07ICES-237

Modeling Stochastic Performance and Random Failure

Haibei Jiang, Kaustubh Bhalerao, Luis F. Rodríguez
Department of Agricultural and Biological Engineering, University of Illinois at Urbana-Champaign

Scott Bell, David Kortenkamp
NASA-Johnson Space Center, Houston TX 77058

Alfred Soboyejo
Department of Food, Agriculture & Biological Engineering, Ohio State University

ABSTRACT

High costs and extreme risks prevent the life testing of NASA hardware in relevant scenarios. These inherent limitations prevent the determination of sound reliability bounds for NASA hardware, thus it is unclear the true risk assumed in future missions. The simulation infrastructure for determining these risks is developed in a configurable format here. Preliminary results in preparation for validation testing are reported. A stochastic filter simulates non-deterministic output from the various unit processes. A maintenance and repair module has been implemented with several levels of complexity. Two approaches to simulated system life testing have been proposed for future validation.

BACKGROUND

Unlike the software or transistor industry, the cost and risk involved in determination of system reliability is very high for NASA. Flight ready hardware is often built only once due to the extremely high production costs. A better understanding of reliability and robustness is motivated by recent NASA objectives of implementing a Lunar Outpost. Such long-term missions offer stark contrast in life support to current implementations of the International Space Station (ISS) and the Space Shuttle. Regenerative systems become an increasing necessity as mission length increases. It is also suggested that the Lunar Outpost shall be utilized as a test bed for Martian exploration indicating the mission lengths and the necessity for regenerative systems will only increase going forward. (NASA, 2006)

Such systems will need to rely on resources within the system, operating nearly independently from Earth. Currently considered architectures indicate that resources will only arrive on a six month basis to a Lunar Outpost. Intervals on Mars may be as long as 2 years. As mission length increases, even hardware with the highest manufacturing standards should be expected to fail on occasion, however, the system level impacts are not entirely clear in any complex system. It is desired to increase this understanding such that redundancy, contingency, and maintenance planning can occur.

Current NASA alternatives for contingency planning involve primarily anecdotal evidence stored in 'lessons learned' databases such as the ISS Risk Management Application (IRMA) and a Probability Risk Assessment (Futron and Perera, 2005). The IRMA utilizes a two dimensional risk assessment approach for analysis of systems. The 'likelihood' and 'consequence' of an event is judged by designers, operators, astronauts, and analysts in a two dimensional matrix. If both the likelihood is high and the consequence is dire then action in this area is prioritized by ISS program managers. The limitation to this is system is that it does not rely on actual life data, but rather on the expert opinions of individuals close to the system. As admirable as these individuals are, it is impossible to completely remove all subjectivity in this process.

Similarly, NASA has ongoing work in Probabilistic Risk Assessment (PRA) studying the failure modes of the Space Shuttle (Pate-Cornell, Dillon, and Guikema, 2005). This approach also relies heavily on expert opinion. Failure modes are identified, possibly via the IRMA database, but any individual active in Shuttle design, maintenance, operations, or analysis. The impact of faults is tracked through the system as these failed components impact related system components. The probability of subsequent failures is determined via experiment or assumed. The probability of these failures causing a total system failure is determined by considering the cascade of conditional probabilities. The limitation in this system is the magnitude of the effort required to assemble the entire fault tree. All of the conditional probabilities will never be completely known due to limited ability to test each combination of system failures in relevant environments. Expert opinions are utilized in the areas where actual data is lacking,

however, this suffers from the same objectivity limitations as IRMA.

Other alternative approaches exist as well including Failure Modes and Effects Analysis (FMEA) (Case Jones, 1978), What-If (Arsham et al., 1989, Arsham, 1990), and HAZOP (Center for Chemical Process Safety, 1992, Vesely, et al., 1981, Kletz, 1999), all coming from analogous challenges existing from within the chemical processing industry (refs). Effectiveness in each case depends on the focus and objectivity of the assessment teams, the availability of quality data, and the ability to acquire actual missing data to eliminate the prospect of subjectivity.

Classical approaches to describing and analyzing system reliability are well documented (2, 3, 4, 5). Life testing of component hardware and integrated system is the key to enabling those studies. Due to cost and risk to crew and expensive hardware these studies are rarely undertaken in current NASA system development. Although simulation tools have been proposed for predicting reliability for complex systems, very little work has considered the reliability of repairable systems in the space applications previously.

While the model-based approach suggested here is not devoid of the lack of data problem, it is offered as a compromise between the classical data driven approaches and the expert analysis oriented approaches described above. The system will accept either actual experimental data, where available, or assumptions based on the opinions of key experts in each field. Then analysts will simulate the integrated systems seeking insight in desired performance metrics. The BioSim simulation tool utilized here has been shown to be highly reconfigurable (Rodríguez, Bell, and Kortenkamp, 2006). This is suitable for optimization analyses seeking system configurations inherently reliable and robust.

It is anticipated that by creating a virtual environment capable of testing component level and integrated system reliability the following advantages should result:

1. Virtual testing can lead us to determine minimum component reliability which provide various system level reliabilities
2. Maintenance scheduling can be experimented with to determine the workload for crew members
3. The trade off between crew time costs, redundant hardware, and other contingency plans can be considered for their impact on system reliability.
4. Parametric analyses considering equivalent system mass, mean time to failure, mean time between failure, and mean residual life among others should all benefit from an additional reliability assessment contingency planning resource.

