability of the system to absorb shocks, or limit performance loss due to a disruption. Restorative capacity is the ability of the system to recover from a loss. The value for each of these capacities is made up of 3 values that describe different aspects of the performance curve. Each of the capacity values is multiplied by a coefficient, α or β. The sum of the two coefficients must be 1 but their values can be varied to emphasize the importance of the system's absorptive or restorative capacity if one is deemed more important than the other. It is assumed here that they are equally important, so $\alpha = \beta = 0.5$.

$$R_7 = \alpha(Ab) + \beta(Res) \tag{13}$$

Ab is comprised of three values that describe different aspects of the performance curve. δ_d is the ratio of the actual performance of the system to the ideal performance in the absence of a disruption. σ_d is the ratio of the minimum performance to the original performance. ρ_d accounts for natural degradation and is assumed to be 1, meaning there is no expected degradation.

$$Ab = \delta_d \sigma_d \rho_d \tag{14}$$

$$\delta_d = \frac{\int_{t_1}^{t_{min}} P(t)dt}{(t_{min} - t_1)P(t_1)} \tag{15}$$

$$\sigma_d = \frac{min(P)}{P(t_1)} \tag{16}$$

Res is comprised of the same three values as Ab, but for the restorative stage of the curve. ρ_r again accounts for natural degradation and is assumed to be 1, meaning there is no expected degradation.

$$Res = \delta_r \sigma_r \rho_r \tag{17}$$

$$\delta_r = \frac{\int_{t_{min}}^{t_4} P(t)dt}{(t_4 - t_{min})P(t_1)} \tag{18}$$

$$\sigma_r = \frac{P(t_4)}{P(t_1)} \tag{19}$$

Table 7: Cheng et al. Resilience Metric Usage

Shape	Value of R_7	Works?	Explanation
Baseline	$0 < R_7 < 1$	No	Uses time of minimum performance
Baseline with Performance Lost/Gained	$0 < R_7 < 1$	No	Uses time of minimum performance
Bucket	0	No	Uses time of minimum performance
V/U/Scoop	$0 < R_7 < 1$	Yes	
No Recovery, Smooth No Recovery	DNE	No	t_3, t_4 do not exist
Multiple Minima	$0 < R_7 < 1$	Yes	

For every shape except for "U"/"V"/"Scoop" and multiple minima, this metric does not return meaningful values. This is due to the metric relying heavily on the time of minimum performance. For any shape that levels out at the minimum performance between t_2 and t_3, there is no single time of minimum performance. This could lead to a variety of outcomes depending on how the software being used locates the minimum of a constant line, making the metric unusable in those cases.

Conclusion

The results from investigating the function of each of these metrics when applied to a variety of performance curve shapes are summarized in the chart below. A green box refers to a metric that produces a meaningful result for that shape. A yellow box indicates that meaningful values can be extracted from a metric for that shape with a slight adjustment to the method. A red box means that the metric does not produce a meaningful result for a curve of that shape.

Table 8: Resilience Metric Usage Conclusions

	Baseline	Baseline w/ Loss/Gain	Bucket	U/V/ Scoop	No Recovery, Smooth	Multiple Minima
Das et al.						
Henry & Ramirez-Marquez 1						
Bruneau et al.						
Ayyub et al.						
Henry & Ramirez-Marquez 2						
Yarveisy et al.						
Cheng et al.						

This chart shows that the metrics that are meaningful for every shape investigated are the Das et al. metric and both Henry & Ramirez-Marquez metrics. All three of these require a slight modification to the equation when there is no recovery, but this can be handled through a few basic lines of code. If a study is not dealing with any curves that do not show any recovery, or if curves with no recovery are automatically assigned the worst possible score for a metric, the Bruneau et al. and Yarveisy et al. metrics can also be used.

1.2.6 Development of the Modular Coupled Virtual Testbed

Now that we have defined resilience for our purposes, brainstormed potential hazards and safety controls, created a metric to inform design decision-making, and identified how to evaluate the resilience of a habitat quantitatively, we can build a Resilience-Oriented Habitat Design Process, and validate and demonstrate it using one of RETHi's simulation platforms.

We will perform the validation and demonstration using RETHi's Modular Coupled Virtual Testbed (MCVT) simulation platform. The MCVT is based on RETHi's Notional Real Habitat (NRH). The NRH is a conceptual habitat design created to inform the development of simulation tools. The NRH consists of a dome with a specified size and shape. Inside the dome there are specified locations for the ECLSS equipment, power equipment, and airlock. The dome can be treated as one interior environment, or a door can be closed through the middle of the dome, creating two separate zones.

The MCVT is a $1/5^{th}$ scaled rendition of the NRH to achieve close to real-time simulation.

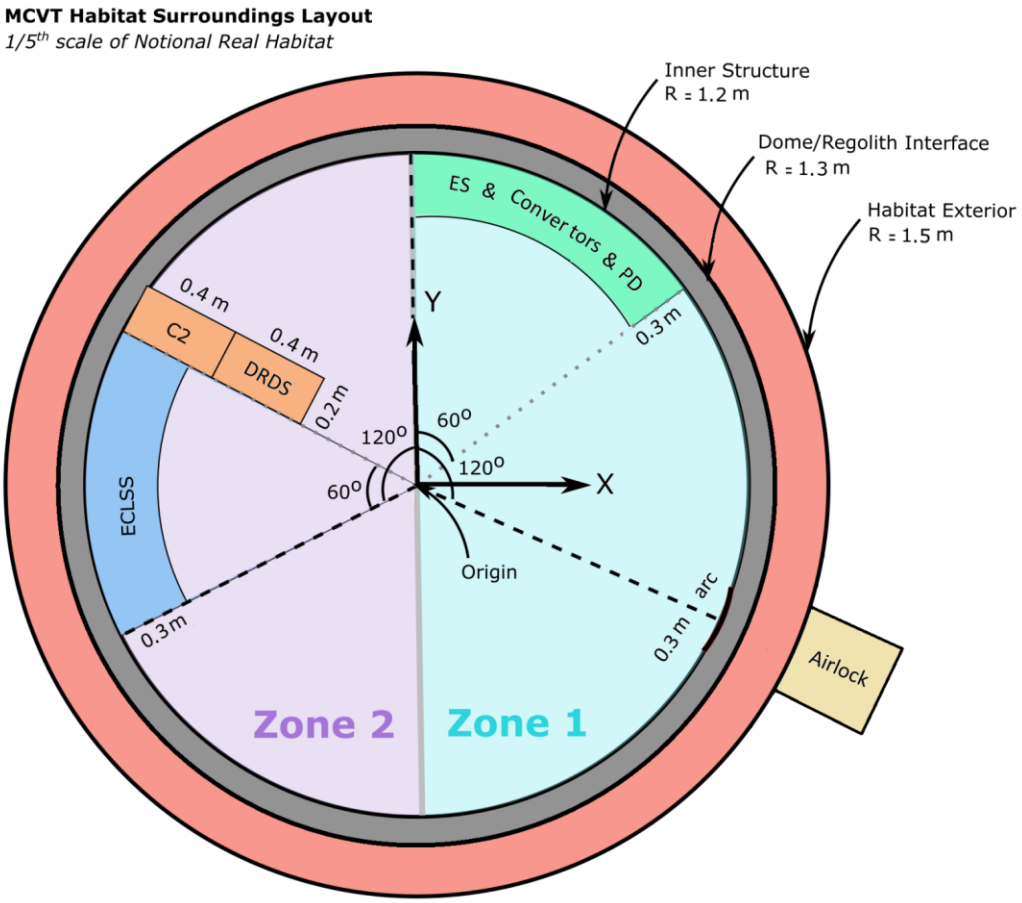


Figure 5: MCVT Dome Interior

There is also a schematic for the equipment that sits outside the dome. Figure 6 shows where the solar panels, nuclear reactor, radiator panels, inventory, and launch/landing site are relative to the dome at the center. The inventory is a storage facility for materials, including those needed to perform safety controls.

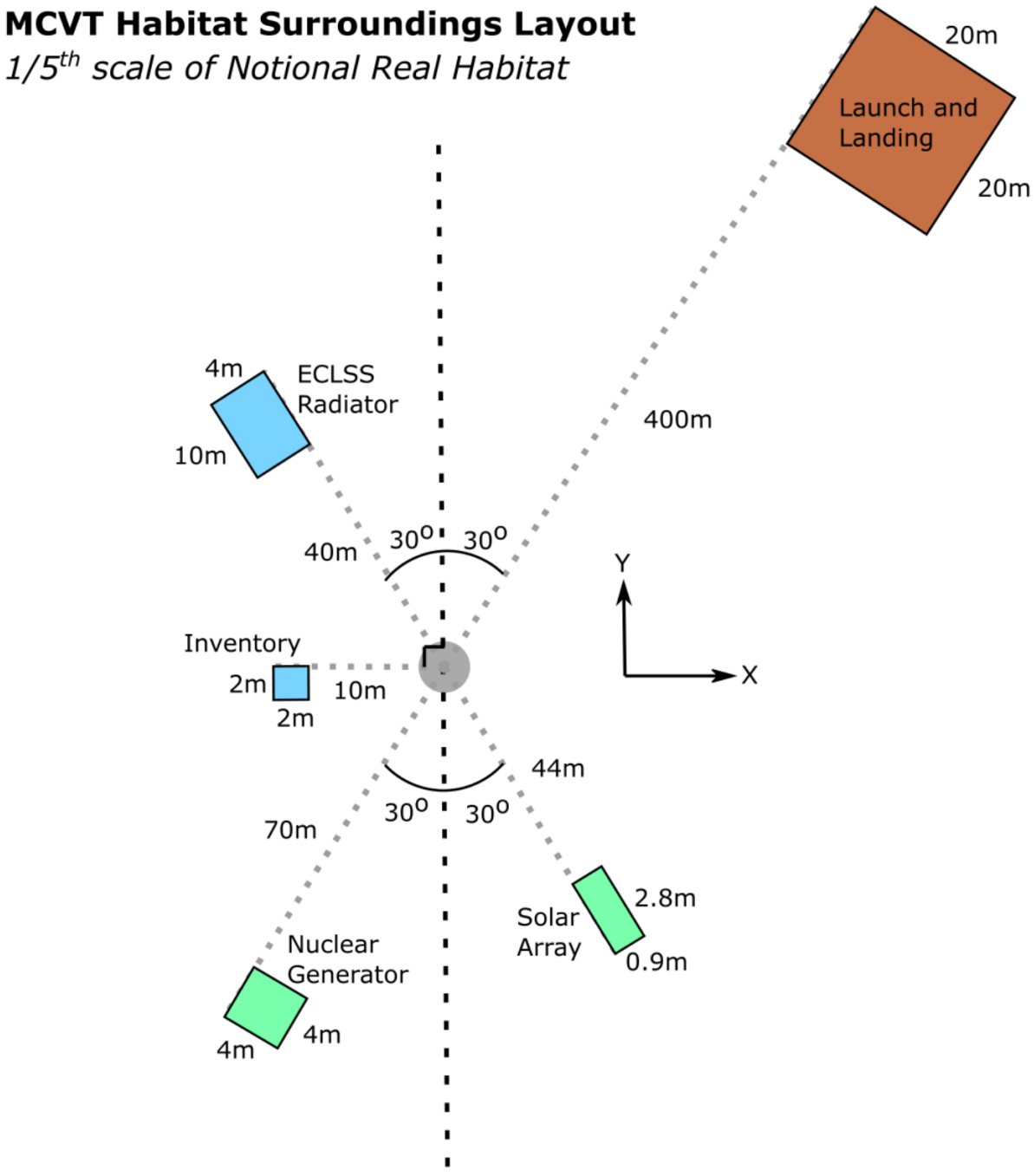


Figure 6: MCVT Dome Exterior

The MCVT simulates the subsystems required for a habitat to function and their interactions. The platforms can run nominally (i.e., nothing goes wrong), or users can input disruptions to see how the habitat responds to different faults. Users can also select which safety controls are present on a habitat and how well they work to investigate how the habitat might recover from disruptions.

2. DEVELOPING A RESILIENCE-ORIENTED HABITAT DESIGN PROCESS

2.1 Proposed Design Process

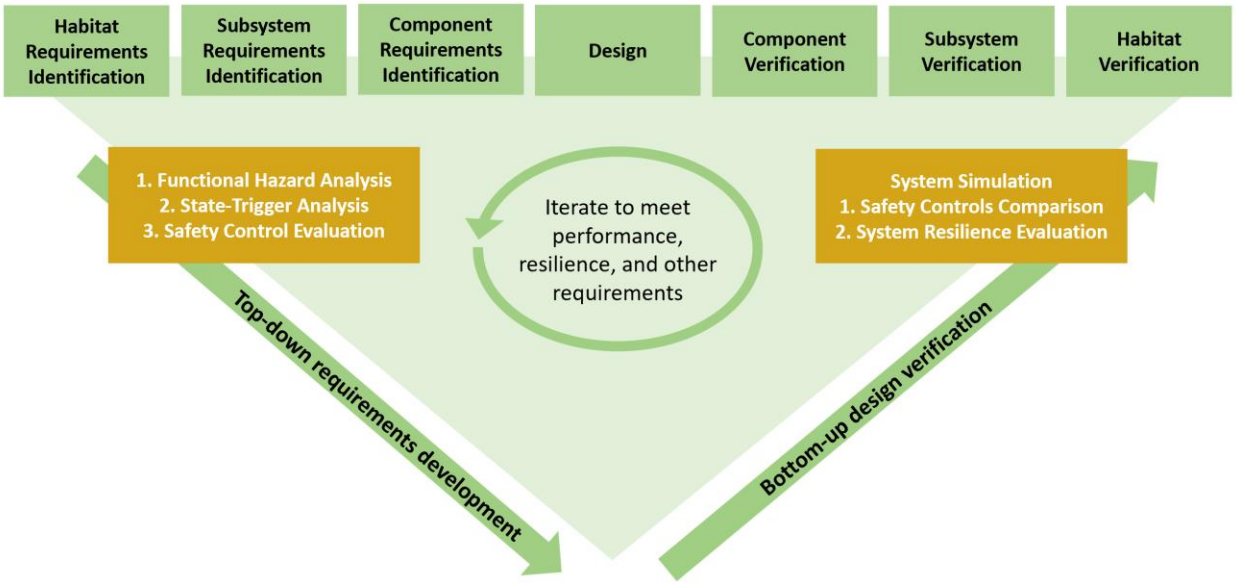


Figure 7: Proposed Resilience-Oriented Habitat Design Process

Figure 7 outlines the design process. The arrows at the bottom outline the typical steps of any design project, starting with top-down requirements development and then moving into bottom-up verification of those requirements for a particular design.

The steps in the boxed at the top break the design process down into system levels to guide the design process. The yellow boxes on each side of the triangle represent the steps that should be taken to build resilience into each step of the design process.

During the top-down requirements phases, three steps must be taken to design for resilience at each level of design.

1. The first step is a Functional Hazard Analysis. The FHA is a method originally developed for preliminary aircraft safety assessments (Kritzinger, 2016). We use an FHA to develop an

understanding of what might happen if the functions necessary in a system are not present. For the purposes of resilient habitat design, we use an FHA with three columns. In the first column, we list all the functions that the habitat needs to have at the system level in question. We start at the whole-habitat level by listing basic functionalities of the habitat. In the second column, we list possible hazardous states that could affect those functionalities. The final column lists safety controls that could rectify each of those hazardous states, restoring the affected functionality. The purpose of this exercise is to get an idea of what kinds of hazardous states we should be designing for.

2. The FHA develops a basis for the next step, which is a State-Trigger Analysis of each system level. The STA lists all the possible things that could go wrong within a particular system and links them to the hazardous states that would result if they were to happen. When all the states and triggers and their links are compiled into a state-and-trigger network, we can analyze the results. From the network, we can determine which disruptions result in the highest number of hazardous states, and which result in the fewest. We can also see which hazardous states result from many triggers and which only happen under a few circumstances. This type of analysis gives us an idea of which disruptions and hazardous states should be prioritized for safety controls.

   The STA does not represent the difference in severity between hazardous states, or the difference in likelihood of disruptions. A hazardous state that only results from one disruption might look like a low priority based on the STA, but if it is a disruption that is likely to occur causing a hazardous state that has severe consequences, it should be prioritized. The STA is a starting point for that kind of analysis.

3. The third step is an evaluation of the safety controls that could mitigate each hazardous state. We brainstorm safety control ideas, then calculate their Control Effectiveness.

Once these three steps have been taken at the habitat, then subsystem, and then component level, we begin moving up the right side of the chart. The right side of the chart is where we use simulations to choose the set of safety controls with which we equip the habitat before analyzing the resulting habitat design.

To move up the right side of the diagram, we start at the lowest level with individual components. We select any component safety control we came up with in the first half of the design process and use the MCVT to perform a high-fidelity simulation of how the component performs with one implementation strategy of the chosen safety control during a disruption scenario designed to target the component. Once a series of simulations have been carried out with an implementation strategy, we select the best performing implementation strategies for each safety control to carry up to the subsystem-level investigation.

We then perform a subsystem-level investigation by combining the chosen component-level implementation strategies in each subsystem and analyzing how the subsystem performs before combining the subsystems into a whole habitat.

Then we analyze the habitat design to determine if it meets performance, resilience, and other requirements. Other requirements on a mission include cost, weight, mission lifetime, or number of astronauts. These factors all affect the number of safety controls and the types of implementation strategies that will work for a mission. If there were no constraints, the most resilient habitat would of course include a highly effective safety control for every single possible hazardous state. In a real mission design, options will be limited by constraints.

Then, the arrow in the center of the chart denotes that we then select a different implementation strategy and perform the same sequence of simulations. We can iterate through this loop as many times as there are different safety controls and implementation strategies for the component in question, compare the performance of the resulting habitats, and select the best one.

2.2 Developing a Disruption Scenario

Before we can dive in to testing out the design process, we must decide what kind of disruption we want to use to do so. There are a number of pre-existing disruptions in the MCVT that we can use for this research: Meteorite Impact, Fire, Moonquake, Airlock Failure, Launch/Landing Event, and Sensor Failure.

We want to use a disruption that will allow us to demonstrate every step of the proposed design process, meaning we need a disruption scenario that involves failures at every system level. To make the disruptions more realistic and complex, there are three characteristics we want to work into the disruption scenarios we choose.

1. Safety control implementation strategies

We want to investigate hazardous states that can be rectified in multiple ways. The user can choose to equip the habitat with different safety controls for each hazardous state and compare the resilience results. In the MCVT, this can be done by altering the Agent model, which simulates safety controls being carried out. To compare the habitat's reaction to different implementation strategies, we can change the amount of time a safety control takes to carry out, and the probability that it is successful.

2. Decision making in crewed and uncrewed configurations

We also want to compare the habitat's performance when the habitat has human crew vs. when it does not. In the MCVT, we again use the Agent model to simulate this comparison. We can change the amount of time it takes the agent to make a decision about what to do and prepare to carry out the safety control, as well as how fast the agent moves through the habitat.

3. Cascading failures

Finally, we want to include examples of disruptions that have cascading effects the affect each system level. Cascading effects are failures that happen as a result of previous failures. At this time, there are no disruptions in the MCVT that have cascading failures, so we design a new disruption that involves cascading effects.

2.2.1 Cooling System Cascade

The Cooling System Cascade (CSC) is the disruption scenario we design to incorporate all three elements of complexity.

The CSC takes inspiration from an incident aboard the ISS involving leaking coolant in the system that cools the station's power system (Harwood, 2013). The leak was first noticed in 2007 when astronauts saw ammonia dissipating into space outside the station, likely as a result of a micrometeorite impact. At the time, the leak was slow. The mitigation plan was to let the leak continue and refill the coolant every four years. In 2013, however, the leak severity quadrupled, leading to an emergency spacewalk to reconfigure the coolant pipes such that the leaking section could be removed.

The following description of the CSC is what would happen if there were no intervention. Activating safety controls will stop the cascade of failures, allowing the habitat to recover.

Cooling System Cascade Description

The scenario starts with a leak in the system that cools the nuclear radiator panels. The leak is outside of the habitat, so an agent is sent outside the habitat to patch the leak.

While the agent is gathering materials and preparing for the repair, the nuclear power generation system is becoming less efficient due to overheating. If the disruption happens at night, the habitat is also not generating solar power, so the only power generation is from the nuclear power generation system. If the leak is severe or unaddressed for long enough, the nuclear power generation will decrease in efficiency until it is generating less than the habitat power demand. At this point, the habitat will begin to use battery power.

Eventually, the habitat can run out of battery power. As the habitat approaches power loss, the Smart Power Distribution system prioritizes critical loads, like thermal and pressure control. When the life support loads are prioritized, less important power demands are not met. Unnecessary lighting, scientific instruments, etc. are switched off, meaning the habitat is supporting human life but not meeting mission objectives.

If the disruption in the nuclear cooling system is still not rectified and the sun has not risen to provide solar power, the habitat can completely run out of power, sending the habitat into an accident state.

Figure 8 below depicts how the CSC plays out using the state-trigger approach.