MODELING AND SIMULATION APPROACH

Two assumptions have been made in order to simulate stochastic performance and random failures within life support systems. The first is that non-deterministic systems will drift away from nominal in a 'random walk'. This drift in any unit process can affect the subsequent unit process in the chain by providing off-nominal input to that system. If unit process models are responsive to this effect, their output should also reflect an engendered drift. If this is true, then there is always a probability that the system state will randomly drift into a condition that is unsafe for the crew—or a system failure. This affect has been captured by providing a mechanism for stochastic performance of any unit process within the system.

The second assumption is that random failures will occur with any unit process based upon its inherent hazard function. Hazard functions describe the probability that a unit shall fail at any instant in time, given it has not yet malfunctioned. Hazard functions can be described by any distribution function. Once a unit within the system fails it no longer consumes resources or produces products. Processes further down the chain may fail to receive inputs as a result, depending on the system buffering capacity

Thus, it should be noted that the systems does not immediately fail when a service provided by an essential unit process is discontinued due to a random failure. This is a slight departure from classical reliability analysis. Instead, our systems are deemed failed only when the state of the system has drifted to an unsafe condition. So, for example, if an oxygen generation assembly fails, the system is not deemed failed until environmental oxygen partial pressures drop below safe levels for the crew. This has the effect of slightly increasing system life time as compared to classical reliability theory. The assumed workable environmental states are detailed in EXPERIMENTAL DESIGN CONSIDERATIONS section.

The infrastructure for stochastic performance and random failures has been added to the previously developed BioSim life support system modeling tool (Kortenkamp and Bell, 2003).

Given this simulation infrastructure we have run a series of simulation experiments to demonstrate the effectiveness of algorithms utilized. Varying assumptions in stochastic performance and failure frequency have been selected to demonstrate degrading system performance in a simple life support system configured to representative of a Lunar Outpost. To demonstrate the usefulness of the infrastructure, several maintenance schedules will be tested to demonstrate the potential for mitigation of off-nominal operational schemes necessitated by component failures and system drift. The objective of this work is the development of a simulation test bed for life testing of integrated closed-loop life support systems. Work discussed here are preliminary results which shall lead to a validating experiment where a system with an assumed reliability shall be life tested via simulation. Successful implementation shall result in finding of assumed

reliability utilizing typical life testing on the virtual test bed designed here.

INTERFACING WITH BIOSIM

BioSim uses an Application Programmers Interface (API) which enables external program modules to access BioSim functionality. The Stochastic Performance Filter, the Random Failure Module and the Repair Controller are all connected with the existing BioSim to control the components performance, to inject failures and to manipulate simulation with repair activities during discrete time interval. The API also allows us to use generic sensors to monitor the component varying inputs/outputs and to record the failure/repair events that occur during the simulation. Details describing the experiment design are provided in the *EXPERIMENTAL DESIGN CONSIDERATIONS* section.

MODELING OF STOCHASTIC PERFORMANCE

Modeling component stochastic performance generally refers to creating an understanding of the effect of off-nominal input and output. To facilitate this, current component states are assumed to be independent of previous states. This assumption is considered to be a Markovian assumption, suggesting memory-less components. This assumption is fair with components that can be effectively modeled by exponential failure time distributions, suggesting constant failure rate. It is unclear which life support components would be best modeled by exponential distributions, thus this simply provides a convenient starting point for analysis.

Independent states are identified by the discretized increment of simulation time, termed *tick* after the name of the algorithm within BioSim which predicts the next state of the system based on its current state. One hour of simulation time is elapsed with each tick. To simulate stochastic system performance, component inputs and outputs are passed through virtual filters each tick. This adds uncertainties to simulation processes. The filters use a Gaussian random number generator to select a deviation which is applied to the deterministic process model output. The probability distribution for Gaussian random variates is,

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (1)$$

A Java function called *nextGaussian* returns the next pseudorandom, Gaussian distributed *double* value with mean μ and standard deviation σ. The parameter μ is set by the deterministic output predicted by BioSim process models based on their input. The parameter σ is set by a user specified intensity value.

The use of the filter is highly configurable by the analyst in BioSim. It can be attached to any component within the system simply by editing an XML configuration file. Four intensity levels are currently available in BioSim as described in Table 1. Currently ongoing work shall add the functionality of alternative distribution functions and additional unit process specific intensity options.

Table 1. Stochastic filter intensity values

	None	Low	Medium	High
Intensity Value	0	0.2	0.4	0.8

In rigorous analyses, the settings to be used here should be based on component experiment data when available or expert opinion. In this study, we have selected values that actually cause system failures, as our objective is to ensure that the algorithms are sound.

MODELING OF RANDOM FAILURE

Component random failures are determined by the assigned hazard functions implemented in a Random Failure Module. Hazard function defines the conditional probability that a component will fail. Given that it is currently operational. That is, if a component is operation at time t, what is the probability it will fail in the interval ($t, t+\Delta t$). Randomly failed components will cease to provide their service. The following distribution functions have been added to BioSim: exponential, normal, lognormal, uniform, logistic, two-parameter Weibull, three-parameter Weibull, and Cauchy. At the beginning of each tick, BioSim determines current component state: failed or operational. This is done by comparing the current hazard rate with a uniformly distributed random number between zero and one. If the random number is less than the failure rate, the component state is switched to Failed and services are no longer rendered by that component. Failed components will also stop consuming resources.

Two major hazard functions used in this study are as follows.

The hazard function used for normal model is:

$$h(x) = \frac{\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{1 - \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{x} \{e^{-\frac{(y-\mu)^2}{2\sigma^2}}\} dy} \quad (2)$$

For example, given that $\mu = 450$, $\sigma = 50$, the plot of $h(x)$ is shown in Figure 1.

The hazard function used for exponential model is:

$$h(x) = \lambda \quad (3)$$

If we let $\lambda = 0.5$, the plot of $h(x)$ is shown in Figure 2.