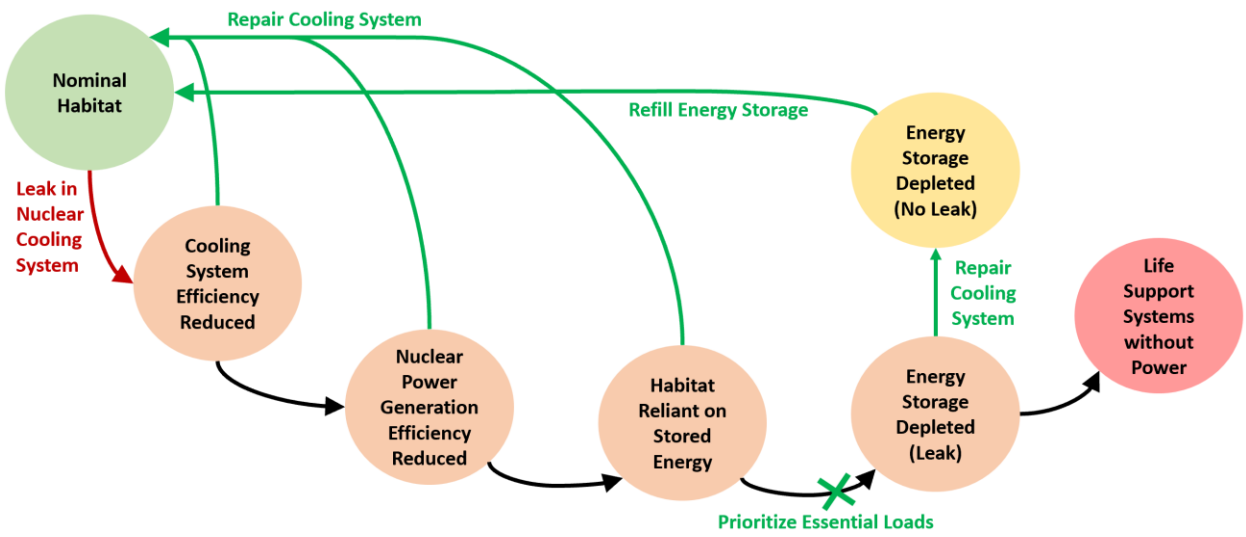


Figure 8: Cooling System Cascade

2.3 Top-Down Requirements Development

Now we are ready to begin following the steps laid out in the Resilience-Oriented Habitat Design Process. The first half of the process deals with selecting safety controls to evaluate.

2.3.1 Habitat-Level Requirements Identification

Before we can select safety controls to include in a habitat, we must brainstorm what kinds of hazardous states the safety controls will need to address. Initial brainstorming can be done using a Functional Hazard Analysis. The FHA below lists the functions that the habitat needs to have at the whole-habitat level. Then, it lists hazardous states that could prevent the habitat from performing each function, and safety controls that could mitigate those hazardous states. The FHA below is not comprehensive; it focuses on the most basic functions that relate to the Coolant Leak scenario.

Table 9: Habitat-Level FHA

Function	Hazardous States	Safety Controls
Provide Power to Support Habitat Loads	Not Providing Power to Support Habitat Loads	Repair Power System
Maintain Livable Atmosphere	Not Maintaining Temperature and Pressure Set Points	Repair ECLSS System
	Not Sealing Interior Environment from Exterior Environment	Repair Structural Mechanical Layer
		Repair Structural Protective Layer

We also use a state-trigger analysis of the initial disruption in the CSC scenario to visualize the hazardous and accident states that we want to prevent. As with the FHA, this STA focuses on the Coolant Leak disruption scenario.

At the habitat level, this is a very basic chain of events. The system starts in a nominal state. A disruption occurs in the power system. The first effect of this disruption at the habitat level is that it could lead to power loss, which would render the habitat's life support systems inoperable. The habitat enters the hazardous state, *Not Providing Power to Support Habitat Loads*. To prevent the hazardous state *Not Providing Power to Support Habitat Loads* from resulting in the accident state *Life Support Systems Cannot Function*, we can implement the safety control *Repair Power System* to return the habitat to its nominal state. As we move down the left side of the Resilience-Oriented Habitat Design Process, we will examine these states and safety controls more closely to understand exactly what a safety control like *Repair Power System* might entail.

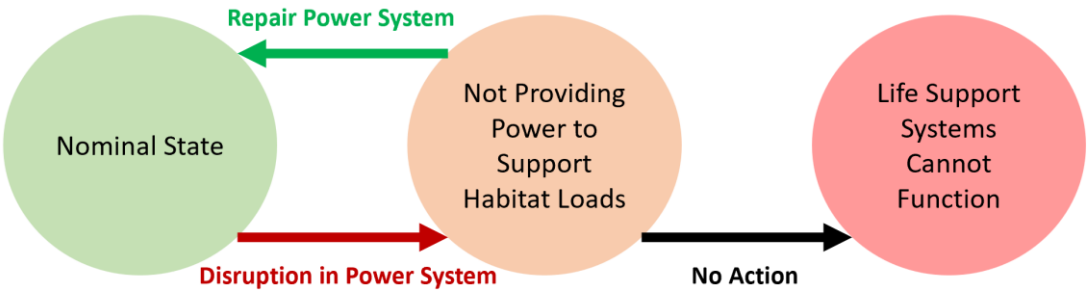


Figure 9: Habitat-Level STA

The only safety control in the habitat-level STA is *Repair Power System*. We can now brainstorm the different ways that this safety control could be carried out. *Repair Power System* could mean that the nuclear or solar power generation system needs repairing, or the power converters are not functional, or the energy storage system is empty or broken. In the next step, we take a closer look at the possibilities by focusing in on particular subsystems.

2.3.2 Subsystem-Level Requirements Identification

Now we move down to the subsystem level and perform the same process, starting with an FHA. We break down each of the hazardous states from the habitat-level FHA into specific potential hazardous states for the subsystem in question, starting with the power subsystem. For the habitat-level hazardous state *Not Providing Power to Support Habitat Loads*, we want to list the possible reasons why this could be happening within the power subsystem. This creates four more specific hazardous states, for which we then develop safety controls.

Table 10: Subsystem-Level FHA

Subsystem	Function	Hazardous state	Safety controls
Power	Generate power	Not Generating Nuclear Power	Repair Nuclear Power Generation System
		Not Generating Solar Power	Repair Solar Power Generation System
	Convert Power	Not Converting Power	Repair Power Conversion System
	Store Energy	Not Storing Energy	Repair Energy Storage System

Below is the state-trigger analysis for the power subsystem during the coolant leak scenario. The STA breaks down what happens in the power subsystem between *Disruption in Power System* and *Not Providing Power to Support Habitat Loads*, which are directly linked in the habitat-level STA. The exercise illuminates the subsystem-level intervention and mitigation safety controls that go into returning the power subsystem to its nominal performance. When the leak happens, we can see that there are intermediate hazardous states between the nominal state and *Not Providing Power to Support Habitat Loads.* First, the nuclear panels will overheat, reducing their efficiency. At that state, we can implement the safety control of fixing the nuclear cooling system, or we can take no action. If no action is taken, the habitat will eventually rely on, and then run out of battery power. Before the battery power is completely gone, we can implement the prevention safety control, *Prioritize Essential Loads.* This results in more time to fix the cooling system before the hazardous state *Energy Storage Depleted* occurs. However, if the habitat enters the hazardous state *Energy Storage Depleted*, simply repairing the cooling system is no longer enough to return the habitat to a nominal state. At this point, *Repair Cooling System* is an intervention safety control, putting the habitat in a temporary safe state, *Energy Storage Depleted (No Leak).* From this

temporary safe state, we employ the mitigation safety control, *Refill Energy Storage*, to return to nominal.

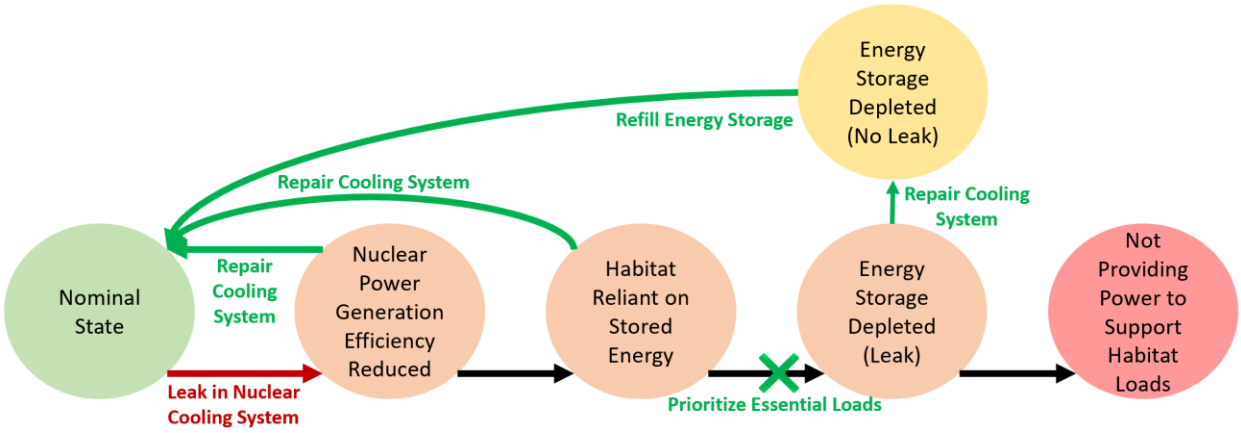


Figure 10: Subsystem-Level STA

2.3.3 Component-Level Requirements Identification

The final step in the left side of the design process is the component-level FHA and STA. As with going from habitat- to subsystem-level, the component-level charts provide further insight into what could happen at the component level that would result in the hazardous states, accident states, and safety controls we identified at the subsystem level. Where, in the power system FHA, we noted hazardous states like *Nuclear Power Generation System Not Functional* and *Solar Power Generation System Not Functional*, we now consider what disruptions could cause those hazardous states to happen. The result is hazardous states like *Nuclear Radiator Panels Covered with Dust* and *Solar PV Arrays Damaged.*

Table 11: Component-Level FHA

Component	Function	Hazardous state	Safety control
Nuclear Reactor	Provide Nuclear Power	Radiator Panel Covered with Dust	Clean Radiator Panel
		Cooling System Not Functional	Repair Cooling System
		Radiator Panel Damaged	Repair Radiator Panel
			Replace Radiator Panel
Solar PV Array	Provide Solar Power	Solar PV Array Panels Covered with Dust	Remove dust from Solar PV arrays
		Solar PV Array Damaged	Repair Solar PV Array
			Replace Solar PV Array
Power Converters	Convert Power	Power Converter Broken	Repair Power Converter
			Replace Power Converter
		Power Converter Overheated	Cool Interior Environment
Battery Cells	Store Energy	Battery Cell Broken	Repair Battery Cell
			Replace Battery Cell

Once again, we can visualize the states and triggers that make up the CSC at the component level using a STA. Below is the STA for just the nuclear cooling loop during the leak scenario. Here, we break down what happens at the component level to cause the nuclear radiator panels to overheat in the first place. When the leak occurs, the coolant loop enters the hazardous state, *Cooling System Efficiency Reduced* with an active leak. One thing that can happen at the leaking state is the intervention safety control, *Patch Leak.* This moves the system to a state where the cooling loop is fixed, but still operating worse than nominally because of the reduced volume of coolant in the loop. From this state, we can add more coolant to return to the nominal state. However, if no safety control action is taken while the loop is in the *Cooling System Efficiency Reduced* hazardous state, the disruption could begin to affect the nuclear power generation, reducing its efficiency due to overheating. Once the loop enters the accident state, *Nuclear Power Generation Efficiency Reduced*, we can implement the mitigation safety control of adding more coolant without patching the leak to shift back to the leaking state, or we can choose the intervention safety control of patching the coolant line. Applying a patch moves the loop to a state where it is fixed but missing coolant, and we can add coolant back to the system to return to a nominal state.

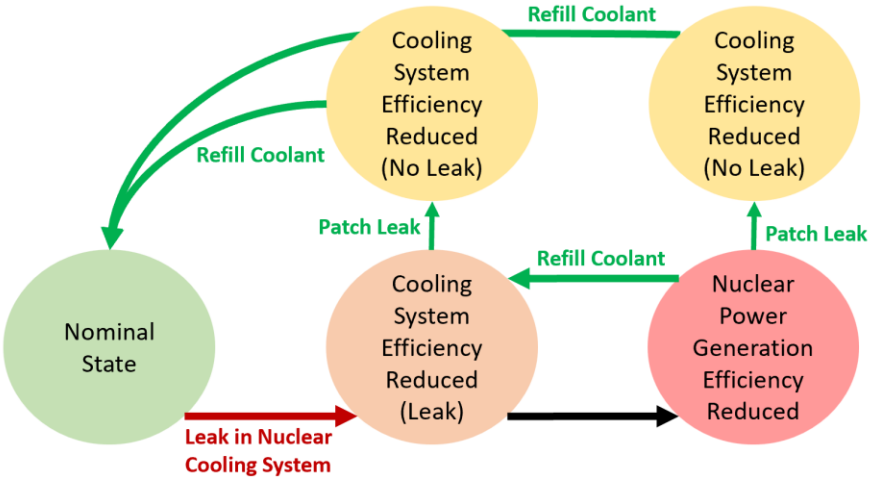


Figure 11: Component-Level STA

Now we can brainstorm specific ways to carry out the safety controls identified throughout the whole FHA/STA process. In the component-level STA we again have the safety control *Patch Leak.* We can now identify that patching the coolant loop could be done by a human or by a robot.

Another safety control we identified is *Prioritize Critical Loads* at the subsystem level. Load prioritization can be done either by having a human switch the power system mode, or automatically by the habitat's command center at a predetermined level of battery power.

2.3.4 Safety Control Evaluation using Control Effectiveness

At this point in the design process, we can perform a safety control evaluation. The evaluation serves to compare different implementation strategies we could select to carry out a particular safety control.

The repair in question for this safety control is applying a patch to the leak. There are a few methods that could be used for patching: a leak sealant that can be injected into the coolant line, an epoxy seal, and brazing. An injectable leak sealant is a powder that is poured into the coolant line at the valve (Sellén, 2023). The powder reacts to the condensation that happens at the location of a leak in the line and plugs the hole. These kinds of leak sealants are only effective with very small leaks. Epoxy putty can also be used to patch holes (*American Leak Detection*, 2015). Using epoxy entails draining the coolant and cleaning the area around the leak, applying epoxy putty, and allowing the epoxy to cure. Brazing is a method of welding in which a filler metal with a lower melting point than the pipe in need of repair is melted and used to fill the leak (HVAC, 2018). Each of the three techniques can be performed by either a human or a robot. We assume that the robot in question is sophisticated enough to carry out any repair technique.

Now we will compare the control effectiveness of each method. In determining these values, we consider how each technique is performed on Earth to get a general understanding of how they compare to each other, then use judgement to make the final selections. In the future, we envision control effectiveness values being informed by data and experimentation, making them reliable enough to be used alone in making design decisions.

The first value we need to assign for each implementation strategy is the probability of availability. For the purposes of this research, we assume that the habitat is equipped with the necessary supplies for the repair in question, so each implementation strategy will have $\boldsymbol{P_{available} = 1}$.

The next value is the probability of competent design. Based on what we learned about these techniques, the leak-sealing method listed here that is most likely to successfully eliminate the hazardous state is brazing, followed by epoxy, followed by an injectable sealant. This component of Control Effectiveness only varies with the design of the technique, so $\boldsymbol{P_{design}}$ will not change depending on what kind of agent is performing the safety control. Some implementation strategies will have different probabilities of competent design depending on the severity of the hazardous state. Such implementation strategies have multiple columns under $\boldsymbol{P_{design}}$ indicating how the value varies with the severity of the disruption. With this information in mind, we assign the values for $\boldsymbol{P_{design}}$ that can be found in Table 12.

The third component of Control Effectiveness is the probability of perfect implementation. Because all of the implementation strategies here involve complex movements (as opposed to automated functions), a robot will be less likely to implement them perfectly than a human. The techniques' complexities are inversely proportional to their probability of competent design, so the injectable sealant will have the highest $\boldsymbol{P_{implementation}}$, followed by epoxy, followed by brazing.

The final Control Effectiveness value is the response margin. Response margin compares the amount of time a safety control takes to perform ($\boldsymbol{t_{affect}}$) to the amount of time the disruption being rectified takes to negatively impact the habitat ($\boldsymbol{t_{effect}}$). For the coolant leak disruption, we assign $\boldsymbol{t_{effect}}$ the time when the habitat's stored energy runs out. That time changes depending on the intensity of the leak and the time of day of the disruption. If the habitat's circumstances are such that the stored energy never runs out as a result of this disruption, the response margin is the ideal 1. $\boldsymbol{t_{affect}}$ is the repair time of the safety control plus the time the agent takes to prepare for the repair action. If $\boldsymbol{t_{affect}}$ is *larger* than $\boldsymbol{t_{effect}}$, the response margin does not exist.

To assign values for response margin, we must determine the amount of time that each repair will take. One assumption we make throughout this research is that a robot will take double the amount of time that a human will take to perform repair tasks. This assumption is made out of necessity, as we do not have any empirical data to base our values on. The lack of a perfect model for the amount of time it takes to perform these repairs does not undermine the integrity of the research, because we just want to compare these implementation strategies to each other. All that matters is that we can tell which strategies will likely take longer to carry out than others.

The injected leak sealant is the simplest method of patching a leak. It only entails acquiring the sealant, opening the coolant valve, and adding the sealant. We assign a repair time of 20 minutes for a human injecting sealant, and 40 minutes for a robot.

Both the epoxy method and the brazing method require that the coolant pipe is emptied and the area surrounding the leak is dried before patching. We assign 60 minutes for a human to drain and dry, and 120 minutes for a robot.

After acquiring supplies, which can be done while the coolant is draining, applying epoxy does not take long. It comes in the form of strips or putty that can simply be placed over the hole, which would take only a few minutes. The time-consuming part of the epoxy method is the time it takes to cure. Typical cold-weld epoxy takes 4–6 hours to set and 15–24 hours to fully cure. The coolant can be refilled before the epoxy is fully cured, but not before it has set. We will use a repair time of 6 hours for the epoxy method done by a human and 7 hours for a robot. This does not follow our rule of thumb of doubling the time for a robot because the only part that would take longer would be the epoxy application. The drying time is the same regardless of the type of agent.

Brazing requires one hour for draining coolant and preparing the area around the leak, during which supplies could be acquired. Once that is done, the time the actual brazing process takes depends on the size of the hole. We will use a range of 60–100 minutes for a human to braze, and 120–200 minutes for a robot. This makes the total time 120–160 minutes for a human and 240–320 minutes for a robot.

Taking these repair times into account, the values for response margin for each implementation strategy are in Table 12.

Table 12: Control Effectiveness of Leak Repair Implementation Strategies

Safety Control	Method		$P_{available}$	P_{design}		$P_{implementation}$	$M_{response}$	
							☼	☾
Repair Cooling System	Injectable Leak Sealant	👤	1	0.4	0.2	0.9	1	0.28
		🤖	1	0.4	0.2	0.7	1	ø
	Epoxy Resin	👤	1	0.4		0.7	1	ø
		🤖	1	0.4		0.5	1	ø
	Brazing	👤	1	0.8		0.5	1	ø
		🤖	1	0.8		0.3	1	ø

0.00 – 0.45	0.45 – 0.85	0.85 – 1.00

2.4 Bottom-Up Design Verification

In this section we use the MCVT to compare the performance of the components, subsystems, and habitat when we make different safety control selections. The selections being made are the technique used to patch the leak and the type of agent performing the safety control.

In the MCVT, there are three parameters that affect how the agent performs each safety control.

1. Agent Speed

The first parameter is agent speed, referring to the speed with which the agent moves around the habitat. The MCVT uses the agent speed and the distances between components in the NRH to calculate the amount of time it takes the agent to move between locations to perform repairs.

In a crewed configuration, we assume that every safety control that requires agency is performed by a human. According to NASA, the Apollo astronauts moved at an average of 0.61 m/s on the moon's surface, noting that their slow speed was a result of the heavy, inflexible space suits they were wearing (De Witt et al., 2014). An International Journal of Advanced Robotic Systems paper

on planetary rover navigation estimates a micro rover's speed to be approximately 0.01 m/s, which we will use for our robot's agent speed (Ilyas et al., 2016).

2. Repair Time and Repair Rate

The second parameter is either repair time or repair rate, depending on the type of repair.
Repair time is used when a component's health state is binary. A binary health state is characteristic of a component that is either fully functional or completely broken in the MCVT. Such components cannot be repaired a little at a time, or are not functional while being repaired. Repair time for binary health states is the amount of time it takes to perform the repair.

Components that have continuous health states use a combination of repair time and repair rate. The repair time here represents the amount of time the agent spends working on the component. The repair rate affects how much healthier the component is at the end of the repair time compared to where it began. The repair *time* can be adjusted to evaluate different safety control prioritizations. The repair *rate* can be adjusted represent the capability of the agent performing the repair. For example, if Agent A can repair a component more efficiently than Agent B, Agent A would have a higher repair rate. Agent A would then cause more improvement in the component's health state than Agent B in the same repair time.

3. Agent Preparation Time

The final parameter is the agent preparation time. This represents the time it takes for an agent to recognize an issue and decide what to do before it actually starts moving. Preparation time is also used to account for the time it takes an agent to prepare to perform a task. For example, if the agent is human and the repair is outside of the habitat, then the preparation time would include the time it would take the human to prepare to leave the habitat.

The most time-consuming task a human must perform before a spacewalk is prebreathing. Prebreathing is when astronauts slowly alter the composition and pressure of the air they are breathing to avoid symptoms of decompression sickness upon exiting the habitat. While current spacesuits require prebreathe times upwards of 2 hours, next-generation space suits are being

designed to operate at a higher suit pressure, significantly reducing prebreathe times (Wada). By the time habitats are being constructed on extra-terrestrial surfaces, we can conservatively predict that prebreathe times will be on the order of minutes rather than hours. We will use an agent preparation time of 30 minutes when the agent is a human that needs to leave the habitat.

In the Notional Real Habitat on which the MCVT is based, most safety control supplies are kept in an Inventory building 10 m from the central dome. The agent performing a safety control requiring supplies from the Inventory must travel to the Inventory and then to the location of the hazardous state before it can begin repairing. Because the Inventory building is outside, a human will have to don an EVA suit and perform all the necessary preparations before leaving the central dome if they need supplies from the Inventory.

2.4.1 Component-Level Verification

We start at the component level by examining the component most affected by the Cooling System Cascade: the nuclear reactor.

First, we establish values for the MCVT inputs we have discussed.

The first implementation strategy we will investigate using the MCVT is the brazing technique. We choose the brazing technique because it has a high probability of competent design and a short enough repair time to simulate in real time using the MCVT.

Table 13 outlines the values we input into the MCVT. The Agent Speed and Prep Time take the values we outlined above, but we determined that there was a range of repair times for this implementation strategy. We input different repair times depending on the severity of the leak, which can be controlled using a parameter called *Intensity Level* (IL) in the MCVT. Intensity Level is a characteristic of a disruption that specifies its severity. Intensity Level ranges from 1 (no disruption) to 5 (most severe). For this research, we will compare the results of Intensity Levels 1, 3, and 5.

Table 13: Brazing MCVT Inputs

	Human			Robot		
Intensity Level	1	3	5	1	3	5
Agent Speed (m/s)	0.61			0.01		
Prep Time (minutes)	30			0.5		
Repair Time (minutes)	N/A	120	160	N/A	240	320

We input these values into the MCVT and run a variety of simulations varying the intensity level, type of agent, and time of day. Figure 13 shows some key metrics of the habitat when the intensity level is 1, meaning there is no disruption. The two lines on each plot show the habitat's behavior at dawn compared to at midday. This figure provides a baseline to which we compare the other simulations.

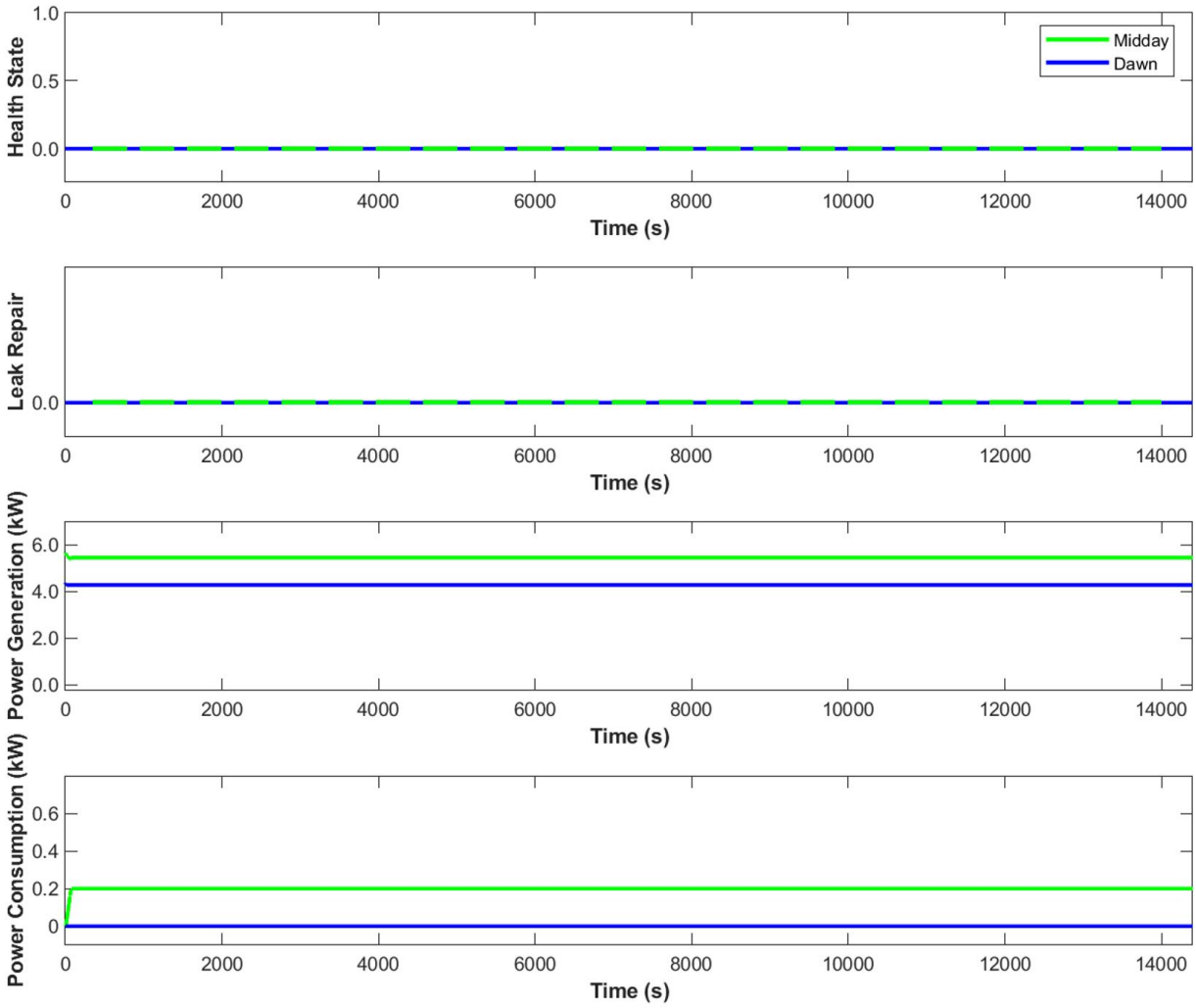


Figure 12: Nominal Habitat Operations Starting at Midday and Dawn

The first subplot shows the health state of the nuclear coolant line. The health state is a binary signifier of whether a component in the MCVT is damaged. The health state is zero when a component is healthy and one when it is damaged. In these nominal simulations, the health state is always zero.

The second subplot shows the repair action. While a repair is in progress, it shows the repair rate input by the user for the particular repair action being plotted, even when the repair is binary and therefore does not use a repair rate. In the binary case, the repair rate is always 5. The second subplot will show zero when there is no repair action and 5 when the leak is being repaired. In this case, there is no disruption and no need for repair, so the leak repair plot remains at zero.

The third subplot shows nuclear power generation, which is most efficient when the reactor is at its ideal temperature and less efficient if it is warmer or colder than ideal (Chebbo et al., 2023). If the exterior environment is colder than the ideal temperature, nuclear power is generated at a reduced efficiency, as in the dawn simulation. If the exterior environment is warmer than the ideal temperature, the cooling loop cools the reactor to the ideal temperature to maximize efficiency of power generation, as in the midday simulation. The environment outside the habitat in the MCVT changes temperature as a function of the angle of the sun. The solar angle is between zero and 180 degrees, with zero degrees meaning sunrise and 180 degrees meaning sunset. The user inputs the initial solar angle when starting a simulation, and the sun moves throughout the simulation. The rate at which the sun moves is such that the sun would move the full range from zero to 180 degrees over the course of one lunar cycle (~29.5 days). The change in solar angle is nearly imperceptible in a four-hour simulation, so the nuclear power generation appears constant at each time of day we simulate.

The fourth subplot shows the power consumption of the coolant pump. The pump consumption starts at zero during both simulations and remains at zero during the dawn simulation when the exterior environment is colder than the ideal reactor temperature. At midday, the pump consumption quickly rises as the pump adds more coolant to the cooling loop to keep the reactor at the ideal temperature.

The next series of plots depicts these values for three simulations run during the day with the human values with varied intensity level.

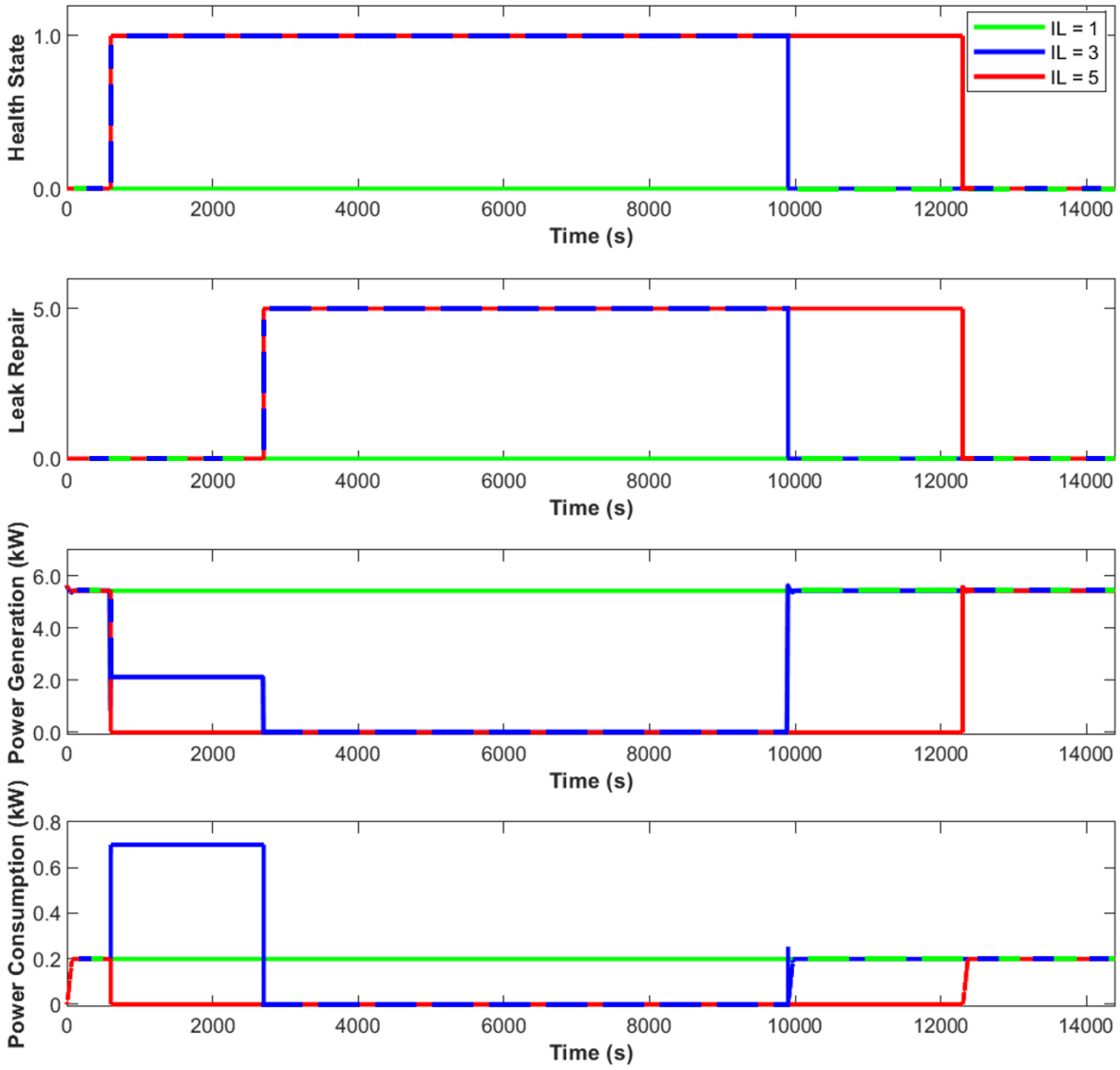


Figure 13: Habitat Response to Varied Intensity Coolant Leak at 600s with Human Agent Starting at Midday

This series of simulations shows variations in the health state and repair action plots compared to the habitat's nominal operation. In the first subplot, the coolant leak causes the health state of the nuclear reactor to flip to one at the time of the disruption (10 minutes, or 600 seconds) for intensity levels 3 and 5, signaling that the component is damaged. 30 minutes after the leak begins, the human finishes suiting up and prebreathing and begins the process of repairing the leak (second subplot). When the intensity level is 3, the repair takes less time than when it is 5 because the leak is smaller, so less brazing needs to be done to patch it. Around 10,000 seconds (~2 ¾ hours), the IL 3 leak is patched, so the component's health state returns to zero. The change in health state

from damaged to healthy tells the human that the repair is finished, so the human stops performing the leak repair action. The same series of events happens in the IL 5 simulation about 2,000 seconds (~ ½ hour) later.

The third subplot shows how nuclear power generation is affected by the disruption. For intensity level 3, the initial disruption reduces the efficiency of the nuclear power generation by about half. For IL = 5, the leak is so severe that the nuclear reactor is isolated from the habitat instead of trying to make up for the leaking coolant. The reactor's isolation means that power generation drops to zero immediately following the disruption. When the repair action begins, the IL 3 line joins the IL 5 line at zero, because the safety protocol requires disconnecting the nuclear reactor from the rest of the habitat during repair. When the repair action finishes for each intensity level, nuclear power generation returns to nominal levels. At the beginning of all three simulations, and at time of repair for intensity levels 3 and 5, there are small bumps in power generation. The bumps display the difference between the ideal level of power generation (when the temperature is ideal) and the generation during midday temperatures. It takes about two minutes for the reactor to heat up, decreasing the output generation. We also notice a downward spike in nuclear power generation at the time of the leak. This spike will be seen in every intensity level 3 simulation in this thesis. The spike is a bug in the MCVT and has no physical source or meaning.

The fourth subplot shows the coolant pump power consumption. Power consumption starts at zero and quickly rises about 0.2 kW, which is the power consumption needed to maintain the ideal reactor temperature during midday temperatures. Notice that the increase in pump power consumption corresponds with the small bumps in power generation in the third subplot. The pump operates steadily at that power consumption until the disruption strikes at 10 minutes (600 seconds). At the time of the disruption, the pump power consumption increases for IL = 3 as the pump tries to maintain the correct coolant temperature by adding more coolant to the cooling loop. For IL = 5, the pump power consumption drops to zero immediately following the disruption because the cooling system shuts down completely. During the repair, both disruption simulations show a pump power consumption of zero, until each repair finishes and the respective pump power consumption returns to nominal.

Next, we compare these midday, human-agent simulations to the dawn, human-agent simulations. We compare the midday IL 3 simulation to the dawn IL 3 simulation, both with human agents.

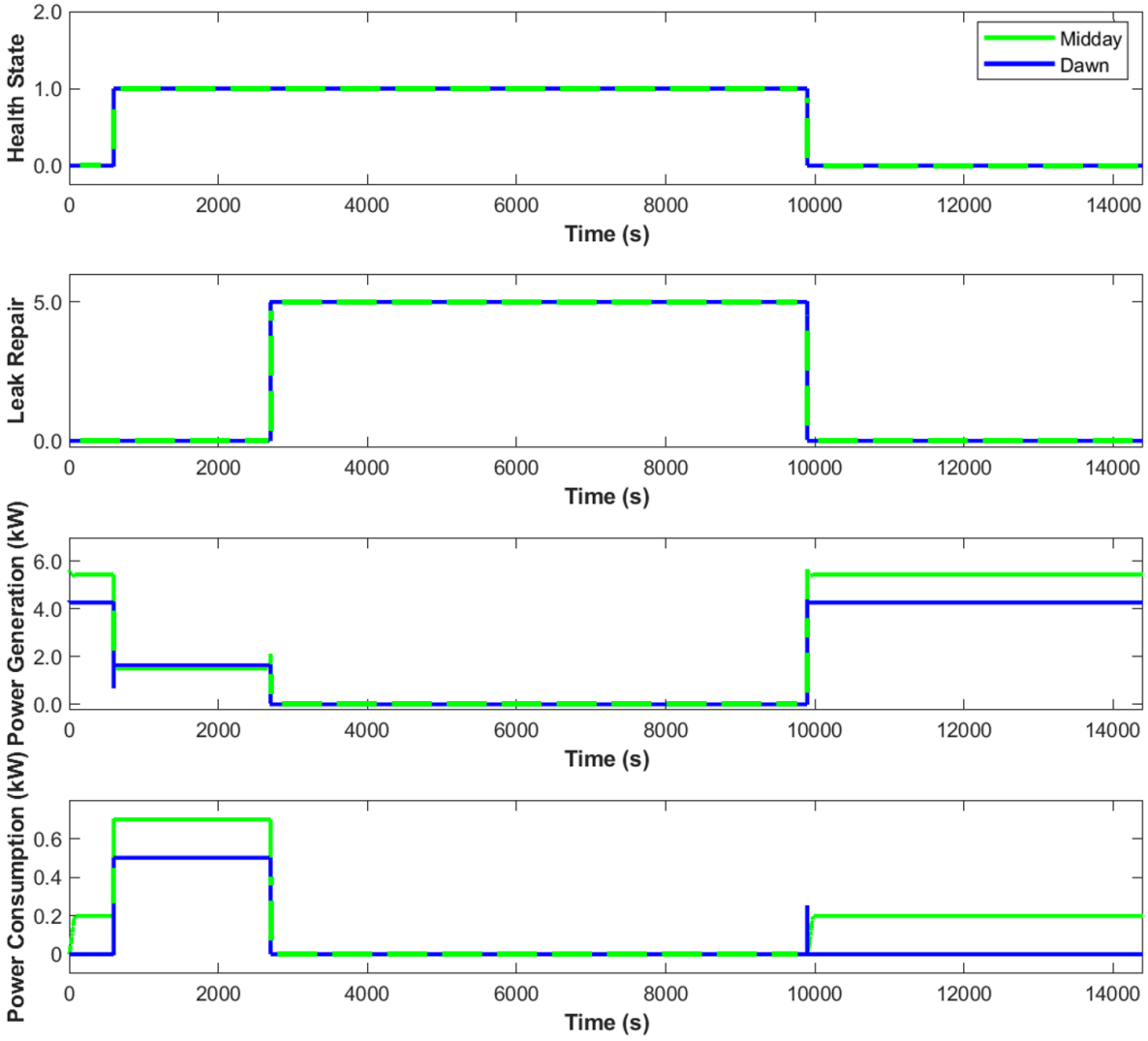


Figure 14: Habitat Response to Intensity Level 3 Coolant Leak at 600s with Human Agent at Varied Time of Day

In Figure 14, the timing for the health state and repair action are the same for these two simulations because the agent is human for both simulations and takes the same amount of time to prepare to exit the habitat and to perform the repair. The bottom two subplots show the effect of the change in time-of-day. Nuclear power generation is slightly lower at dawn than during midday (due to the temperature difference) and zero during the repair action, as we have previously noted. During the heat of midday, the pump consumes more power because the coolant is cycled faster to maintain

the ideal reactor temperature. Pump power consumption is higher at midday than at dawn except for during the repair, when the entire power system is shut down for safety. We also note that there are no bumps in the power generation at dawn like there are at midday, because at dawn the temperature of the reactor does not increase past its ideal temperature.

There is an anomaly in the pump power consumption in the dawn simulation. When the repair finishes, we see an upward spike in pump power consumption. This is another bug that has no physical cause or meaning.

Next, we compare these midday, human-agent simulation to the midday, robot-agent simulation, both with IL 3.

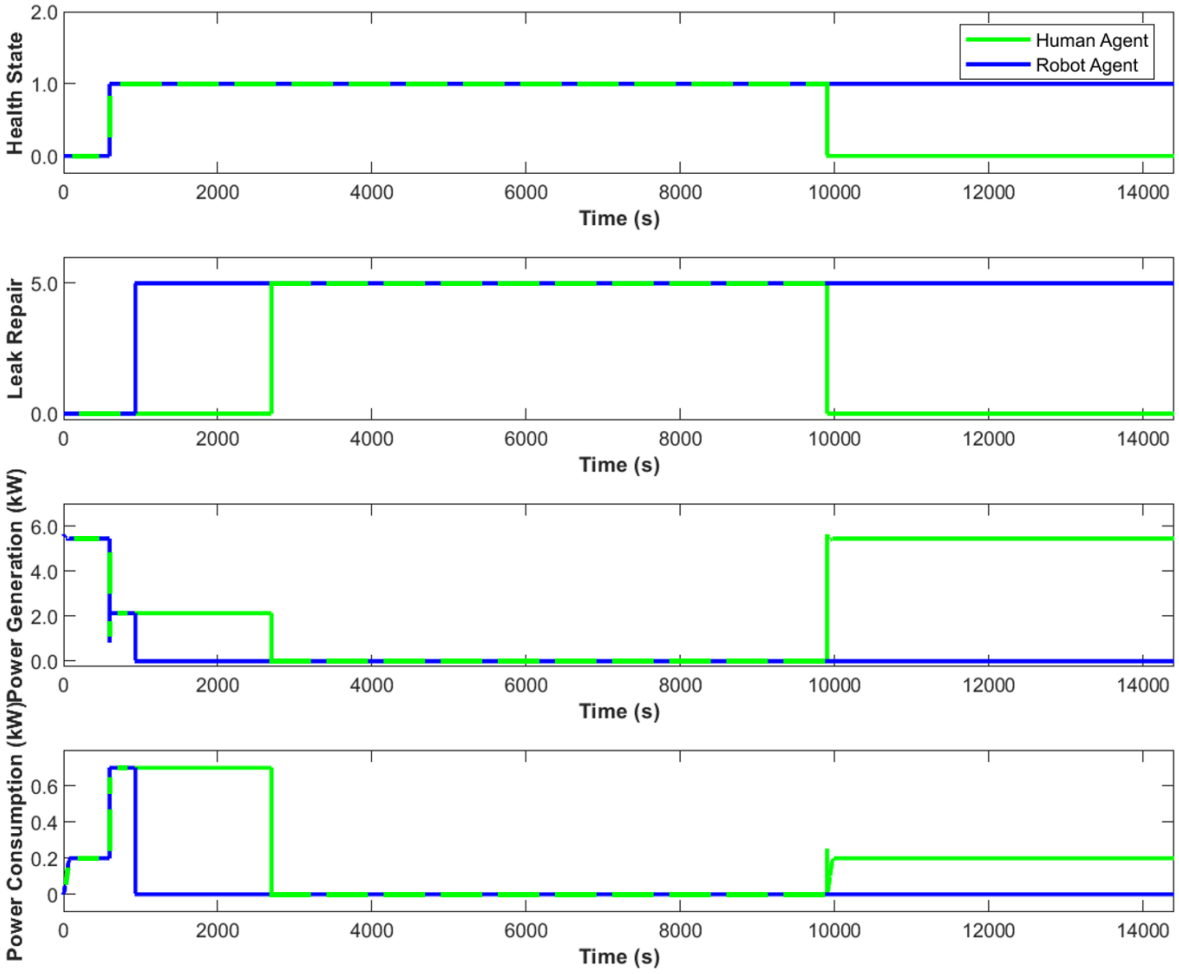


Figure 15: Habitat Response to Intensity Level 3 Coolant Leak at 600s with Varied Agent Type at Midday

The pair of simulations in Figure 15 shows the effect of the type of repair agent. In both situations, we start the disruption at ten minutes. The robot can begin repairing almost immediately, because it does not need to prebreathe before exiting the habitat. Because the repair starts earlier for the robot-agent simulation, the nuclear power generation and pump power consumption both drop to zero earlier than in the human-agent simulation. The robot cannot repair the leak within the four-hour maximum run time of the MCVT, but if the simulation could run for longer, the repair would eventually be successful.