REPAIR AND MAINTENANCE ALGORITHMS

A set of algorithms are currently under implementation in BioSim to simulate the corrective maintenance, or repair and preventive maintenance actions that might be undertaken by the crew. Three levels of repair and maintenance are proposed at this time. The algorithms for each are shown as follows. Figure 3 is the flow chart representation of level I, II, III corrective maintenance algorithms and preventive maintenance algorithm.

Level I Repair

1. Malfunctions are diagnosed instantaneously and repair actions are taken immediately after a component is failed;
2. One tick, or one hour, will be the time required to repair a failed component;
3. All the repairs are perfect, restoring the component to an as-good-as-new condition;
4. Simultaneous failures are possible. If several components, n, fail at the same tick i, or on back-to-back ticks, for example, i and i+1, they will all be restored at the tick number i+2.

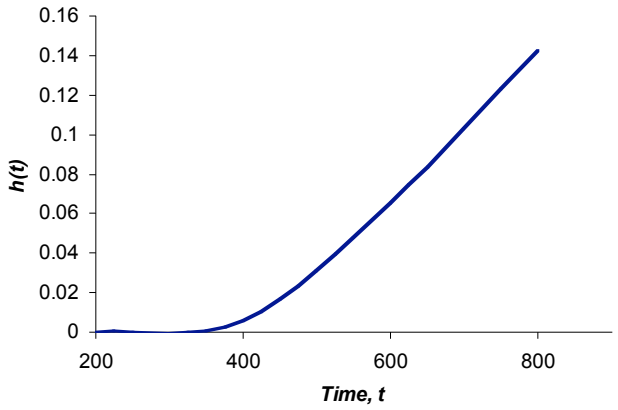

Figure 1. *h(t)* plot for Normal Model

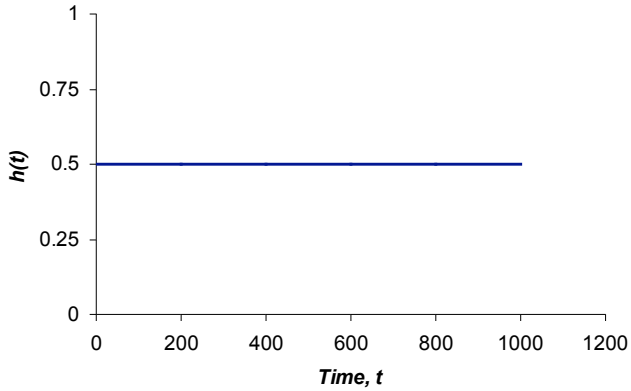

Figure 2. *h(t)* plot for Exponential Model

Level II Repair

The following conditions are added to Level I Repair focusing on repair time.

5. If more than one component is failed, repair activities will be queued in the order which malfunctions occur. Priority repairs can be specified on a component specific basis by the analyst;
6. To simulate realistic troubleshooting and repair events, a random number is selected from the uniform distribution and applied to the repair time. The duration of each repair activity will range from 1 to 24 hours, which is adjustable by editing simulation configuration XML file;

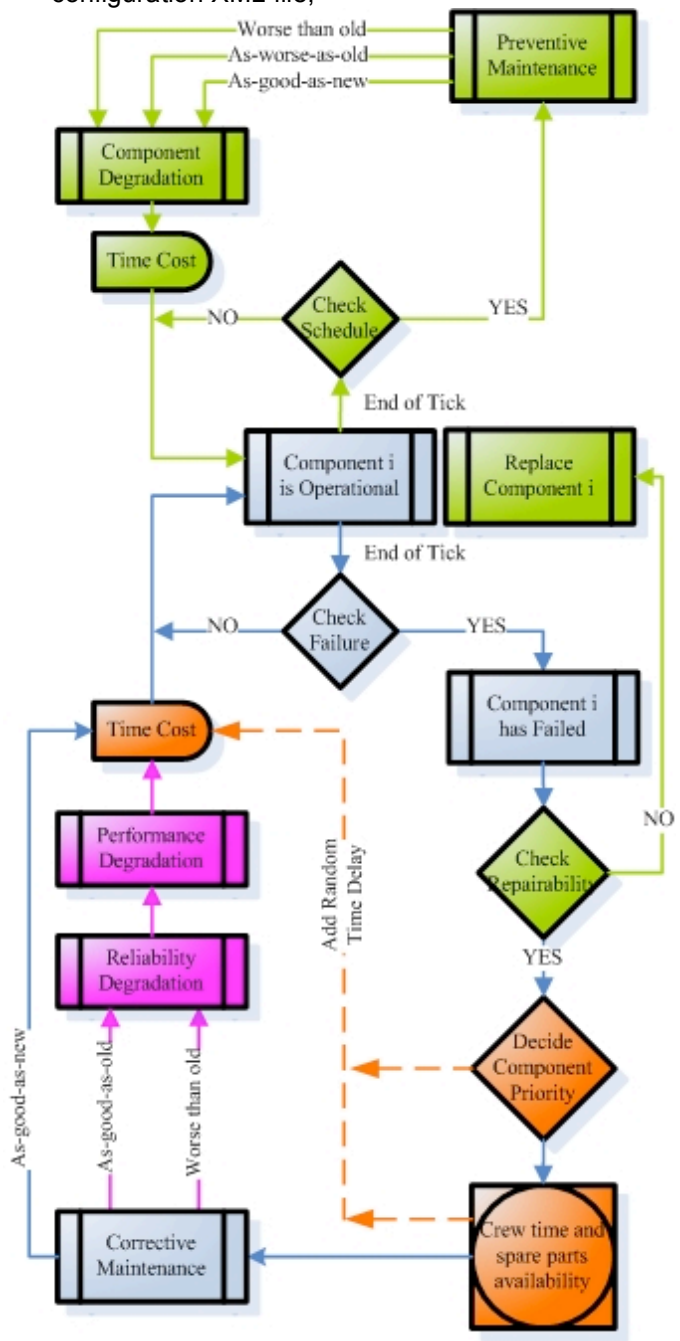

Figure 3. In this figure, BLUE, ORANGE and PINK blocks and arrows denote Level I, II and III corrective maintenance algorithms, respectively. GREEN blocks denote preventive maintenance algorithm design. If lower level maintenance is assumed, the algorithm by passes higher level block unaltered.

Level III Repair

The following conditions are added to Level II Repair focusing on the quality of repair.

7. Imperfect repairs may occur. This has several potential implications. Hardware may be restored to as-good-as-old or worse-than-old conditions. Further, the quality of hazard rate functions may be diminished as a result of substandard repair. These imperfections can occur in any combination desired by the analyst. The quality of repair is depicted graphically in Figure 4 with an assumed log-normal hazard function.

Preventive Maintenance

All repair algorithms are modified for periodic preventive maintenance simulation.

8. Rather than waiting for hardware failure, hardware can undergo a preventive repair event which restores it to as-good-as-new or as-good-as-old, or even worse-than-old, as is described in Figure 4. Further, hazard rate functions may also be diminished, due to faulty repair. These options are selected on a component basis by the analyst.
9. Non-repairable components will be considered. In this case, when a non-repairable component fails, the action taken is to replace with a standby, if available.

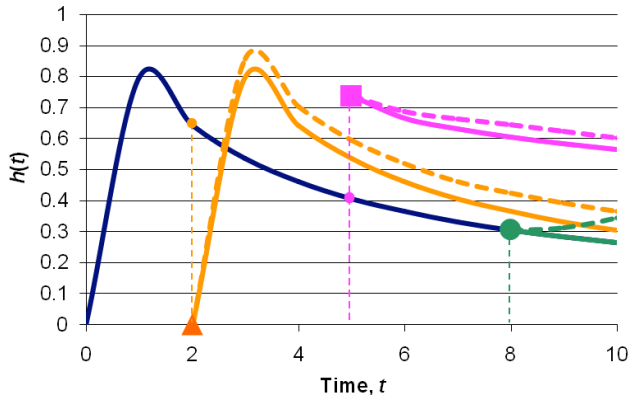

Figure 4. Imperfect repair suggests a failed unit can be repaired to as-good-as-new, as-good-as-old or worse-than-old condition. As is shown in this diagram, three different repair events occurred which represent as-good-as-new (circle at t=2), worse-than-old (square at t=5) and as-good-as-old (triangle at t=8) repair conditions. In addition to the occurrence of imperfect repairs, it is shown reliability degradation can also happen with new hazard functions resulting as demonstrated by the dotted line.

SIMPLIFIED LUNAR OUTPOST CONFIGURATION

A scenario related to upcoming Lunar Outpost is selected for the purposes of testing here. The configuration is simplified for these purposes of validation via future simulation experiments. Simplification provides the ability to determine system reliability via the classical reliability analysis approaches. This system is depicted in Figure 5 where the system boundary, the thin black line defines a control volume. The crew lives within the volume consuming prepackaged food and disposing of solid waste overboard. Excess carbon dioxide generated by the crew via respiration is removed from the bulk atmosphere utilizing a variable carbon dioxide concentration and removal system (VCCR) and sent overboard. Grey water produced by the crew is collected and processed via a wastewater processing system (WWP). Potable water is produced by the WWP and consumed by the crew and the oxygen generation system (OGS). Elemental oxygen and hydrogen are the products of the OGS; hydrogen is released overboard and oxygen is buffered within the system. An injector from the oxygen buffer releases oxygen into the bulk atmosphere at a user specified rate. The crew consumed oxygen from the bulk atmosphere. A Lunar Design Reference Mission prepared by Hanford (2006) has been consulted in preparing this scenario. No control systems will be incorporated here as to allow the system to malfunction, to facilitate the study of reliability.

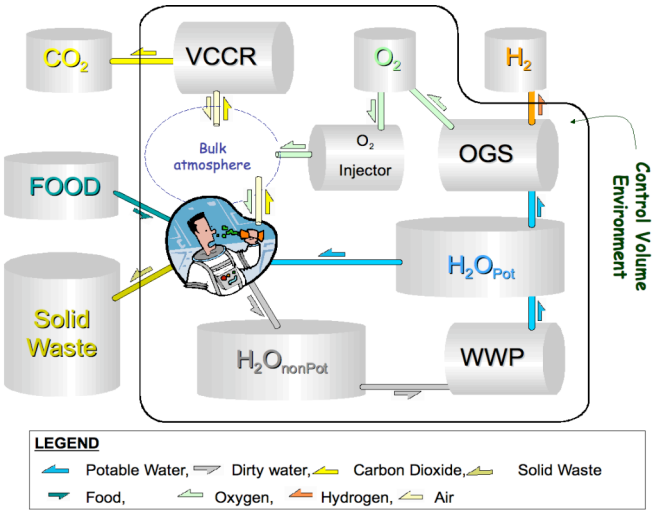

Figure 5. Configuration utilized during preliminary testing

Any or all of the components within the system can perform deterministically or stochastically. To further simplify this analysis it has been assumed that the water buffers shall perform deterministically, as they are considered to be open vessels. In particular, the OGS and the WWP shall have brittle hazard functions, which should engender system failures. The crew will also have a hazard function, though not as brittle as the hardware, representing a life expectancy of 80 years. Stochastic performance is also an available option for each component. Buffers will not perform stochastically, although injectors shall. The Crew, OGS, VCCR and WWP shall also perform stochastically. Please refer to Table 2 for each unit process within the system

regarding hazard functions and stochastic performance assumptions used in this preliminary study.