The final comparison we make is between the midday, human-agent simulation and the dawn, robot-agent simulation. Again, we focus on the IL 3 simulation for comparison.

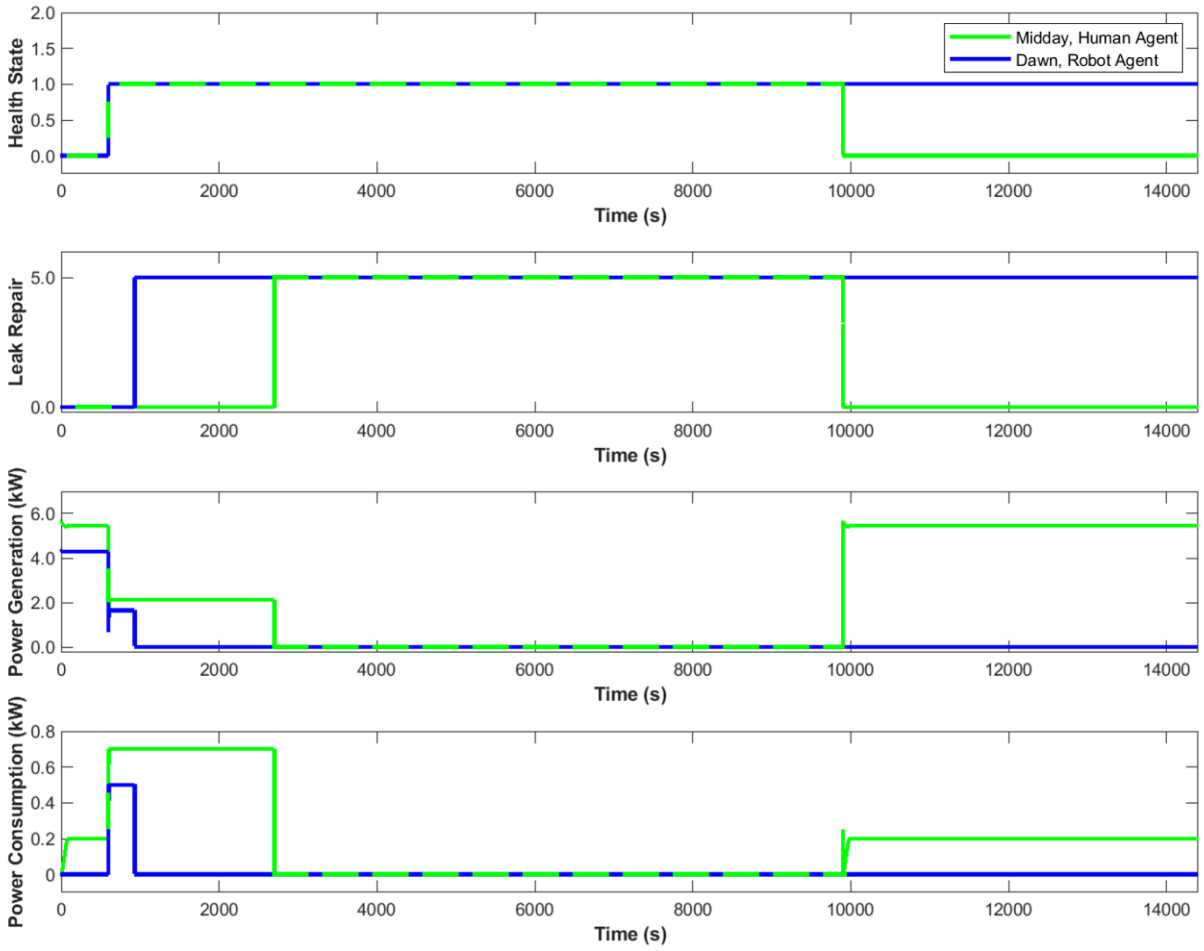


Figure 16: Habitat Response to Intensity Level 3 Coolant Leak at 600s with Varied Agent Type and Varied Time of Day

Again, the robot can begin repairing sooner than the human. There are differences between these two simulations and the previous two in the power generations and pump power consumptions. The differences stem from the robot-agent simulation taking place at night, so we see similar effects to when we compared day to night in Figure14. At night, the power generation is slightly lower and the pump power consumption is higher except during the repair.

Implementation Strategy 2: Injectable Leak Sealant

The second implementation strategy we investigate using the MCVT is the injectable leak sealant. We chose the injectable leak sealant to contrast with the brazing technique because it has a low

probability of competent design, but still a short enough repair time to simulate in real time using the MCVT.

Table 14 outlines the values we input into the MCVT for this simulation set. The Agent Speed and Prep Time are the same as the previous simulation set. The repair times are as defined in Table 12. This implementation strategy takes equally long regardless of the intensity level of the disruption.

Table 14: Injectable Leak Sealant MCVT Inputs

	Human			Robot		
Intensity Level	1	3	5	1	3	5
Agent Speed (m/s)	0.61			0.01		
Prep Time (minutes)	30			0.5		
Repair Time (minutes)	N/A	20		N/A	40	

We input these values into the MCVT and run the same set of simulations as we did with the brazing implementation strategy.

Quantifying Resilience for Comparison of Implementation Strategies

The next step in the design process is evaluating component resilience with each of these implementation strategies to determine which strategy results in the most resilient component. We compare resilience using a version of the R_3 resilience metric introduced in Section 1.2.6. We choose this metric because it is an intuitive way to compare performance when each simulation being investigated has the same exact x-axis, and each performance metric has the same exact y-axis. R_3 employs integrals to quantify the amount of performance lost due to a disruption. In section 1.2.6, we determined that this metric was not viable for all curve shapes because it requires

a value for t_4, which does not exist when there is no recovery. However, we are comparing simulations that are all being run for the same amount of time, meaning that we can use the length of the simulation for t_4 for all no-recovery cases, allowing us to compare them.

In selecting the performance metrics we use to calculate the resilience metrics, we consider which MCVT outputs represent the health of the component in question: the nuclear reactor. Nuclear power generation is an obvious indicator of the nuclear reactor's performance because it is the whole reason the habitat has a nuclear reactor at all. A second performance metric we consider is the power consumption of the coolant pump. The coolant pump works harder when there is less coolant in the system, which increases its power consumption. The coolant pump power consumption is, therefore, a good way to track how well the nuclear reactor system is performing even when there are small changes in output nuclear power. Together, these two performance metrics represent the nuclear reactor's resilience.

We start by calculating the nuclear power generation resilience metric for the human performing the brazing technique. We calculate the metric four times, varying time of day and intensity level. We start with the day, IL 3 simulation. Figure 17 shows the nominal day simulation compared to the IL 3 simulation with a human performing the repair.

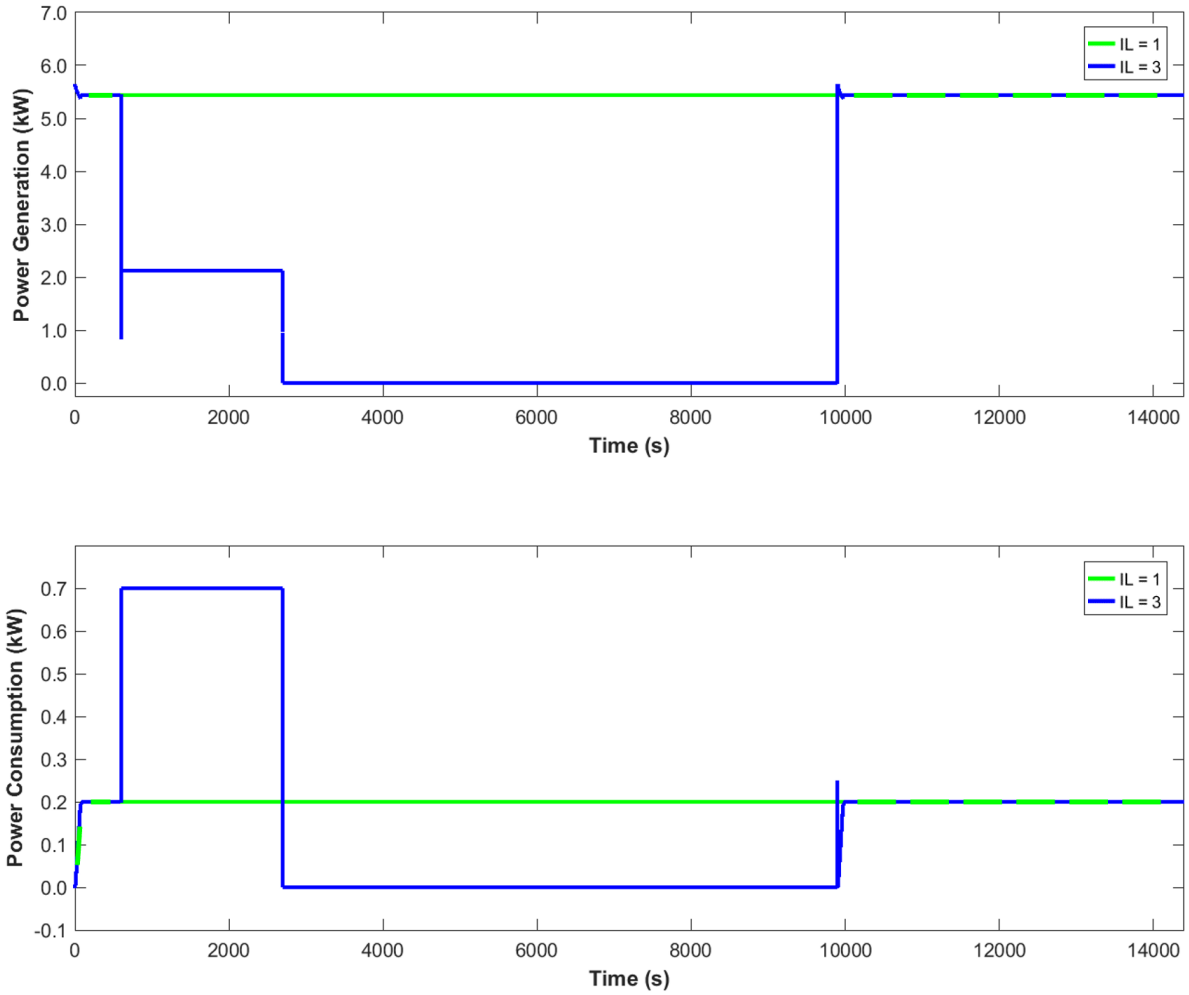


Figure 17: Nominal Habitat vs. Response to Intensity Level 3 Coolant Leak at 600s with Human Agent at Midday

The nuclear power generation plot shows that the disruption reduces the power generation. The resilience metric here is the integral of the nominal performance minus the disrupted performance. The value is the area of the shaded region in Figure 18. The red dashed area is where the disrupted simulation is performing worse than nominal.

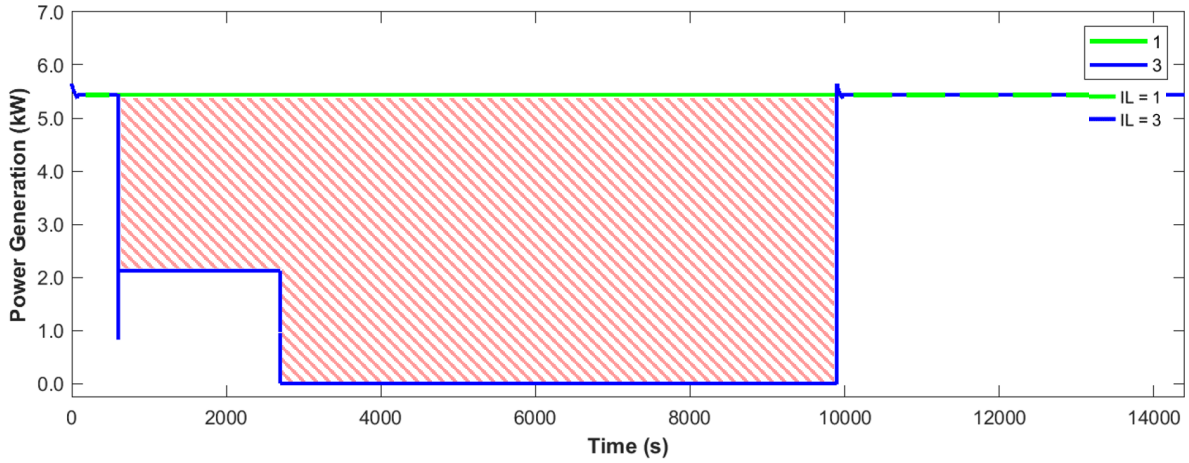


Figure 18: Resilience Metric for Intensity Level 3 Coolant Leak at 600s with Human Agent at Midday

Figure 19 shows how we calculate the pump power consumption resilience metric. With this performance metric, there are segments of the simulation where the pump power consumption higher than nominal, shaded in red. There are also segments where the pump power consumption is lower than nominal, shaded in green. The ideal scenario is the one with the lowest total power consumption, so here the resilience metric is the difference between the two shaded regions.

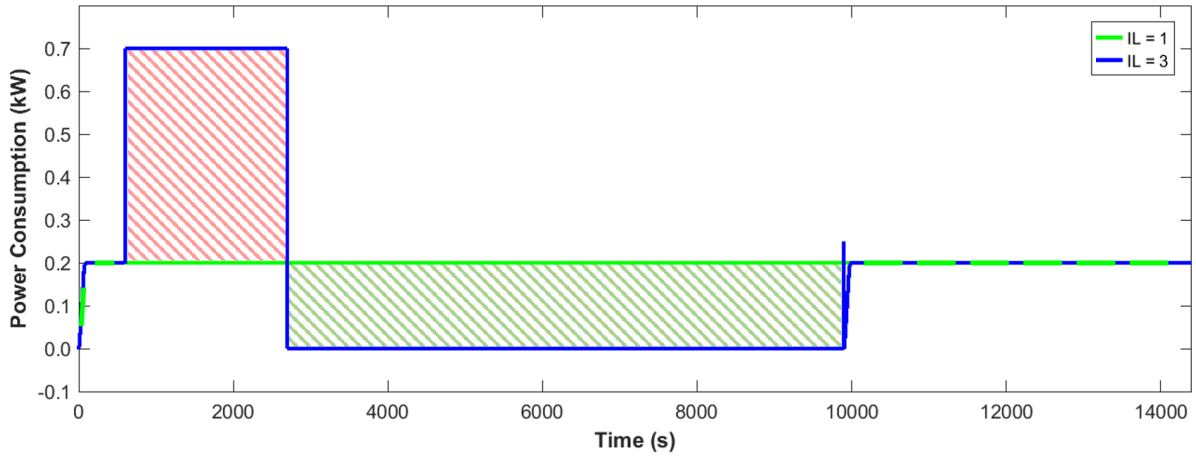


Figure 19: Resilience Metric for Intensity Level 3 Coolant Leak at 600s with Human Agent at Midday

We assign positive values to regions where the disrupted performance is better than nominal (shaded green), and negative values to regions where the disrupted performance is worse (shaded

red). Higher, positive values for each resilience metric represent better nuclear reactor performance.

We perform these calculations for each of the implementation strategies and compile the resulting resilience metrics into Table 15.

Table 15: Component-Level Resilience Metrics

		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)	
		Midday	Dawn	Midday	Dawn
Human Brazing	IL 3	-46,156	-36,327	397	-1,050
	IL 5	-63,671	-50,028	2,348	0
Robot Brazing	IL 3	-74,402	-58,468	2,529	-165
	IL 5	-75,101	-59,008	2,760	0
Human Injectable Leak Sealant	IL 3	-13,504	-10,671	-802	-1,050
	IL 5	-17,957	-14,110	668	0
Robot Injectable Leak Sealant	IL 3	-14,156	-11,113	323	-165
	IL 5	-14,855	-11,673	554	0

Worst			Best

We use colored cells to visualize how each implementation strategy performs based on these resilience metrics. We use the four colors to rank the resilience metrics for each implementation strategy under the same circumstances. The first four values we compare are the daytime, intensity level 3 simulations. Of these four boxes highlighted in red, we shade the most positive value green.

We then shade the next most positive value yellow, then the next orange. The most negative value gets shaded in red.

The values in Table 15 are listed with up to five significant digits, as provided by the MCVT. The MCVT is a simplified model based on numerous assumptions, so we cannot count on its results being accurate to so many significant digits. For this reason, in our analysis, we do not make any decisions based on the actual numerical value of each resilience metric. Instead, we focus on the difference between the values. We round the MCVT output down to essentially zero significant digits by only considering the ranking of implementation strategies, which we track using a five-color system.

In Figures 15 in 16, we see that the robot brazing implementation strategy takes longer than four hours to carry out the repair for both intensity levels, so the damage caused by the disruption is not completely accounted for in these resilience metrics. If we could calculate resilience metrics for the entire repair, the nuclear power generation metrics for the robot brazing technique would all be even more negative, and the pump power consumption metrics would all be even more positive. We will revisit this caveat at the end of the component-level verification step, at the end of Section 2.4.1.

One limitation of this analysis is the fact that it assumes that all four implementation strategies are successful at repairing the leak every time. In reality, according to the assigned control effectiveness values, the brazing technique is more likely to successfully repair the leak than the injectable leak sealant method. We consider the difference in effectiveness in the next section.

Probability of Success

Because the probabilities of competent design of these implementation strategies are both less than one, we investigate the possibility of a failed safety control. What happens if a safety control fails to mitigate a hazardous state? Or if the hazardous state is only fixed for a short time before the problem reemerges?

Figure 20 shows a tree diagram of the possible paths the habitat could take.

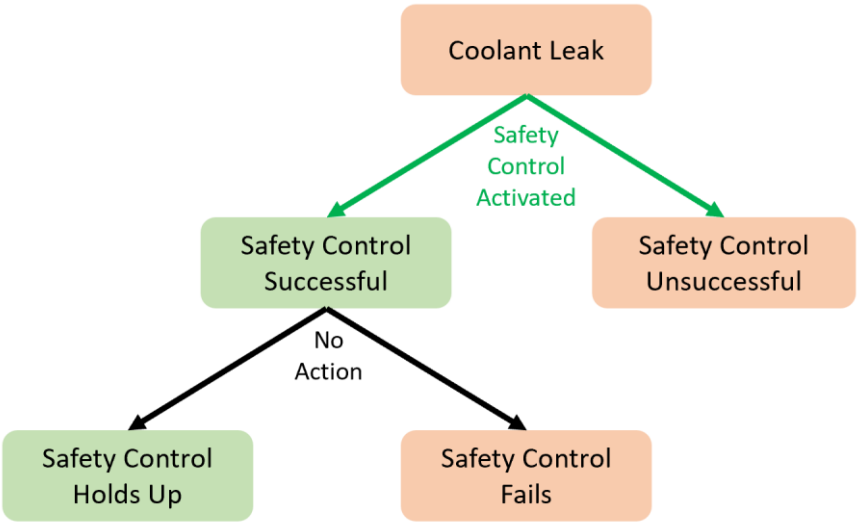


Figure 20: Possible Outcomes of Implementing a Safety Control

The three distinct paths we investigate are: 1. The safety control is successful and holds up indefinitely, 2. The safety control is successful for a period of time before failing, and 3. The safety control is unsuccessful. We assign probabilities to each path using control effectiveness values in Figure 21. The probability that the safety control is successfully implemented is the probability of perfect implementation, representing the capability of the agent. The probability that it holds up after successful implementation is the probability of competent design, representing the effectiveness of the technique.

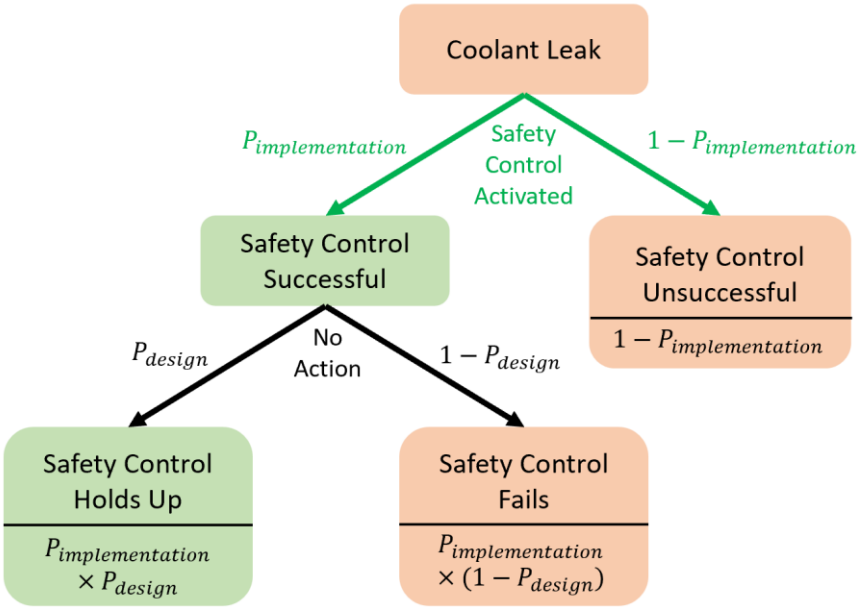


Figure 21: Probabilities of Each Possible Outcome of Implementing a Safety Control

Using the control effectiveness values that we assigned as our probabilities for the different path metrics places a lot of importance on values that are not based on any scientific process, which prompts us to consider how changing a control effectiveness value affects the final ranking. To answer this question, we could perform a sensitivity analysis for the probability of perfect implementation and probability of competent design. We will discuss what such a sensitivity analysis might look like in Section 3.3.

We now consider what each of these paths means for habitat performance using the injectable leak sealant simulations to demonstrate the three different paths.

Figure 22 depicts the first path the habitat can take, where the safety control is implemented effectively and holds up for the duration of the simulation. The three simulations plotted are the nominal case and the intensity levels 3 and 5 cases, all simulated with a human at midday.

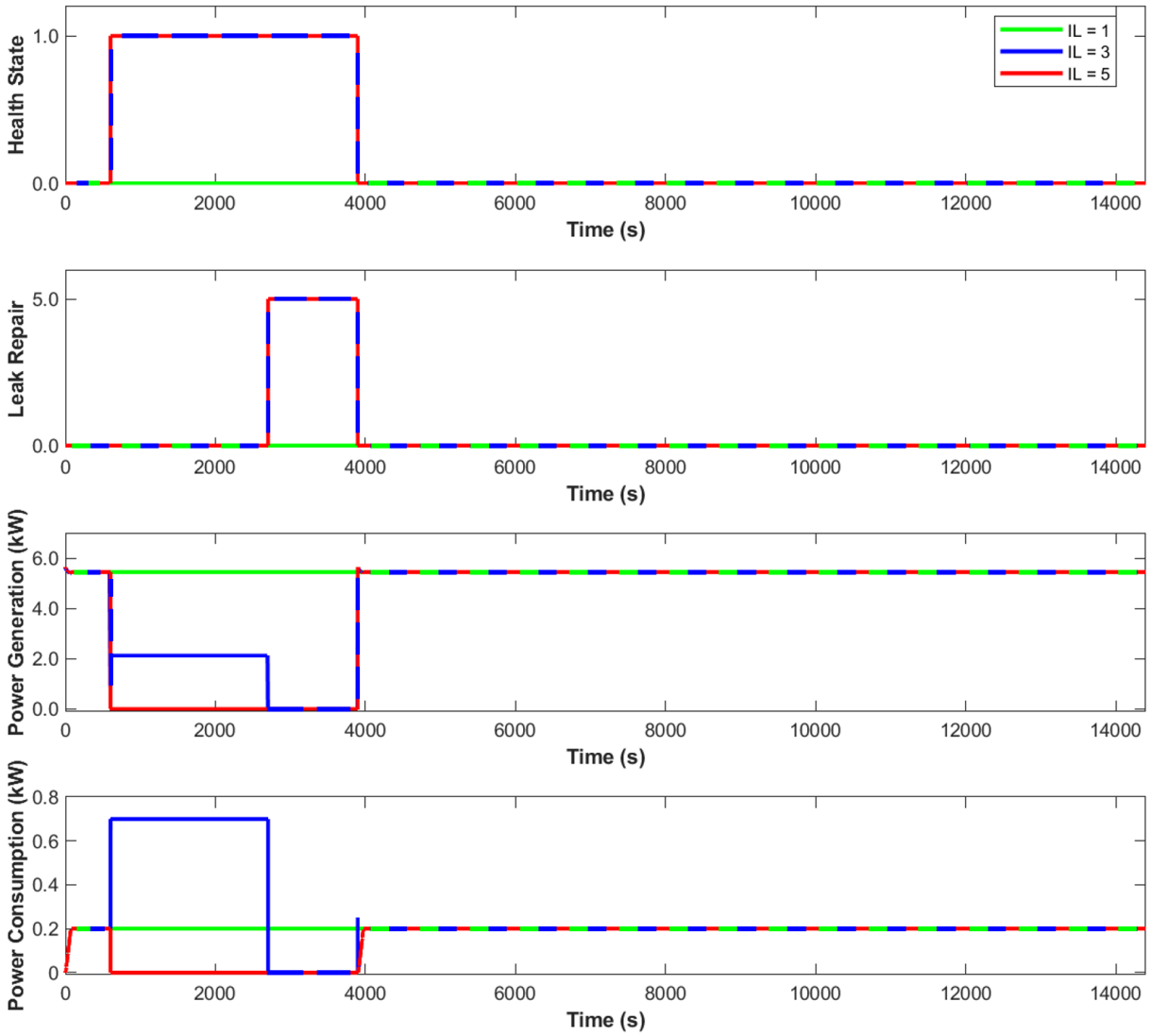


Figure 22: Indefinitely Successful Injectable Leak Sealant Implementation for Intensity Level 3 Coolant Leak at 600s

Figure 23 shows path two, where the initial implementation of the safety control is successful for a period of time before the hazardous state reemerges.

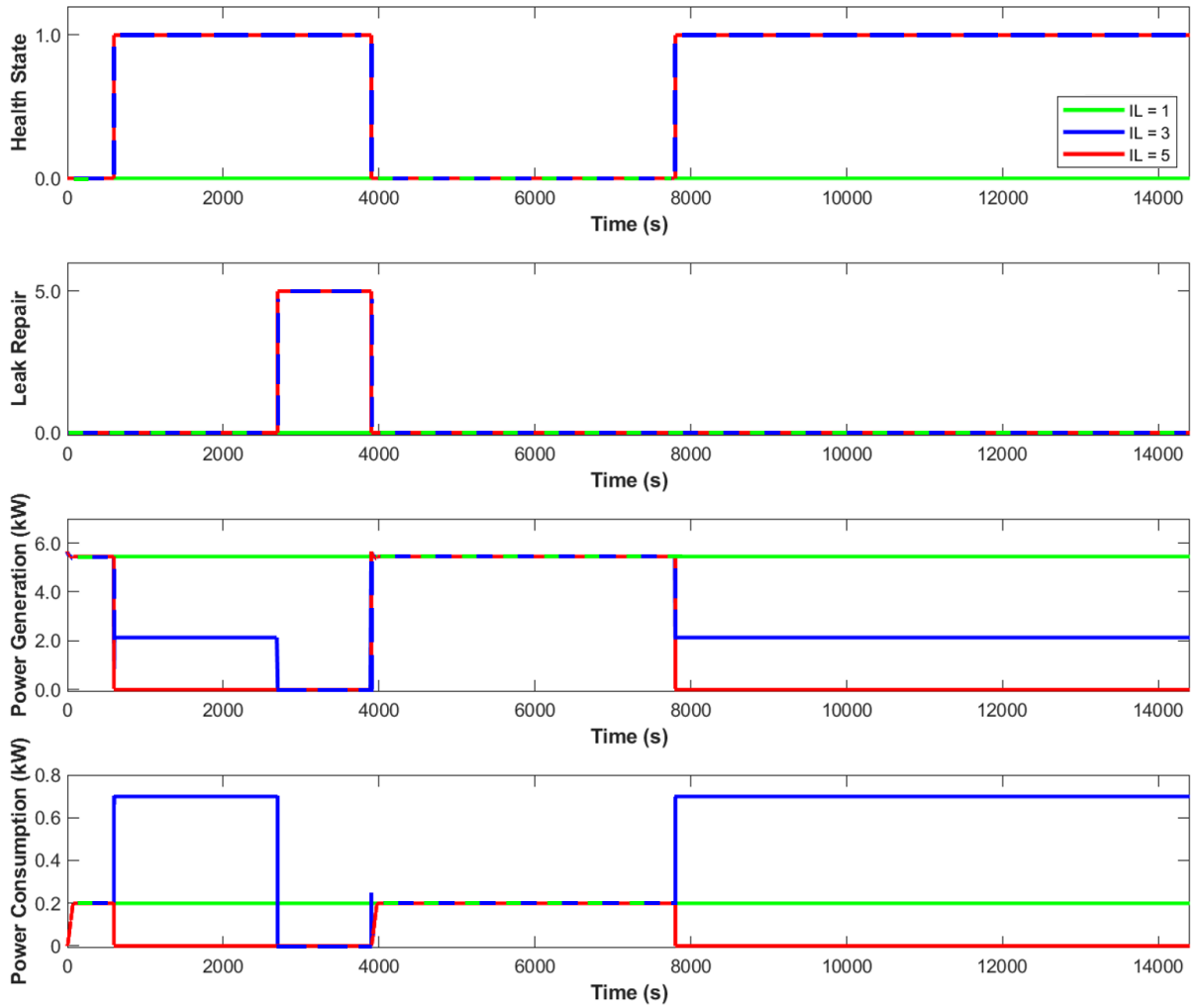


Figure 23: Temporary Successful Injectable Leak Sealant Implementation for Coolant Leak at 600s

Figure 24 shows the third path, where the implementation is immediately unsuccessful and the hazardous state is never resolved.

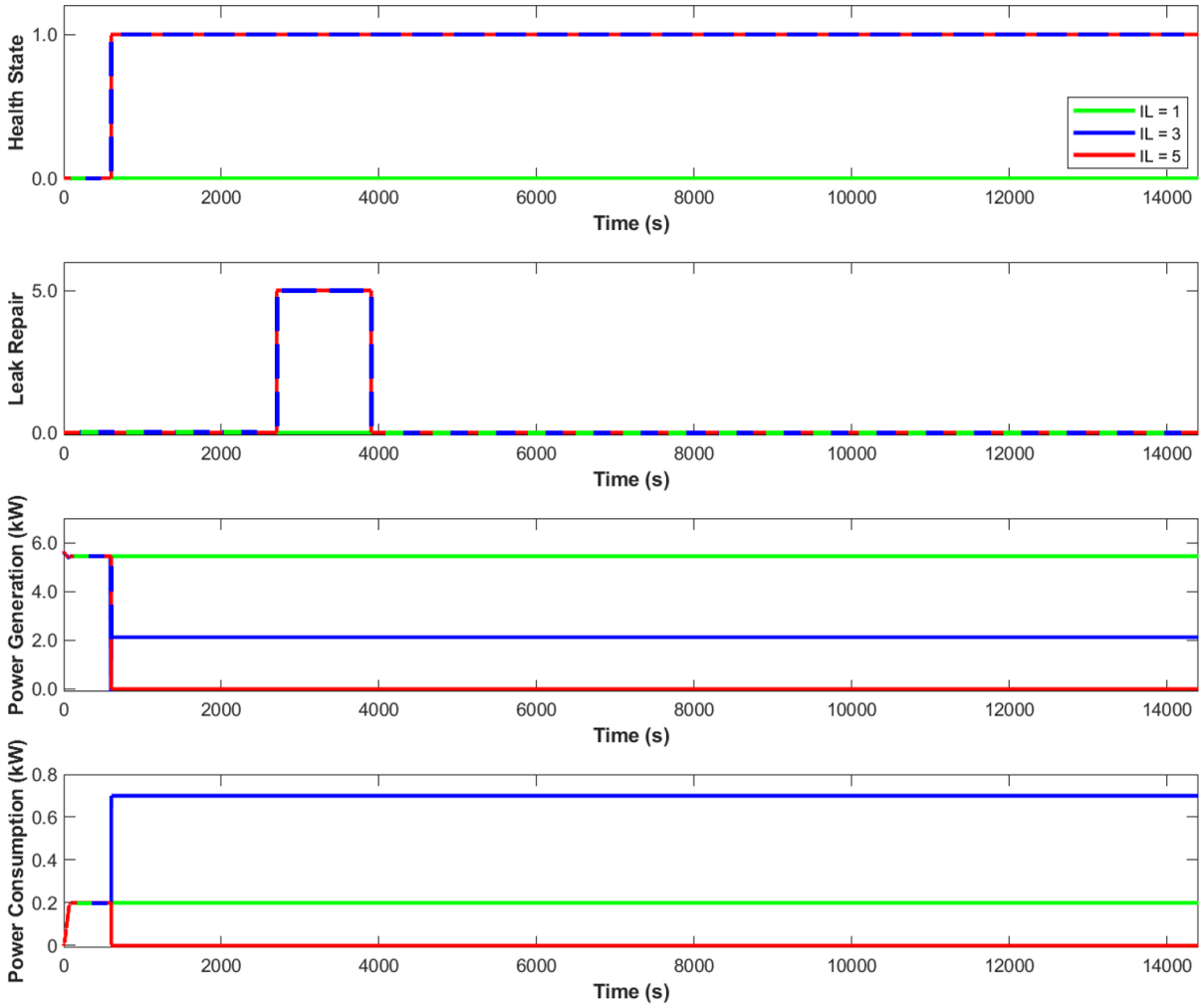


Figure 24: Failed Injected Leak Sealant Implementation for Coolant Leak at 600s

We take the two performance metrics (nuclear power generation and pump power consumption) from these types of simulations and calculate resilience metrics for each one. We calculate the metrics for every path with every intensity level, type of agent, and time of day. Tables 16 and 17 show the metrics from paths two and three.

Table 16: Path 2 Component-Level Resilience Metrics

		Path 2			
		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)	
		Midday	Dawn	Midday	Dawn
Human Brazing	IL 3	-50,806	-40,020	-302	-1,750
	IL 5	-71,290	-56,014	2,628	0
Robot Brazing	IL 3	-74,402	-58,468	2,529	-165
	IL 5	-75,101	-59,008	2,760	0
Human Injectable Leak Sealant	IL 3	-35,422	-28,082	-4,102	-4,350
	IL 5	-53,875	-42,331	1,988	0
Robot Injectable Leak Sealant	IL 3	-36,075	-28,544	-2,977	-3,465
	IL 5	-50,773	-39,894	1,874	0

Worst			Best

For path 2, the dawn, IL 5 simulations all have zero for the pump power consumption metric, regardless of implementation strategy because the nominal pump power consumption at dawn is zero kilowatts, and the intensity level 5 disruption shuts down the cooling system completely.

Table 17: Path 3 Component-Level Resilience Metrics

		Path 3			
		Nuclear Power Generation		Coolant Pump Power Consumption	
		Midday	Dawn	Midday	Dawn
All Implementation Strategies	IL 3	-45,831	-36,405	-6,900	-6,900
	IL 5	-75,101	-59,008	2,760	0

For path 3, all the resilience metrics are identical across the different implementation strategies, because there is no successful repair at all. The only factors that affect nuclear power generation and pump power consumption when there is no repair action are time of day and intensity level.

We take the resilience metrics calculated for each path and combine them according to the weights outlined in Figure 21. Using the weights derived from control effectiveness values allows us to account for two of the control effectiveness values, probability of perfect implementation and probability of competent design. A third value, response margin, is accounted for in the repair time input. The final metrics below give us a well-rounded idea of how these implementation strategies will perform.

Table 18: Final Component-Level Resilience Metrics

		Final Metrics			
		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)	
		Midday	Dawn	Midday	Dawn
Human Brazing	IL 3	-46,459	-36,735	-3,321	-4,045
	IL 5	-70,148	-55,117	2,582	0
Robot Brazing	IL 3	-54,402	-43,024	-4,071	-4,880
	IL 5	-75,101	-59,008	2,760	0
Human Inj. Leak Sealant	IL 3	-28,572	-22,646	-3,194	-3,417
	IL 5	-49,532	-38,919	1,828	0
Robot Inj. Leak Sealant	IL 3	-32,864	-26,022	-3,230	-3,572
	IL 5	-53,043	-41,677	1,955	0

We consolidate the metric rankings for each path and for the final metrics into Table 19.

Table 19: Component-Level Resilience Metric Rankings for each Path

		Path 1				Path 2				Path 3				Final Metrics			
		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)		Nuclear Power Generation (kJ)		Coolant Pump Power Consumption (kJ)	
		Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn
Human Brazing	IL 3																
	IL 5																
Robot Brazing	IL 3																
	IL 5																
Human Inj. Leak Sealant	IL 3																
	IL 5																
Robot Inj. Leak Sealant	IL 3																
	IL 5																

Table 19 shows how each implementation strategy performs under each circumstance and path. Looking at how the colors change from left to right, we see that there are some consistent patterns and some that change.

In the first two paths, the brazing technique performed by human and robot consistently has the lowest nuclear power generation resilience metric. At the same time, brazing has some of the best coolant pump consumption metrics for path one and two, especially at midday. The injectable leak sealant technique tends to perform better with a human at intensity level 3 and a robot at intensity level 5 in terms of the power generation metric, and the opposite in terms of pump power consumption.

Adding in path three, which has the same metrics for every implementation strategy, helped clarify which implementation strategy is most likely to perform the best: an injectable leak sealant applied by a human, which has the highest resilience metric value in every case except for the pump power consumption with an IL 5 leak at midday. Even with a significantly worse probability of competent design, the injectable leak sealant outperforms the brazing technique by having a higher probability of perfect implementation and a better response margin, highlighting the importance of every facet of control effectiveness.

In the actual design process, we would now perform a similar analysis of each other component in the habitat, using relevant disruptions and resilience metrics to select the best implementation strategies for each one.

For this thesis, we select the two best-performing implementation strategies to compare at the subsystem level: human agent injectable leak sealant and robot agent injectable leak sealant. In general, we can choose to analyze as many implementation strategies as we want at the subsystem level. If there are three implementation strategies at the component level that are all performing about equally, all three can be compared at the subsystem level. If there is only one standout implementation strategy, we do not have to do any subsystem-level comparison for that safety control.

The fact that the repair does not finish for the robot brazing implementation strategy does not affect these results. If we had been able to calculate the metrics for the whole repair, the nuclear power generation values would have been even worse. The pump power consumption values would have been better, but that improvement would be small in magnitude compared to the loss in terms of power generation. Overall, the robot brazing strategy would still have had the worst performance for all four power generation metrics, but it might have been able to claim the top spots for three of the four pump power consumption metrics. This would have prompted us to consider whether we value maximizing power generation or minimizing power consumption.

We would ultimately have decided to value maximizing power generation, as the power generation is what actually supports the habitat. After making that decision, we would have ended up with the same top two implementation strategies.

2.4.2 Subsystem-Level Verification

We now move on to the subsystem-level verification step of the design process. Our goal with this step of the design process is to select between the implementation strategies we selected for individual components at the component level by comparing how they perform at the subsystem level.

For this thesis, we focus on the subsystem most affected by the CSC: the power subsystem. First, we carry out all the same simulations we ran during our component-level analysis, tracking different output variables to understand how the entire power subsystem is performing instead of just the nuclear reactor.

Figure 25 shows the subsystem-specific output from the nominal simulations at midday and dawn to give a baseline.