EXPERIMENTAL DESIGN CONSIDERATIONS

The hypothesis which is to be tested is as follows:

The enhanced BioSim simulation tool should allow prediction of system reliability at least as good as those predicted by the classical reliability theories.

The experimental objectives are to validate the usefulness of the simulation infrastructures using the simplified Lunar Output configuration. Ultimately, by evaluating the impacts brought by the factors simulated here, it is anticipated that true system reliability should improve through better maintenance and contingency planning. In addition, the simulation tool should also be useful for defining component reliability requirements, given a desired system reliability objective. In performing these experiments, it is desired to quantify and interpret the potential impact and the deviation of the theoretical reliability from experimental reliability.

Table 2 Simulation Assumptions

Module Name	Hazard Function and Parameters	Stochastic Intensity Level
OGS	*Exp(0.0027)*	*Medium*
VCCR	*Exp(0.001)*	*Medium*
CO2 Store	*Exp(0.0033)*	*N/A*
O2 Store	*Exp(0.0033)*	*N/A*
H2 Store	*Exp(0.0033)*	*N/A*
Crew	*Normal* $(7*10^5, 4*10^4)$	*Medium*
Air Injector	*Exp(0.001)*	*Medium*
Food Store	*Normal(300, 5)*	*N/A*
Power Store	*Exp(0.0033)*	*N/A*
WWP	*Normal(450, 5)*	*Medium*
Waster Water Store	*Exp(0.0033)*	*N/A*
Dirty Water Store	*Exp(0.0033)*	*N/A*
Grey Water Store	*Exp(0.0033)*	*N/A*

EXPERIMENT CONTROLS

Simulation components are classified into two categories: storage components and regenerative components. A complete list of experimental control variables and the settings are shown in Table 3. *Default* in the table means the assumptions are as same as the ones shown in Table 2.

System failure conditions are selected based on crew safety and are described in Table 4.

Table 3 Experimental control variables

Configuration No.	Stochastic Intensity Level	Storage Component Reliability	Regenerative Component Reliability	Repair Level
1	*None*	*Default*	*Default*	*I*
2	*Low*	*Default*	*Default*	*I*
3	*Medium*	*Default*	*Default*	*I*
4	*High*	*Default*	*Default*	*I*
5	*Medium*	*High*	*Low*	*II*
6	*Medium*	*Low*	*High*	*II*
7	*High*	*High*	*Low*	*III*
8	*Low*	*Low*	*High*	*III*
9	*High*	*Low*	*High*	*III*
10	*Low*	*High*	*Low*	*III*

Table 4. System Failure Conditions

	CO_2 Partial Pressure	O_2 Partial Pressure	Water Mass	Source
Crew Failure	>1kPa	<10.13kPa >30.39kPa	<1 kg	BVAD Hanford, 2004

LIFE TESTING EXPERIMENTAL DESIGNS

The following describes two life testing approaches proposed in this study based on simulated life test data.

<u>Maximum Likelihood Estimation using uncensored data</u>

To obtain an estimate for the parameters of a distribution based on life testing we define a maximum likelihood estimator L where

$$L(\tilde{t},\theta)=\prod_{i=1}^{n} f(t_i,\theta)$$

And the vector $\tilde{t}$ defines the failure times t_i observed in an experiment characterized by an assumed distribution f with unknown parameter θ. Presuming the failure times are independent the likelihood function can be minimized, providing the opportunity to solve for θ. For example, assuming exponential failure times gives

$$L(\tilde{t},\theta)=\prod_{i=1}^{n}\frac{1}{\theta}e^{\frac{t}{\theta}}=\frac{1}{\theta^n}e^{-\frac{1}{\theta}\sum_{i=1}^{n}t_i}.$$

This can be linearized to

$$\ln\left(L\left(\tilde{t},\theta\right)\right)=-n\ln\theta-\frac{1}{\theta}\sum_{i=1}^{n}t_i\ .$$

Subsequently, if this function is minimized a maximum likelihood estimate results of

$$\frac{\partial}{\partial\theta}\ln\left(L\left(\tilde{t},\theta\right)\right)=0=-\frac{n}{\theta}+\frac{1}{\theta^2}\sum_{i=1}^{n}t_i$$

or

$$\hat{\theta}=\frac{1}{n}\sum_{i=1}^{n}t_i\ .$$

This theory can be expanded to censored datasets as well, where test units are removed from operation prior to failure. In this case, the joint probability comprising might include reliability functions rather than strictly probability density functions. For example,

$$L\left(\tilde{y},\theta\right)=\prod_{i\in U}f\left(t_i,\theta\right)\prod_{i\in C}R\left(c_i,\theta\right)$$

where $\tilde{y}$ is the set of all the operational times. The vector $\tilde{y}$ is comprised of two subsets of failed and censored U and C, respectively. U is defined by the failed unit lifetimes, t_i as before, whereas C are the censored times c_i.

The specific distribution functions can be selected based upon prior knowledge or assumption. In our case we can run successive BioSim simulations to generate both failed and censored data, depending on computational requirements.