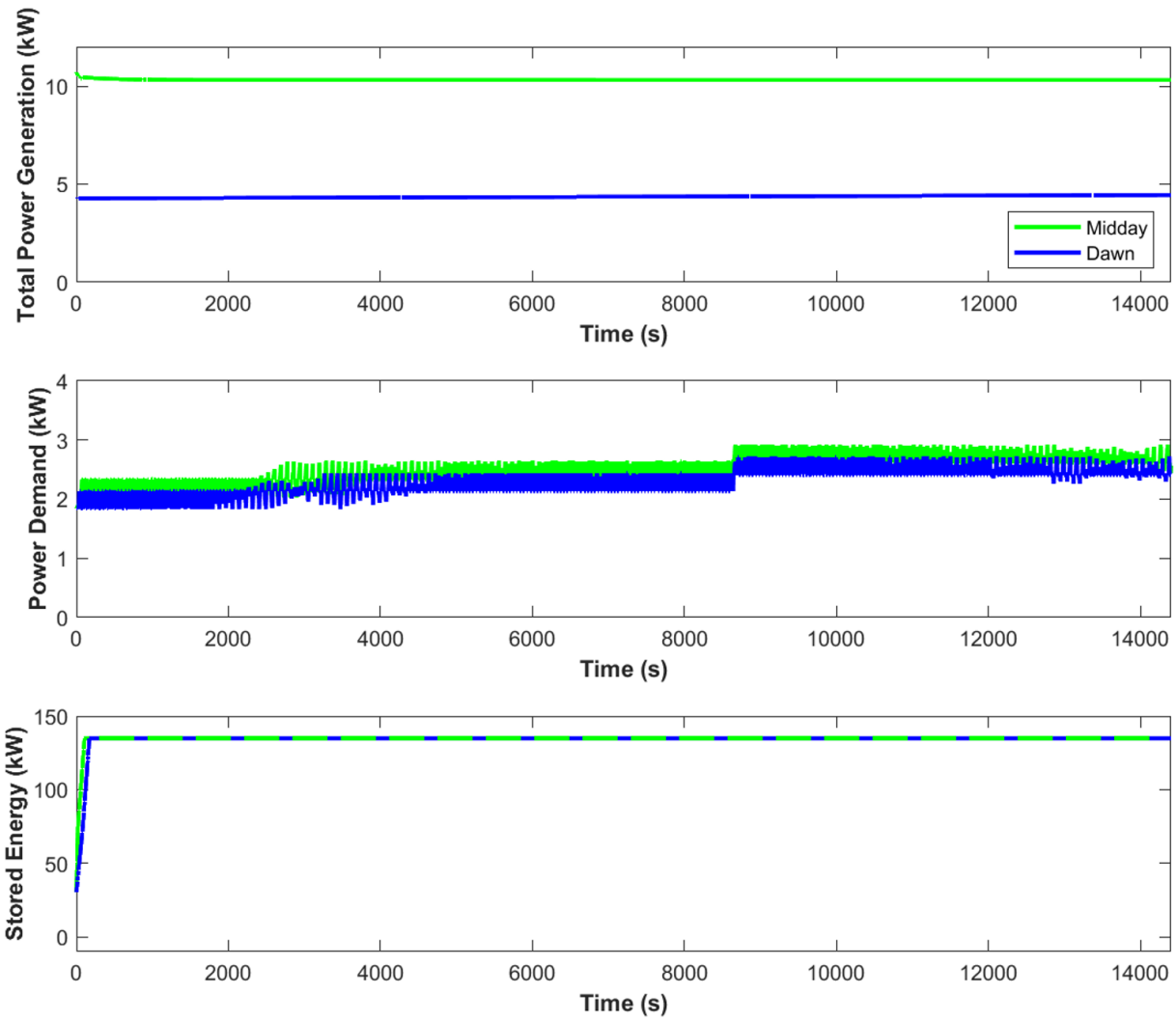


Figure 25: Nominal Habitat Output at Midday and Dawn

The first subplot in Figure 25 shows the total generation of the power subsystem. The total power generation is the sum of the nuclear and solar power generations. At dawn, there is almost zero solar power generation because the sun is shining almost perpendicular to the solar PV array. The result is that the midday power generation is more than double the dawn value. This subplot shows the effect of the moving sun. When the simulation starts at dawn, the sun is slowly moving to a better angle for solar power generation. The dawn total power generation increases slightly over the four-hour period. At midday, we see the opposite. The midday simulation starts when the sun is at the ideal angle for solar power generation, but slowly moves to a less ideal angle. The midday total power generation decreases slightly over four hours.

The second subplot shows the habitat's power demand. The power demand is comprised of the power needed by life support (ECLSS), habitat monitoring (sensors, FDD), housekeeping and science, and the coolant pump. Figure 26 shows how each demand contributes to the total. The housekeeping and science load is modeled to reflect how astronauts' usage of the habitat might influence power demand on a 24-hour cycle. Figure 27 shows how the housekeeping and science load changes over the length of an Earth day using time scaling. The 32-second simulation shows the load's behavior over a 24-hour period.

The coolant pump load is constant in the absence of a disruption, and the monitoring load is always constant. The ECLSS load fluctuates as the life support system regularly activates to reach temperature and pressure set points and turns off once they are reached.

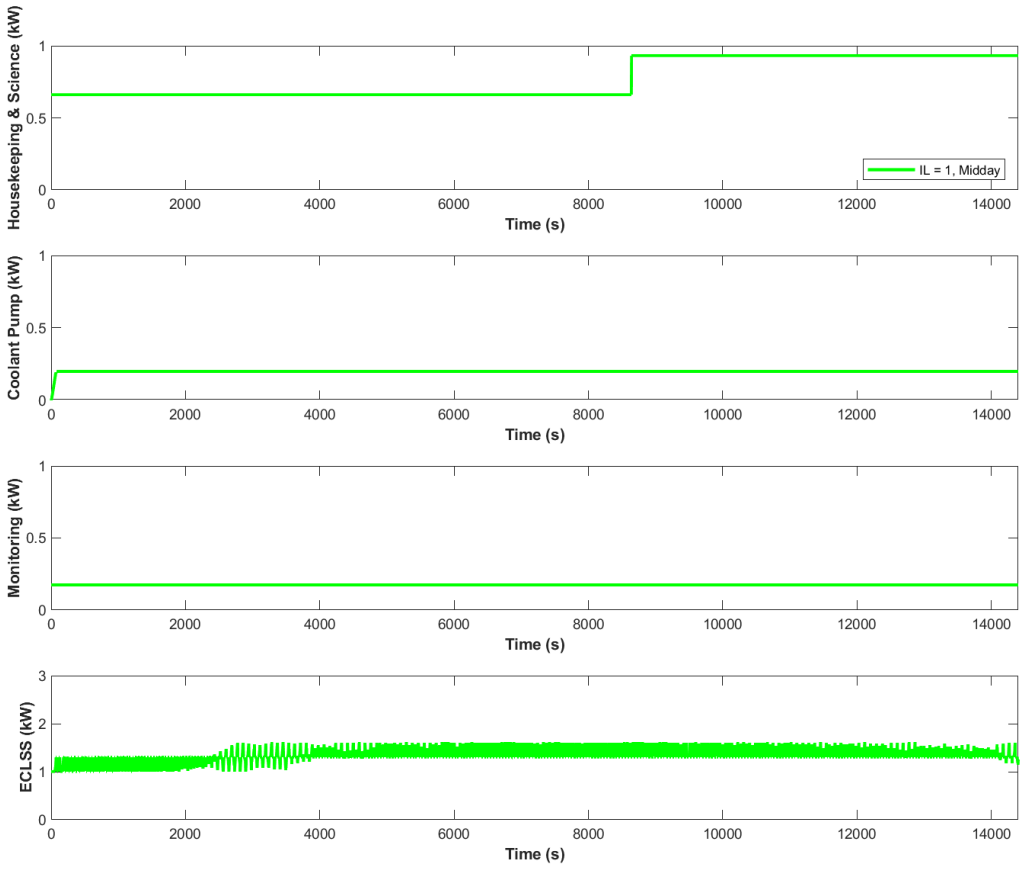


Figure 26: Habitat Power Demand Attributes

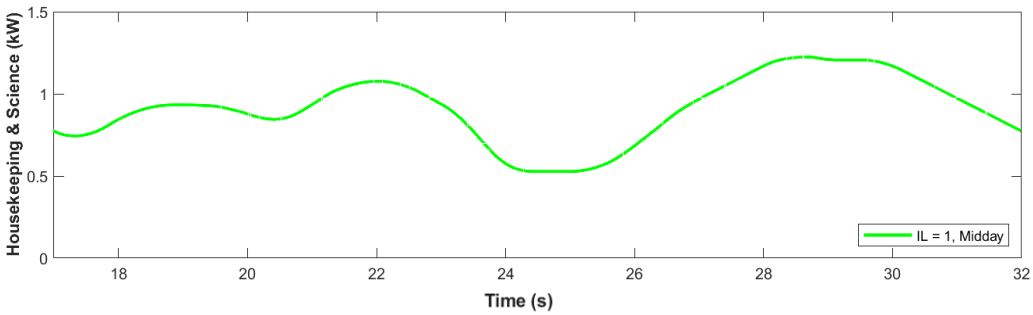


Figure 27: Housekeeping Power Demand over 24 Hours (time-scaled)

The final subplot of Figure 25 tracks the amount of energy stored in the habitat's batteries. The energy storage is constant unless there is a disruption that causes diminished power generation, necessitating the use of stored energy. In the subplot, the stored energy is about 40 kW at the start of the simulation due to the MCVT's initial conditions. The habitat is generating more power than is being demanded at dawn and at midday, so the energy storage quickly fills to its maximum level. Looking closely, energy storage reaches its maximum value quicker at midday than at dawn due to the larger difference in power generation and power demand at that time of day.

Now consider what happens to each of these outputs when we introduce the coolant leak disruption. Figure 28 compares the habitat reaction to the leak with intensity levels 3 and 5 at midday with the implementation strategy of a human using an injectable leak sealant.

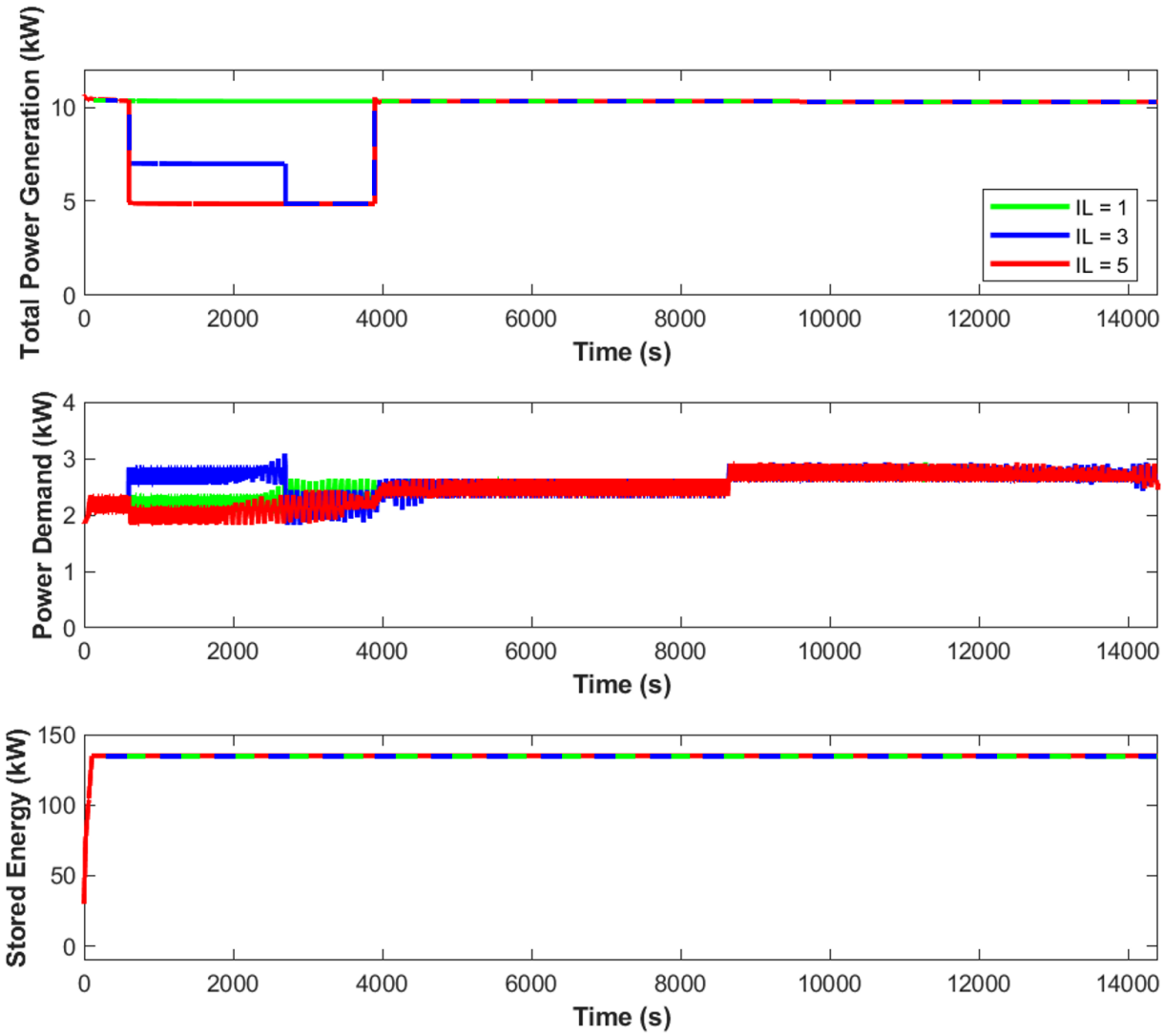


Figure 28: Power System Response to Intensity Level 3, 5 Coolant Leak at 600s at Midday with Human Agent

The habitat power generation decreases when the coolant leak disrupts nuclear power generation starting at 10 minutes (600 seconds). When the intensity level is 3, the power generation initially decreases less than when the intensity level is 5 due to the difference in severity of the leaks. When the repair action starts ~30 minutes later (~2,700 seconds), the nuclear reactor is disconnected from the habitat for safety. The nuclear power generation is then zero for both intensity level 3 and 5 until the leak sealant is injected and takes effect around 65 minutes (3,900 seconds), returning power generation to nominal.

The increase and then decrease in power demand at intensity level 3 and the decrease for intensity level 5 are caused by the change in power demand of the coolant pump, as discussed in detail in Section 2.4.1.

At midday, the solar power generation is large enough to handle the entire habitat load in the absence of nuclear power, so we see no change in the habitat's stored energy.

Figure 29 the habitat response to the same intensity level 3 coolant leak at 600s at midday compared to at dawn.

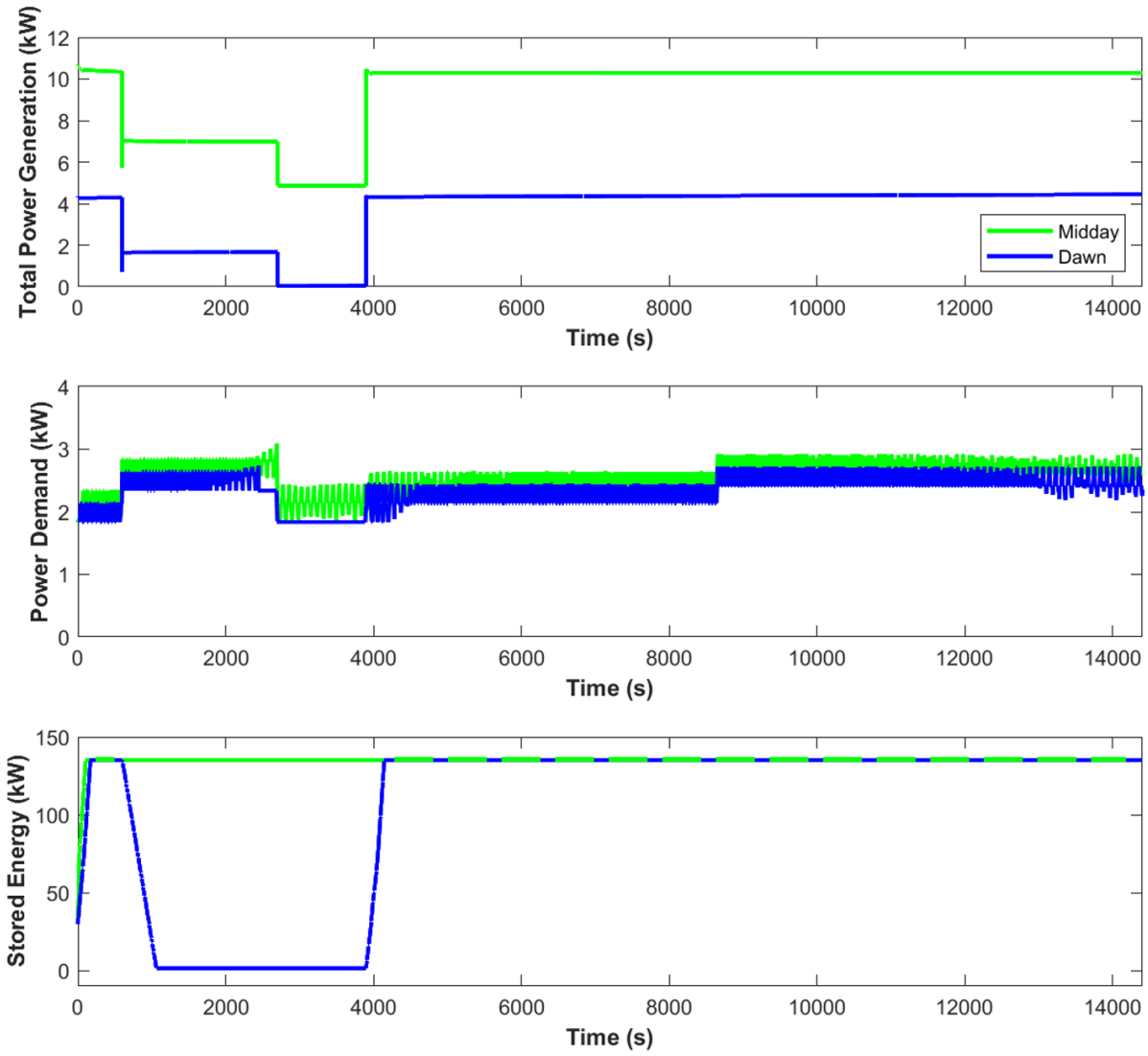


Figure 29: Power System Response to Intensity Level 3 Coolant Leak at 600s at Midday and Dawn with Human Agent

Changing the time of day of the leak affects the nuclear power generation the same way it did in Section 2.2.3.1. When the leak happens at dawn, the solar power generation is not enough to meet the habitat demand, and for the first time we see a decrease in the stored energy. In less than eight minutes, the energy storage goes from full to empty. When the leak is repaired and nuclear power generation returns to full force, it takes about 4 minutes to recuperate the lost energy storage. In the next section we discuss the habitat-wide effects of this disruption and the ramifications of running out of battery power.

Figure 30 compares the same intensity level leak at the same time of day when using a human or robot.

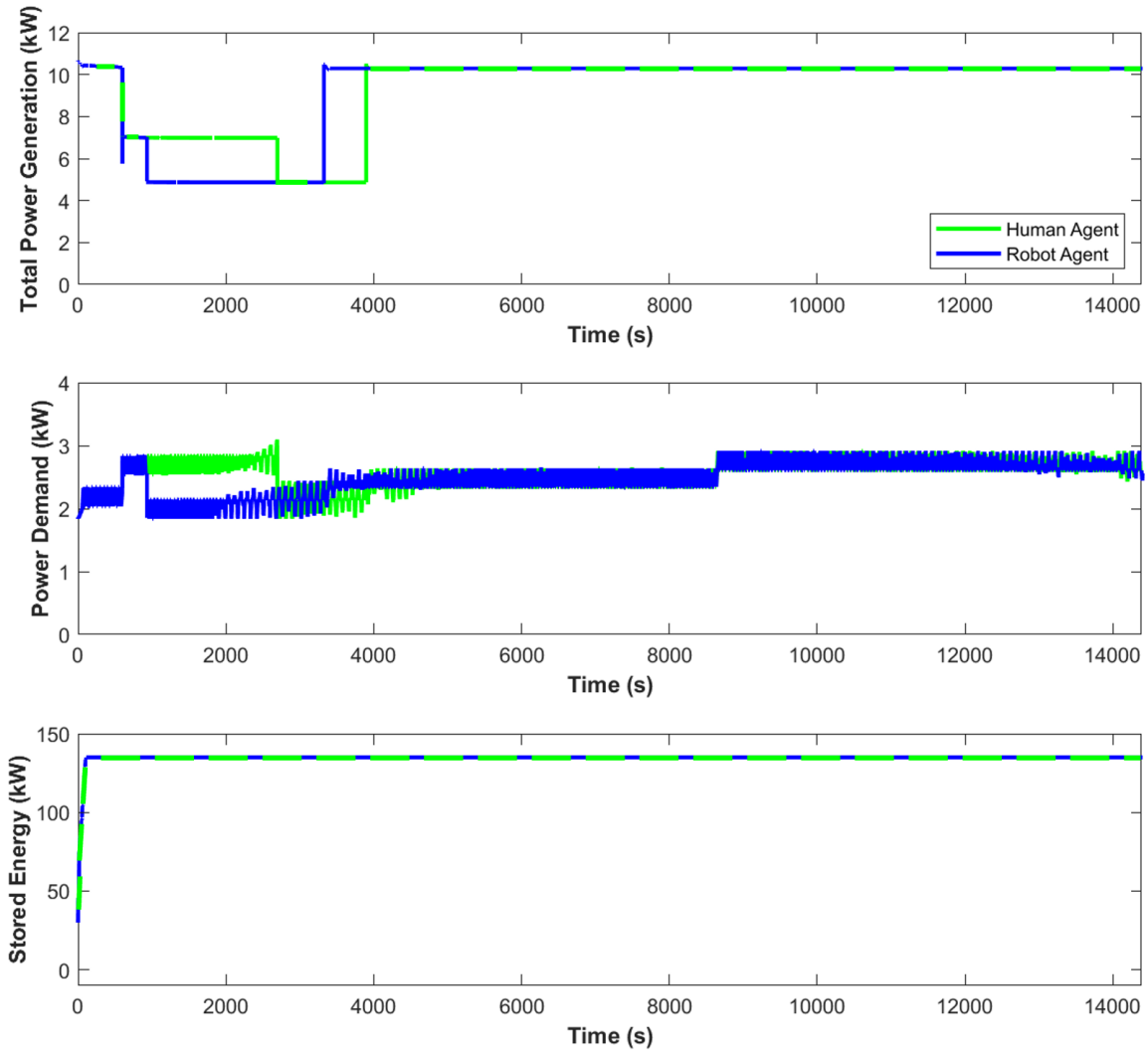


Figure 30: Power System Response to Intensity Level 3 Coolant Leak at 600s at Midday with Human and Robot Agent

The difference between these two responses to the leak stem from how long it takes the agent to prepare for and perform the repair. The human must spend 30 minutes prebreathing but takes only 20 minutes to inject the leak sealant. The robot only takes 30 seconds to prepare for the repair, but then takes 40 minutes to inject the sealant. The human also takes less time to travel to the location of the repair. The need for prebreathing results in the human ultimately taking longer to perform the leak repair than the robot.