Sequential Life Testing

An alternative life testing approach is to use a sequential sampling technique to test the null hypothesis H_0 versus H_1,

$H_0: p \geq p_0$ or, alternatively $H_1: p < p_0$

where p is the reliability the system. The value p_0 is selected as the lowest acceptable reliability bound. Given the definitions for type I and type II errors: let $P(H_1 \mid H_0)=\alpha$ and $P(H_0 \mid H_1)=\beta$, we define a likelihood ratio L_R as

$$L_R \equiv \frac{L_{1,n}}{L_{0,n}}\ .$$

A sequential testing region can be defined as $A=\frac{\beta}{1-\alpha}$ and $B=\frac{1-\beta}{\alpha}$. The following rules will apply:

1. If $L_R \leq A$, accept H_0;
2. If $L_R \geq B$, reject H_0;
3. If $A < L_R < B$ continue life testing

Specific tolerances for α and β will need to be specified with alternative values for H_0 versus H_1. This is represented graphically in Figure 5.

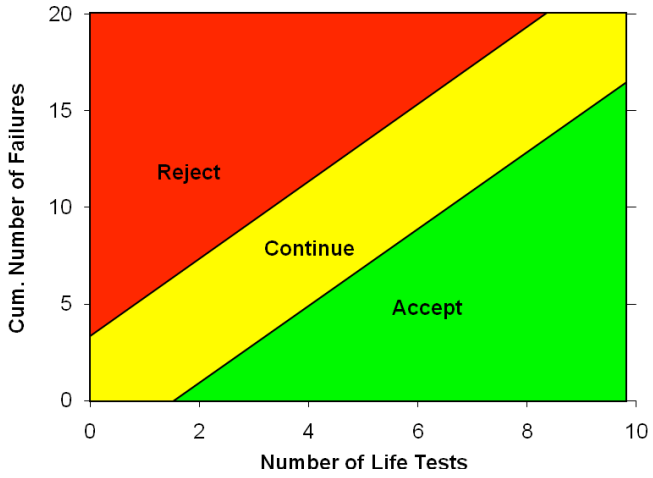

Figure 6. Sequential Life Testing Method

PRELIMINARY RESULTS AND DISCUSSIONS

Configuration 1 to 4 have been tested using the defined experiment control variables. Major experiment results include: (1) Component stochastic performance; (2) Component random failure and repair time; (3) System failure time. Repairable system components are quantified using Mean Time Between Failures (MTBF) while the system itself is non-repairable and System Survival Time is used as a proxy for reliability.

STOCHASTIC PERFORMANCE

In the simulation for configuration 1, none of the components is assigned with stochastic intensity. Data collected show that each of the components was operating with deterministic inputs and outputs.

Figure 7, 8, and 9 illustrate the sampled stochastic performance data of OGS Oxygen production rate, from configuration 2, 3, 4 defined in Table 3. Each configuration has been simulated three times, however, the results shown are individual simulations.

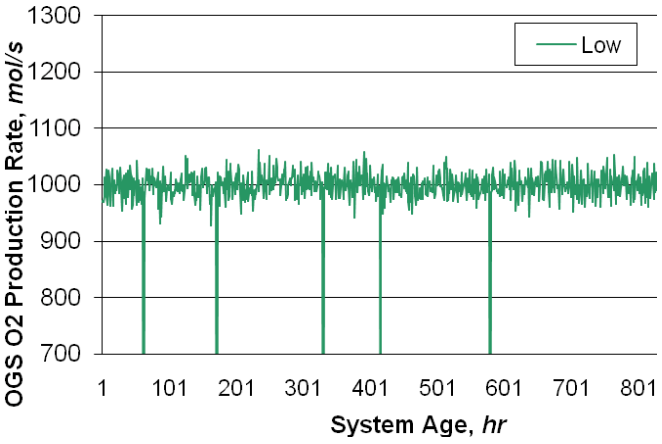

Figure 7. Plot of OGS Oxygen Production Rate using LOW Stochastic Intensity

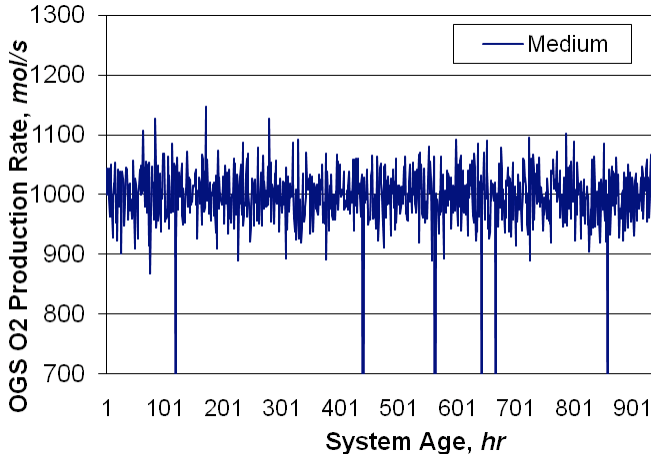

Figure 8. Plot of OGS Oxygen Production Rate using MEDIUM Stochastic Intensity

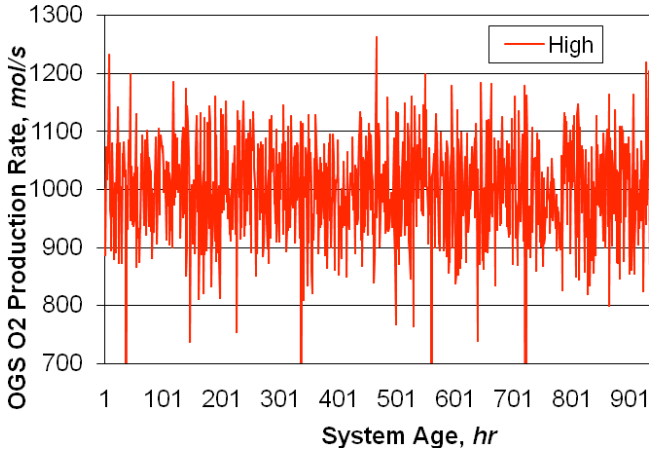

Figure 9. Plot of OGS Oxygen Production Rate using HIGH Stochastic Intensity

The figures show that the actual Oxygen outflow rates in the simulation are oscillating near the nominal production rate which equals to 1000 *mol/s*. The range of the oscillation varies when different intensity values are used. Random failure events can also be observed when the production rate goes below 700 *mol/s*, in which case the component has zero output. The results clearly validate our modeling approach which uses a Gaussian random filter to add uncertainty to component outputs.

RANDOM FAILURE AND REPAIR EVENTS

Based on the observation of the data collected from one of the simulations for configuration 3, where Medium Stochastic Intensity is used, components failure events are detected and the repair module restores the failed component with a fixed time delay of one tick. It is also noticeable that when two components fail in two consecutive ticks, they are restored to initial state at the time that the first failed component should be repaired. For example, Table 5 shows a simulation that lasted for a total of 997 ticks. The OGS randomly failed at tick number 565 and another failure occur later to the Portable Water Storage at the 566th tick. A repair event thus happened at tick number 567 and restored both of the components to as-good-as-new condition. However, this design will be improved in the near future by introducing the constraint that failed components can not be repaired simultaneously.

Table 5. Components random failure time for Configuration 3.

Failure Number *i*	Component Name		
	OGS	VCCR	Portable Water Store
1	101	153	42
2	445	204	60
3	565	324	241
4	682	423	566
5	873	639	881
6	997	852	997
7		997	

Graphically, a Duane plot is utilized to gain insight into the performance of Random Failure Module and Repair Controller Module. The time t_i is the globe failure time plotted on the horizon axis and $N(t_i)$ is the cumulative number of failures through time t_i. The quantity $t_i / N(t_i)$ is called cumulative MTBF which is plotted on the vertical axis in Figure 10, 11, and 12.

Linear functions are fit into the failure data points to test the impact of our repair controller. Three components shown here have different slopes. VCCR has roughly horizontal lines suggesting the component remained stable over the time that the failures were observed and cumulatively neither reliability improvement nor deterioration occurs. Note that all the repair events are designed to restore the failed components to as-good-as-new condition; a question arises: why does the Portable Water Storage unit have such a positive slope indicating an improving reliability function while WWP

has a negative slope indicating a decreasing reliability function? It is currently postulated that is due to the deterministic performance of the portable water storage and WWP.

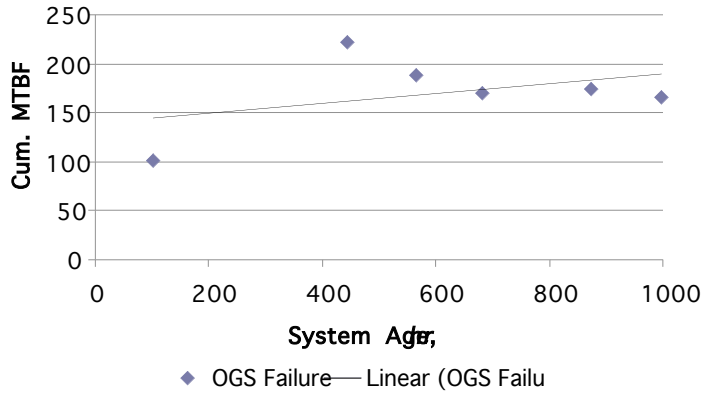

Figure 10. Plot of OGS random failure time versus Cumulative Mean Time between Failures

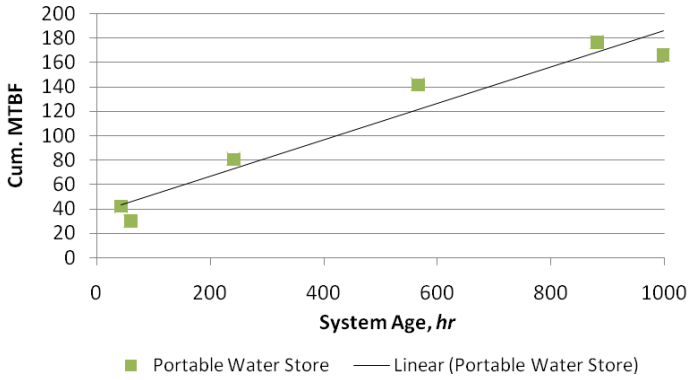

Figure 11. Plot of Portable Water Storage random failure time versus Cumulative Mean Time between Failures

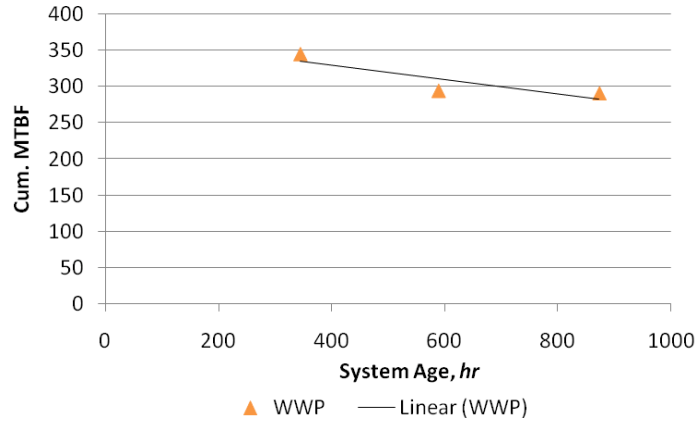

Figure 12. Plot of WWP random failure time versus Cumulative Mean Time between Failures

SYSTEM FAILURE

Current results also show that there is no significant impact caused by the choice of stochastic variables setting on system survival time, although there is some slight decrease with higher intensities. Tests results shown in Figure 13 are scaled in a range from 965 to 1000 ticks. It can be clearly seen that the non-stochastic configuration outperformed the others while the rest are performing at a very similar level. Such a phenomenon is to be explained since the designed optimal input/output levels are comparatively high, and the stochastic uncertainties added to the unit processes are not large enough to reduce the stability of the system. Short repair time and large storage buffers may also be the reason that leads to the result of having a comparatively consistent system survival time.