Now, as we did at the component level, we choose the performance metrics we want to use in our resilience metric calculations. Two of the three functions we listed in the power system FHA are *Generate Power* and *Store Energy*. We track how well the subsystem is generating power by comparing the total power generation during a disruption to the nominal power generation. We measure how well the subsystem is storing excess energy by keeping track of the amount of stored energy in the batteries.

The third function in the FHA is *Convert Power*, which we do not track as a performance metric for two reasons. First, the CSC does not affect the power converters, so we would see no change in their performance during the disruption. Second, the MCVT does not output a metric that would sum up the converters' performance.

Figure 31 shows the performance metrics for the intensity level 3 coolant leak at dawn with a human performing the repair. The shaded areas represent the resilience metrics.

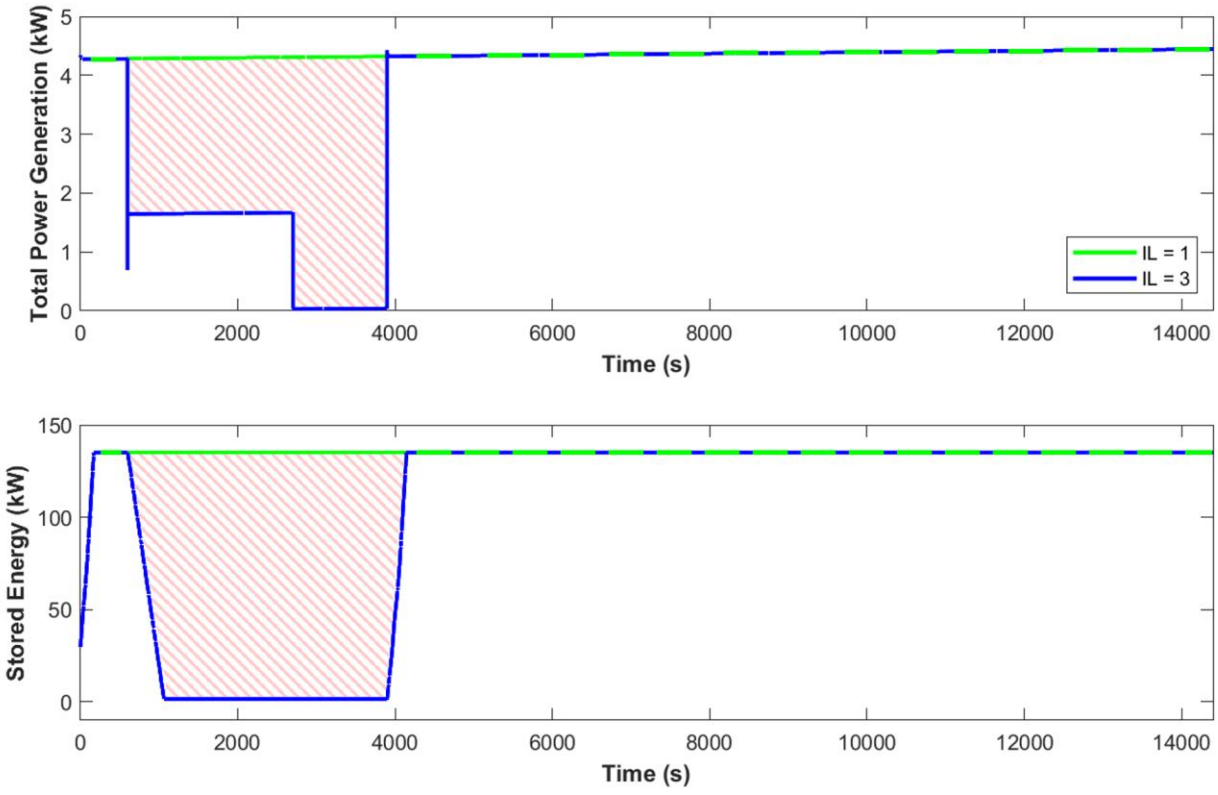


Figure 31: Power System Performance Metrics for Intensity Level 3 Coolant Leak at 600s at Dawn with Human Agent

We calculate the same two resilience metrics for every combination of time of day, type of agent, and intensity level and compile the results in Table 20

Table 20: Path 1 Subsystem-Level Resilience Metrics

		Path 1 Metrics			
		Total Power Generation (kJ)		Stored Energy (kJ)	
		Midday	Dawn	Midday	Dawn
Human Injectable Leak Sealant	IL 3	-13,504	-10,671	0	-427,690
	IL 5	-17,957	-14,110	0	-445,800
Robot Injectable Leak Sealant	IL 3	-14,156	-11,133	0	-352,930
	IL 5	-14,855	-11,673	0	-369,520

We also calculate the same metrics for the other paths the habitat can take, and compile the results into Tables 21, 22, and 23.

Table 21: Path 2 Subsystem-Level Resilience Metrics

		Path 2 Metrics			
		Total Power Generation (kJ)		Stored Energy (kJ)	
		Midday	Dawn	Midday	Dawn
Human Injectable Leak Sealant	IL 3	-26,975	-21,904	0	-855,380
	IL 5	-35,882	-28,782	0	-891,600
Robot Injectable Leak Sealant	IL 3	-28,280	-22,828	0	-705,860
	IL 5	-29,678	-23,908	0	-739,030

Table 22: Path 3 Subsystem-Level Resilience Metrics

		Path 3 Metrics			
		Total Power Generation (kJ)		Stored Energy (kJ)	
		Midday	Dawn	Midday	Dawn
All Implementation Strategies	IL 3	-45,831	-36,405	0	-1,811,300
	IL 5	-75,101	-59,008	0	-949,020

Table 23: Final Subsystem-Level Resilience Metrics

		Final Metrics			
		Total Power Generation (kJ)		Stored Energy (kJ)	
		Midday	Dawn	Midday	Dawn
Human Injectable Leak Sealant	IL 3	-24,011	-19,310	0	-797,004
	IL 5	-36,577	-29164	0	-817,098
Robot Injectable Leak Sealant	IL 3	-29,591	-23,627	0	-938,672
	IL 5	-41,230	-32,725	0	-750,296

Once again, we use Table 24 as an overview of the resilience metric rankings and how they change when we account for the different paths the habitat can take. Paths 1 and 2 have identical coloring with each implementation strategy taking the top spot in half of the metrics calculated, meaning that each strategy is equally likely to perform best. However, once we incorporate the possibility of the repair completely failing, the human wins out over the robot.

Table 24: Power Subsystem Resilience Metric Rankings for Each Path

		Path 1				Path 2				Path 3				Final Metrics			
		Total Power Generation (kJ)		Stored Energy (kJ)		Total Power Generation (kJ)		Stored Energy (kJ)		Total Power Generation (kJ)		Stored Energy (kJ)		Total Power Generation (kJ)		Stored Energy (kJ)	
		Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn	Midday	Dawn
Human Inj. Leak Sealant	IL 3																
	IL 5																
Robot Inj. Leak Sealant	IL 3																
	IL 5																

The way the colors are distributed for the first two paths is interesting. Why does the human perform better for intensity level 3 while the robot performs better for intensity level 5? We take a closer look at why the intensity level of the disruption changes the ideal implementation strategy by plotting the simulations in question. Figure 32 shows the power generation performance metric of the intensity level 3 and 5 coolant leaks at dawn with each type of agent.

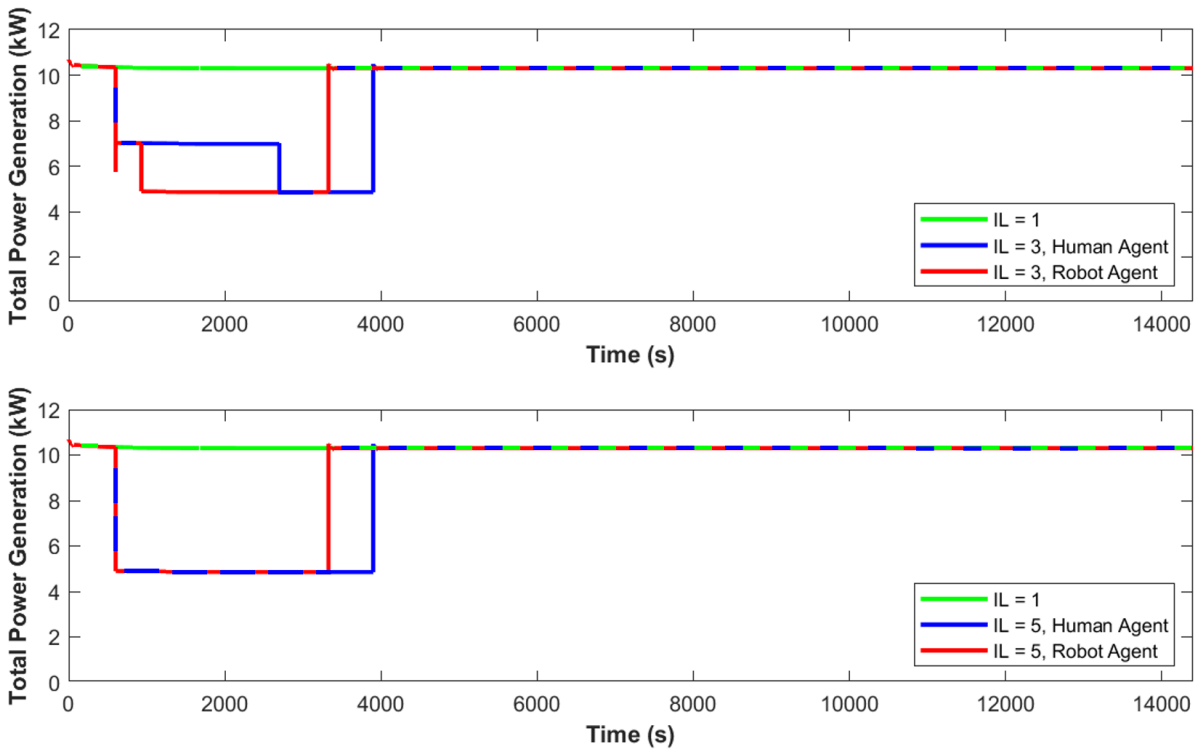


Figure 32: Power Generation for Intensity Level 3, 5 Coolant Leak at 600s at Dawn with Human and Robot Agents

The bottom subplot shows the power generation's response to the intensity level 5 disruptions. The power generation immediately drops to around 5 kW at the time of the disruption for both simulations. The human agent simulation in the blue line shows that it takes the human about 55 minutes to prebreathe, walk to the location of the repair, and inject the leak sealant. The robot does not have to prebreathe and only takes about 45 minutes to inject the leak sealant. The habitat loses less power generation when the agent is a robot for an intensity level 5 coolant leak.

In the top subplot, the blue line shows the power generation as the human prebreathes, walks to the repair site, and then performs the repair. In the red line, we see that the robot takes a much shorter amount of time to prepare for the repair and navigate to the location of the repair. Unlike the intensity level 5 leaks, the power generation is not the same during the preparation/navigation steps as during the actual repair. The human spends more time on the repair process, but much of that time is spent while the power generation is at a higher level than the during-repair level of about 5 kW. The robot spends twice as much time at that during-repair level as the human, resulting in more power generation lost.

In the full habitat design process, we would now perform this same analysis for different safety controls in the power subsystem while holding this safety control constant at the best-performing implementation strategy, human injectable leak sealant. We would cycle through each component in the power system, disrupting its performance, and comparing implementation strategies for fixing it. Eventually, we would have the best-performing implementation strategy implemented for every component in the power subsystem.

Then, of course, we would move on to perform the same process for the rest of the subsystems in the habitat. Eventually, we would have implementation strategies chosen for every component in the habitat. At that point, we would move on to habitat-level verification.

2.4.3 Habitat-Level Verification

Our last step is to perform the same analysis at the habitat level. Once again, we revisit our FHA when deciding which performance metrics to analyze. The two habitat-level functions we listed are *Provide Power to Support Habitat Loads* and *Maintain a Livable Atmosphere.*

To track the first function, we compare the habitat power demand to the power that is supplied to meet that demand. The performance metric is the power supplied minus the power demanded. A value of zero represents all loads being met. Any value less than zero marks unmet power demand. Figure 33 shows the calculation of this metric through three subplots. The first subplot shows the power supplied and power demand of a nominal habitat. The two values are identical. The second subplot shows the power supplied and power demanded in a disrupted habitat. The disruption is an intensity level 3 coolant leak at 600 s at dawn with a human agent. In the wake of the disruption, the power demand is larger than the power supplied. In this situation, not all power demands are met. The final subplot shows the nominal supply minus demand vs the disrupted supply minus demand. The value is always zero for the nominal habitat and negative for a time for the disrupted habitat.

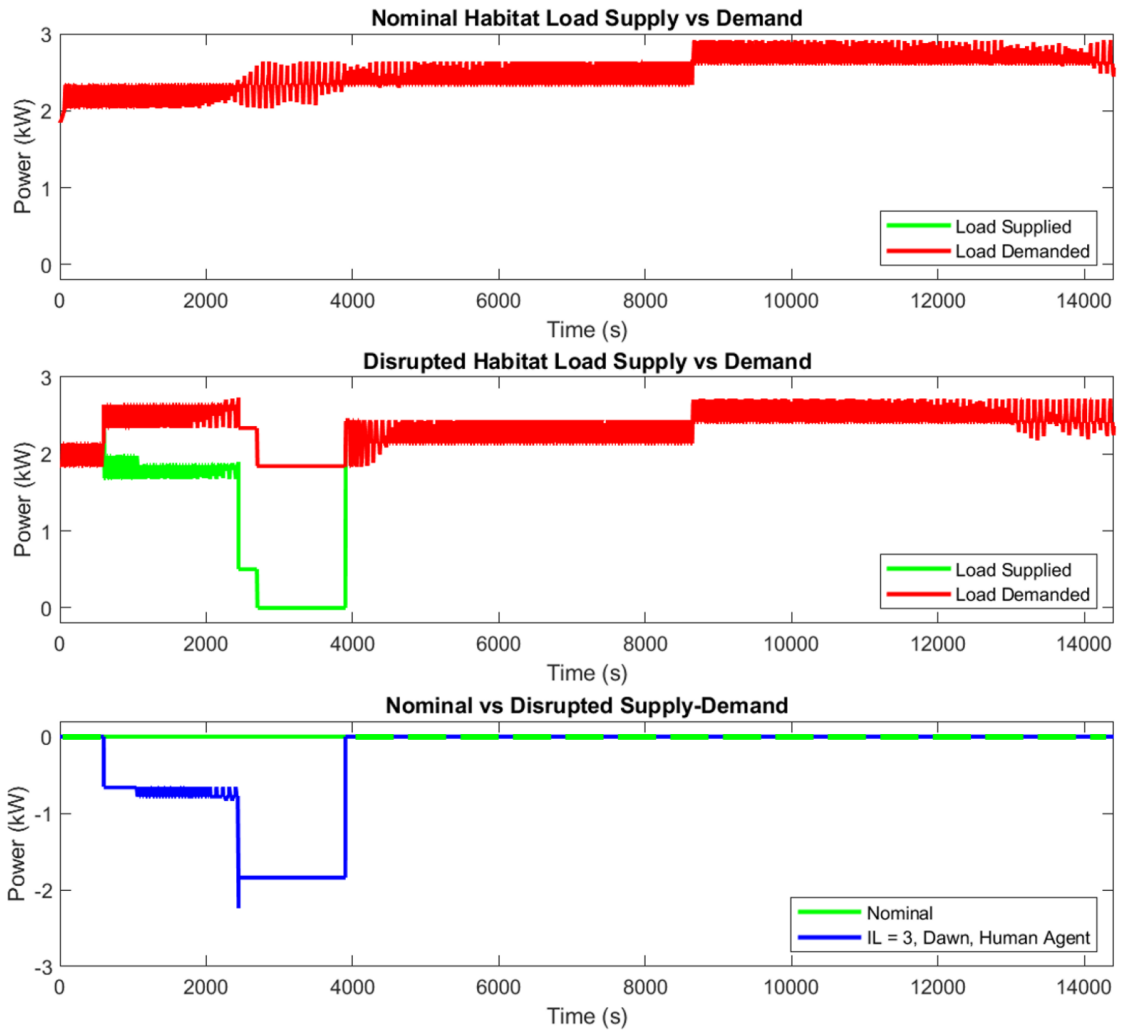


Figure 33: Power Supply and Demand of Nominal and Disrupted Habitat

To track the second function, we look at how often the habitat is within the set temperature and pressure ranges. The temperature in the habitat is set to stay between 296.15 K and 300.15 K, and the pressure is set to stay between 100,325 Pa and 102,325 Pa. When there is no disruption, the temperature and pressure data look like Figure 33.

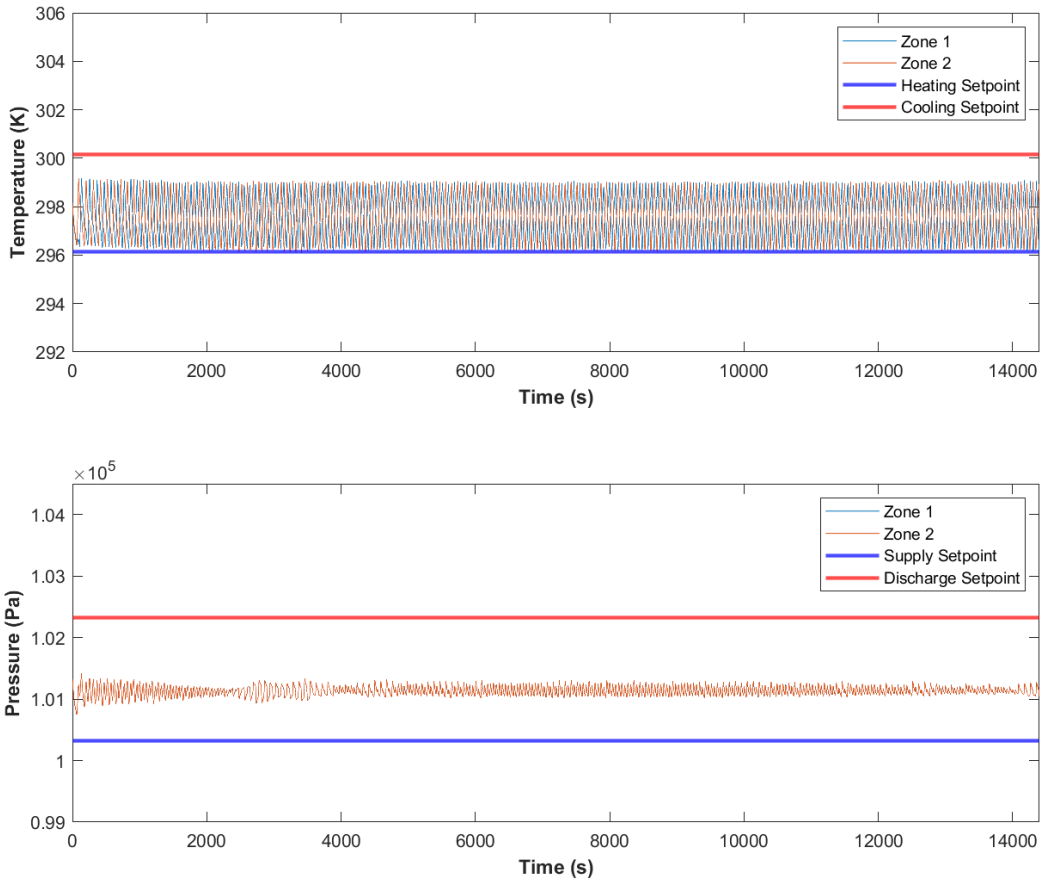


Figure 34: Habitat Temperature and Pressure in Nominal Conditions.

Each subplot in Figure 33 shows two values of temperature and pressure labeled ‘Zone 1’ and ‘Zone 2.’ The two zones exist because one feature of the MCVT is a pocket door that can be closed to seal the interior environment into two separate zones in case of a disruption. For the simulations in this thesis, the pocket door is always open, so the temperature and pressure in the two zones are similar.

When the ECLSS system is not functioning properly, the temperature and pressure inside the habitat can fall outside these ranges, as shown in Figure 34. Both zone pressures are shown in the bottom subplot, they are just so similar that the lines are overlapping.