In the future, a more sensitive system configuration can be designed and the stochastic intensity level can be increased so as to bring more significant unstable factors into the simulation.

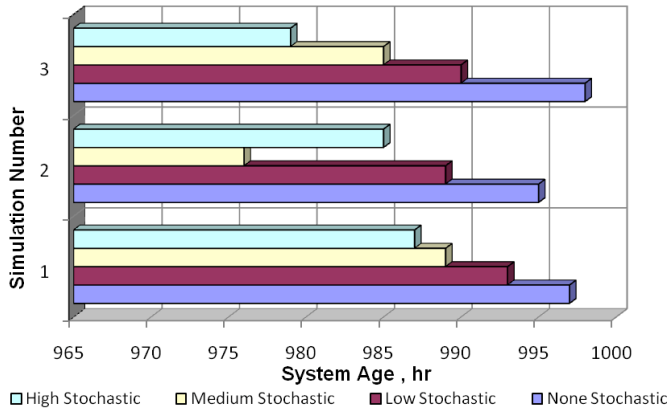

Figure 13. Plot of simulated system survival time for configuration 1, 2, 3, and 4. Each configuration is tested three times

CONCLUSION AND FUTURE WORK

The infrastructure for simulation of a stochastic system including system failure and repair has been implemented in the previously developed BioSim model. This infrastructure has undergone some preliminary testing demonstrating baseline functionality.

To validate this tool, two life testing approaches have been developed. The first approach determines a maximum likelihood estimator, given an assumed probabilistic distribution and either censored or non-censored data. The alternative approach utilizes the likelihood ratio to define bounds upon a testing range seeking to minimize type I an type II errors.

A simplified Lunar Outpost scenario has been proposed. Classical reliability techniques should be applied to this system. It is anticipated that simulation derived reliability shall slightly over analytical predictions due to the inherent buffering capacity of the virtual systems. This is despite some stochastic engendered drift captured by the model. This drift has been shown to vary depending on user selected intensity. It has also been shown that simulations with high stochastic intensity have shorter

simulated lifetimes, though whether the difference is significant can be debated. In any case, it is anticipated that failures due to this drift will be relatively rare, as suggested by the Central Limit Theorem.

System repair has an impact creating increasing and decreasing failure rates on average. This effect should be considered in the design of future systems, if probabilistic distributions characterizing certain hardware can be know.

Eventually with a validated model and proper data describing life support hardware, it is anticipated that several analyses shall be enabled. Minimum component reliabilities may be determined, given a system level objective. Maintenance, crew time costs, and contingency plans can be considered with their impact on reliability. Parametric analyses can also be enabled, such as equivalent system mass, mean time between failures, mean time to failure, and mean residual life, among others.

ACKNOWLEDGMENTS

The work has been funded by NASA grant number NNJ06HA03G.

REFERENCES

1. Kortenkamp, D., and Bell, S., “Simulating Advanced Life Support Systems for Integrated Controls Research,” Society of Automotive Engineers Paper 2003-01-2546, 2003.
2. Ascher, H. and Feingold, H. “Repairable Systems – Modeling, inference, misconceptions and their causes”.
3. Hongzhou Wang, Hoang Pham, “Reliability and Optimal Maintenance”.
4. Steven E. Rigon, Asit P. Basu, “Statistical Methods for the Reliability of Repairable Systems”
5. M.J. Crowder, A.C. Kimber, R.L. Smith and T.J. Sweeting, “Statistical Analysis of Reliability Data”.
6. Hanford, A.J., Lunar Design Reference Mission.
7. Hanford, A.J., Baseline Values and Assumptions Document.
8. Scott Field, Futron, Jeevan Perera, PhD/JD, JSC ,Integrated Risk Management Application (IRMA), NASA Risk Management Conference 2005.
9. M. Elisabeth Pate-Cornell, Robin L. Dillon, Seth D. Guikema (2004), On the Limitations of Redundancies in the Improvement of System Reliability, Risk Analysis 24 (6), 1423-1436.
10. Case, K. E. and Jones, L. L.(1978) Profit Through Quality. Quality Assurance Programs for Manufactures, Institue of Indutrial Engineers, New York.
11. Arsham H., What-if analysis in computer simulation models: A comparative survey with some extensions, Mathematical and Computer Modelling, 13, 101-106, 1990.
12. Arsham H., Feuerverger, A., McLeish, D., Kreimer J. and Rubinstein R., Sensitivity analysis and the what-if problem in simulation analysis, Mathematical and Computer Modelling, 12, 193-219, 1989.
13. Center for Chemical Process Safety (1992). Guidelines for Hazard Evaluation Procedures, with Worked Examples, 2nd Edition, Wiley-AIChE. ISBN 0-8169-0491-X.
14. Vesely, W.E., Goldberg, F.F., Roberts, N.H. and Haasl, D.F. (1981). Fault Tree Handbook. U.S. Nuclear Regulatory Commission. NUREG-0492. Nuclear Regulatory Commission.
15. Kletz, Trevor (1999). Hazop and Hazan, 4th Edition, Taylor & Francis. ISBN 0-85295-421-2.

CONTACT

Questions and comments regarding this work may be addressed to Luis F. Rodríguez at lfr@uiuc.edu