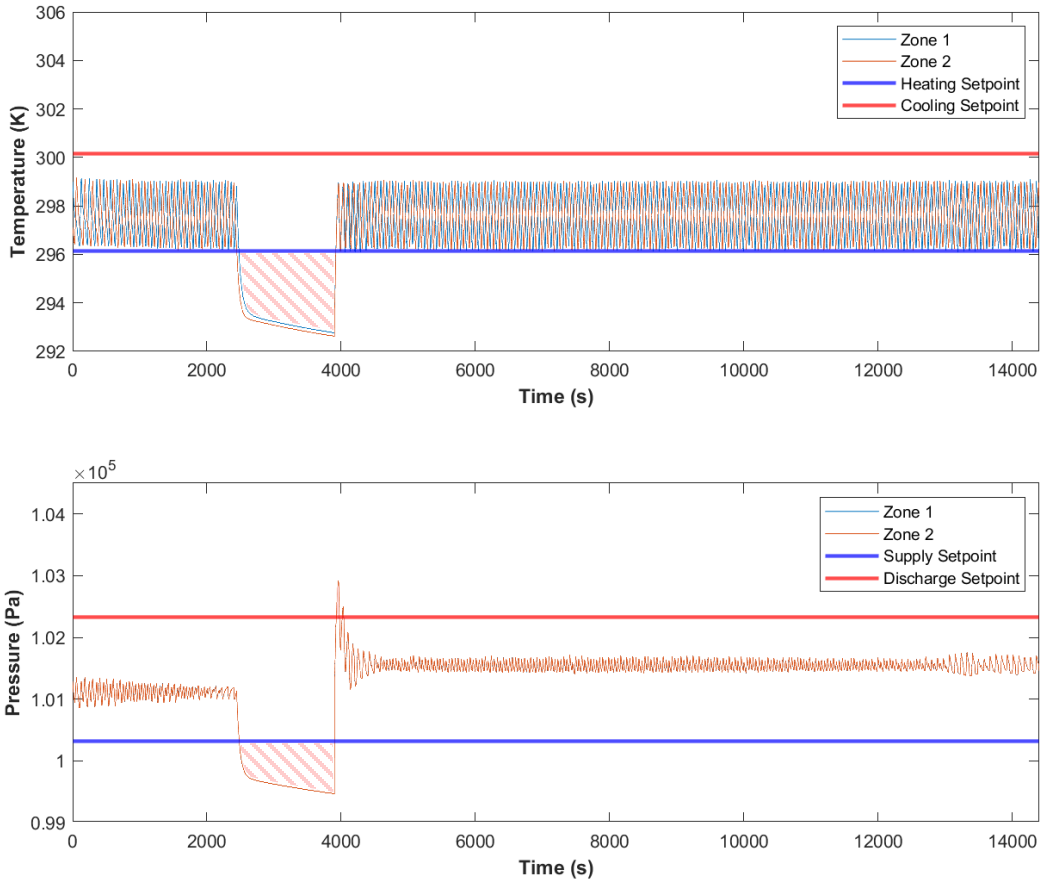


Figure 35: Habitat Temperature and Pressure after Intensity Level 3 Coolant Leak at 600s at Dawn with Human Agent

The metric we track is the area between the setpoint boundary and the temperature or pressure. We calculate the total area for each zone and add them together to get the total area spent outside the ideal temperature/pressure region. The resulting resilience metrics for each path are in Tables 26-28. In calculating these metrics, we notice that the temperature and pressure do not necessarily stay completely within the desired range even in the absence of a disruption. Table 25 lists the resilience metrics for the nominal cases at each time of day.

Table 25: Nominal Habitat-Level Resilience Metrics

Nominal Metrics					
Unmet Power Demand (kJ)		Temperature Setpoint (K•s)		Pressure Setpoint (Pa•s)	
Midday	Dawn	Midday	Dawn	Midday	Dawn
0	0	0.0715	0.1427	0	0

The nominal resilience metrics are zero for unmet power demand and pressure setpoint, but not for temperature. The nonzero values come from times during the simulation where the temperature dips just slightly below 296.15 K.

The lowest the temperature gets in either zone under nominal conditions is 296.10 K, which is less than a tenth of a Kelvin outside the desired range. The temperature dips slightly out of the setpoint range because the ECLSS subsystem is doing its best to maintain the temperature while using as little power as possible, which results in the temperature riding right along the lower boundary of the acceptable range.

The temperature and pressure in the interior environment of the habitat are tightly coupled and the volume of air is small due to the scaling of the MCVT, so slight variations in circumstance (like those caused by disruptions) can affect how often the temperature slips out of the acceptable range. Sometimes the effect of a disruption is positive in terms of maintaining the temperature and pressure, sometimes it is negative.

Even in situations where a disruption does improve the resilience metric, the improvement is very small compared to situations where the temperature or pressure deviates from the desired range due to the cascading effects of the disruption. In Tables 26-28, instances where the resilience metric is smaller than the nominal metric are marked with an asterisk.

Table 26: Path 1 Habitat-Level Resilience Metrics

	Path 1 Metrics					
	Unmet Power Demand (kJ)		Temperature Setpoint (K•s)		Pressure Setpoint (Pa•s)	
	Midday	Dawn	Midday	Dawn	Midday	Dawn
IL 3	0	-4,034	-0.2475	-1,741	0	-412,920
IL 5	0	-5,845	-0.0400*****	-3,182	0	-698,050

At the habitat level, we no longer compare metrics between implementation strategies because we have ideally selected the best implementation strategy for every component in the habitat by this point. Now, we calculate metrics to use in our determination of whether this habitat design is resilient enough. Tables 26-29 show that introducing a disruption causes significant losses in all three metrics at dawn.

Table 27: Path 2 Habitat-Level Resilience Metrics

	Path 2 Metrics					
	Unmet Power Demand (kJ)		Temperature Setpoint (K•s)		Pressure Setpoint (Pa•s)	
	Midday	Dawn	Midday	Dawn	Midday	Dawn
IL 3	0	-6,730	-0.2448	-5,971	0	-1,002,724
IL 5	0	-8,980	-0.0502*	-8,090	0	-1,978,473

Table 28: Path 3 Habitat-Level Resilience Metrics

	Path 3 Metrics					
	Unmet Power Demand (kJ)		Temperature Setpoint (K•s)		Pressure Setpoint (Pa•s)	
	Midday	Dawn	Midday	Dawn	Midday	Dawn
IL 3	0	-19,067	-0.0884	-11,437	0	-2,625,589
IL 5	0	-14,602	-0.0348*	-10,972	0	-2,839,729

Table 29: Final Habitat-Level Resilience Metrics

	Final Metrics					
	Unmet Power Demand (kJ)		Temperature Setpoint (K•s)		Pressure Setpoint (Pa•s)	
	Midday	Dawn	Midday	Dawn	Midday	Dawn
IL 3	0	-6,993	-0.2301	-4,995	0	-952,681
IL 5	0	-8,978	-0.0468*	-7,495	0	1,834,122

At this stage in the design process, we have completed the right-hand design verification side of the vee diagram (see Figure 7).

2.4.4 Iteration to Meet Mission Requirements

Now, the arrow in the center of the diagram in Figure 7 tells us that the next step is to determine whether this design meets mission requirements.

How do we determine whether the design meets mission requirements? This decision must be made separately for each mission requirement. Some major requirements are performance, cost, and resilience.

1. Performance

Determining whether a design meets mission performance requirements involves ensuring that the habitat has the capabilities to carry out the mission objectives. If one mission objective is to map as much of the surface around the habitat as possible, the habitat might need to support rovers, which would entail more power demand than a habitat with no rovers. The best way to ensure that a habitat design meets performance requirements is to include the requirements in the verification process by modeling them into the simulation platform being used. If we know that a habitat will need to support rovers, we should adapt the MCVT to include the power demand associated with using rovers.

2. Cost

Another important constraint is cost. A mission's budget affects every single decision made, from maximum payload to mission length. The actual cost needed for a mission design cannot be known until all the spending has been done (if even then!), so we use estimation techniques to get an idea of price. There are three main ways that cost estimation is done for space missions today: bottom-up, analogous, and parametric (Wertz, 2018).

Bottom-up cost estimation entails making selections for every element involved in a mission, from components to team-member salaries to fuel. Bottom-up estimation works best a little farther along in the design process, when we know better what exactly a mission will look like.

Analogous cost estimation involves comparing a mission to previous missions of similar caliber. Analogous estimation is a highly effective method of cost estimation when the mission at hand is very similar to other missions that have been achieved prior, like flights in the Space Shuttle Era or today's regular trips to and from the International Space Station.

Parametric cost estimation capitalizes on the data that has been gathered on previous missions of the same generic type as the one being designed. Historical data is analyzed statistically to create Cost Estimating Relationships. Examples of parametric cost models include the Unmanned Space Vehicle Cost Model (USCM) and the Small Satellite Cos Model (SSCM). Both these models predict the cost of a constellation of satellites from conception to reentry for satellites of different

mass ranges. There are also other models that predict the cost of individual elements rather than whole missions, like the NASA Instrument Cost Model (NICM) for a satellite's instrument and Cost Construction Model (COCOMO) for software development.

For the Resilience-Oriented Habitat Design Method, we can choose any of these three methods depending on the context of the mission at hand. For the first every mission, a combination of methods will likely be the best way to estimate cost. Analogous or parametric methods can be used for some areas of design, like launch cost, but there will also be some bottom-up estimation. If the mission being designed is the 2nd, 3rd, or 4th mission setting up a habitat of the same size on a particular body, analogous might be the most accurate way to estimate. If many habitats have been built of different sizes on different bodies, a parametric estimation might be better. Of course, neither of those options will work for the first habitat ever built.

3. Resilience

Another requirement we'll need to meet is resilience, which we measure using resilience metrics. The resilience metric we used was the area between nominal and disrupted performance for a number of performance metrics. There are plenty of well-defined resilience metrics that can be used in designing for anything including space habitats. What we don't have literature on today are more sophisticated, habitat-specific performance metrics. The performance metrics we used, like nuclear power generation, were helpful in comparing between different design options for the same habitat, but they would be meaningless if we tried to use them to compare between habitats of different sizes.

If you're designing a building on Earth, there are a variety of performance metrics like utility cost per square foot, or spatial daylight autonomy, that are widely understood and accepted metrics that can be compared between buildings or to industry standards to help designers understand how their design compares to others (Chopson, 2023). Some metrics designed for Earth buildings might be applicable to space habitats, but we also expect that in the near future we'll begin to see some habitat-specific performance metrics devised that will help in standardizing resilience quantification and determining whether a particular design meets requirements.

There is no one correct way to determine whether a habitat design will meet mission requirements, because every mission is different. One day, when habitats are being designed regularly and we have historical data, there might be a more streamlined approach to determining whether a habitat meets requirements, but for now it will be case-by-case.

In any case, if we determine that the habitat we have designed does meet mission requirements, we can be done with the initial design process. However, it is more likely that the habitat will not meet mission objectives, or that we would attempt to make improvements to the design to meet requirements to a greater degree. In that case, we would go back to the FHA and STA steps of the design process and look for different implementations strategies that could better mitigate hazardous states throughout the habitat, improving habitat performance. We would use data we gathered at the component, subsystem, and habitat levels to point out areas of the design that would likely benefit from a closer look and start brainstorming in those areas.

3. CONCLUSIONS

3.1 Summary

The goal of this research was to propose and demonstrate a resilience-oriented extra-terrestrial habitat design process using the ideologies and tools developed by the Resilient Extra-Terrestrial Habitat Institute.

In Chapter 1, we walked through how RETHi came to adopt these ideologies and tools over the past four years. We discussed the institute's state-trigger way of thinking that led to taking a control-theoretic approach to resilience. We talked about the development of RETHi's database of potential disruptions, hazardous states, and safety controls and how they were used in designing the MCVT for habitat simulation. We introduced the concept of control effectiveness and its four sub-metrics for predicting the usefulness of a safety control before simulation, and we did an investigation into existing resilience metrics to find those most suitable to this research.

In Chapter 2, we introduced the Resilience-Oriented Extra-Terrestrial Habitat Design Process by displaying the adapted systems vee diagram. We explained that the left-hand side of the vee is where designers think about what functionalities a habitat, subsystem, or component needs to function, then define requirements to ensure that the needs are met. In this section, we also brainstormed disruptions that could affect each system, the hazardous states they might cause, and the safety controls that could mitigate them. We then moved over to the right-hand side, explaining the next steps of verifying designs created out of the safety controls brainstormed in the first half of the process at each system level using the MCVT.

We then developed the Cooling System Cascade disruption scenario to be the ideal context for demonstrating the design process. We used the CSC to walk through the design process, giving an example of how each step would be carried out. From the first step, habitat-level requirements development, down to the component-level, back up to habitat-level verification, we demonstrated each step of the design process in detail.

We completed the body of this research with a discussion about how we can evaluate a habitat design, and what to do when a habitat design does not meet mission objectives.

3.2 Key Findings

This thesis builds upon the ideas developed in the first three years of the Resilient Extra-Terrestrial Habitat Institute and puts theory into action. The concepts of a state-trigger model for habitat performance and a control-theoretic approach to habitat safety led to the database of habitat disruptions, hazardous states, and safety controls that then became inspiration for the MCVT. The MCVT went from an idea for a way to simulate habitat performance to a highly complex, tightly coupled platform capable of simulating a habitat's response to a variety of disruptions in real time. Control effectiveness was proposed, streamlined, tested, and verified as what it is today.

All the aforementioned progress paved the way for this research to consider how we can put theory into practice. We looked at the big picture of RETHi's ultimate goals and how we could use what has been learned so far to meet them. The result was a resilience-oriented extra-terrestrial habitat design process that combined the state-trigger approach, control effectiveness, and the MCVT to demonstrate how we can use RETHi's work to carry out the institutes ultimate goal of aiding in the design of extra-terrestrial habitats.

In developing the design process, we found the best way for each tool to contribute to a habitat design. We learned that doing a state-trigger analysis toward the beginning of the process is a good way to develop an option space of safety controls and implementation strategies, and we learned that control effectiveness can be used to compare implementation strategies for the same safety control without needing to do any experimentation.

We also learned that the MCVT can help us visualize the impact of implementation strategies with different control effectiveness values and aid in design verification. Using the MCVT was more effective for verifying and evaluating habitat designs than an analytical approach because we were able to see how the different subsystems and components in a habitat affect each other without evaluating the effect of each one on its own and combining them at every time step by hand.

In addition to using work that had already been done, we learned that there are aspects of a design process that had not yet been considered. We learned that, before performing a state-trigger analysis, we had to brainstorm states and triggers to analyze, which led to the addition of the functional hazard analysis step of the design process. We learned that we needed more than just one simulation to evaluate how an implementation strategy was performing, because we needed to account for situations in which the implementation strategy failed. To do so, we included the discussion on what paths the habitat can take and combined the weighted resilience metrics from each path to give a more complete evaluation of each implementation strategy. We also learned the importance of being able to quantify the resilience of a component, subsystem, and habitat, and did our best to capture each system's performance for our needs.

3.3 Limitations and Areas for Improvement

One limitation of the research presented in this thesis is the subjectivity in the process of assigning control effectiveness values to implementation strategies. An implementation strategy's control effectiveness vaules are assigned using engineering judgement as a way to quickly express how it compares to other implemetation strategies. The drawback of having such a quick and easy way to compare strategies is that the value selected are not based on data or experimentation. The values assigned represent how the designer predicts they compare, as opposed to how they actually compare.

The subjectivity in assigning control effectiveness values heavily affects the final resilience metrics for each implementation strategy because the probabilities that a strategy is implemented well and holds up over time are taken directly from control effectiveness. To get an idea of how much our assigned control effectiveness values affect the final ranking of implementation strategies, we could perform a sensitivity analysis. A sensitivity analysis would ential varying one strategy's control effectiveness values little by little to see how much change it takes to alter the final ranking of implementation strategies. If changing the control effectiveness values just a little bit causes the rankings to shift, that tells us that we should either be more careful in assigning control effectiveness values, or take the final implementation strategy rankings with a grain of salt. In that case, we could determine that using judgement-based control effectiveness values as our probabilities of success and failure is not an effective way to capture an implementation strategy's

behavior. Of course, the ultimate goal of control effectivenss is to be able to assign its values based on data and experimentation, making them more realiable which would solve this problem. We could also discover that it takes a lot of change in control effectiveness values to change the final ranking of implementation strategies. In that case, we could continue doing this method of analysis.

A second limitation of the research is its dependence on the MCVT for verification. The MCVT can only reliably run for four hours, which is not enough time to simulate many repair actions. For this research, we selected a relatively quick safety control in hopes of avoiding situations where the MCVT could not simulate the repair from start to finish, but we still ran into times when the simulation ended before the repair was complete.

In addition to the simulation-length limitation, the MCVT also presented a limitation in terms of timing. The MCVT underwent an update starting in December 2022 that was supposed to be finished by February 2023, leaving enough time to analyze multiple safety controls. In reality, the MCVT was not bug-free and ready for use until much later, so this research only investigates one safety control.

The MCVT is also not equipped to simulate the probability of availability portion of control effectiveness, so we were not able to see the effect of varying that parameter. RETHi is currently modelling a new simulation platform that will specialize in lower-fidelity, long term simulations on the order of decades called the Control-Oriented Dynamic Computational Model (CDCM). The CDCM will be a useful platform for investigating how the probability of availability affects an implementation strategy's effectiveness.

An area of improvement for this research would be the development of more sophisticated resilience metrics at each system level, especially at the whole-habitat level. Because we only investigated one safety control and one subsystem, we were able to capture the habitat's performance with a few performance metrics, but if we were actually outfitting an entire habitat, we would need to track many more performance metrics to grasp the habitat's response to disruptions.

REFERENCES

American Leak Detection. American Leak DetectionTM. (2015, July 15). https://www.americanleakdetection.com/blog/2015/july/fixing-leaky-pipes-with-epoxy/

Ayyub, Bilal M. “Systems Resilience for Multihazard Environments: Definition, Metrics, and Valuation for Decision Making.” *Risk analysis* 34.2 (2014): 340–355. Web.

Bruneau, Michel et al. “A Framework to Quantitatively Assess and Enhance the Seismic Resilience of Communities.” *Earthquake spectra* 19.4 (2003): 733–752. Web.

Chebbo, L., & Bazzi, A. (2023). Power. *MCVT v6.3 Documentation*, 24–35.

Chopson, P. (2023, May 2). *5 building performance metrics every architect should know*. cove.tool. https://cove.tools/blog/5-building-performance-metrics-architects

Cilento, M. (2022). *Resilient Extra-Terrestrial Habitat Design using a Control Effectiveness Metric* (Master’s thesis, Purdue University, West Lafayette, United States).

Cheng, Congcong et al. “Improved Integrated Metric for Quantitative Assessment of Resilience.” *Advances in mechanical engineering* 12.2 (2020): 168781402090606–. Web.

Das, Laya et al. “Measuring Smart Grid Resilience: Methods, Challenges and Opportunities.” *Renewable & sustainable energy reviews* 130 (2020): 109918–. Web.

De Witt, J. K., Edwards, W. B., Scott-Pandorf, M. M., Norcross, J. R., & Gernhardt, M. L. (2014). The preferred walk to run transition speed in actual Lunar Gravity. *Journal of Experimental Biology, 217*(18), 3200–3203. https://doi.org/10.1242/jeb.105684

Dyke, S. et al. (2018). RETHi Proposal Technical Narrative.

Dyke, S. J. (2019, June 26). *Resilient Extraterrestrial Habitats Institute (RETHi)*. NASA. https://www.nasa.gov/directorates/spacetech/strg/stri/stri_2018/resilient_extraterrestrial_habitats_institute_rethi/

Dyke, S.J., Marais, K., Bilionis, I., Werfel, J. (2022). RETH institute Annual Report Appendix A Modular Coupled Virtual Testbed (Version 6.0), NASA Sharepoint.

Harwood, W. (2013, May 10). *NASA says coolant leak on International Space Station no threat to crew, but needs a fix*. CBS News. https://www.cbsnews.com/news/nasa-says-coolant-leak-on-international-space-station-no-threat-to-crew-but-needs-a-fix/

Henry, Devanandham, and Jose Emmanuel Ramirez-Marquez. "Generic Metrics and Quantitative Approaches for System Resilience as a Function of Time." *Reliability engineering & system safety* 99 (2012): 114–122. Web.

Howell, E. (n.d.). *The New Space Race*. Encyclopædia Britannica. https://www.britannica.com/explore/space/the-new-space-race/

Ilyas, M., Cho, K., Park, S., & Baeg, S.-H. (2016). Absolute navigation information estimation for micro planetary rovers. *International Journal of Advanced Robotic Systems*, *13*(2), 42. https://doi.org/10.5772/62250

Kitching, R. (2020). *A Control-Theoretic Approach to the Resilient Design of Extra-Terrestrial Habitats* (Master's thesis, Purdue University, West Lafayette, United States).

Kritzinger, D. (2016). Functional Hazard Analysis. In *Aircraft System Safety: Assessments for initial airworthiness certification* (pp. 37–57). essay, Elsevier, Woodhead Publishing.

Leveson, N. (2004). A New Accident Model for Engineering Safer Systems. *Safety Science*, Vol. 42, No. 4, pp. 237-270.

Sellén, M. (2023, March 22). *Does AC stop leak sealer actually work?*. Mechanic Base. https://mechanicbase.com/ac/ac-leak-sealer/

Wada, Y. (n.d.). *A Mini Spaceship for One*. JAXA. https://global.jaxa.jp/article/special/eva/wada_e.html

Wertz, J. R., Everett, D. F., & Puschell, J. J. (2018). Cost Estimating. In *Space Mission Engineering: The New SMAD* (pp. 290–314). essay, Microcosm Press.

Yarveisy, Rioshar, Chuan Gao, and Faisal Khan. “A Simple yet Robust Resilience Assessment Metrics.” *Reliability engineering & system safety* 197 (2020): 106810–. Web.