

System Engineering and Integration of Controls for Advanced Life Support

David Overland
NASA Johnson Space Center

Dr. Karlene Hoo
Texas Tech University

Marvin Ciskowski
Hamilton Standard

ABSTRACT

The Advanced Integration Matrix (AIM) project at the Johnson Space Center (JSC) was chartered to study and solve systems-level integration issues for exploration missions. One of the first issues identified was an inability to conduct trade studies on control system architectures due to the absence of mature evaluation criteria. Such architectures are necessary to enable integration of regenerative life support systems. A team was formed to address issues concerning software and hardware architectures and system controls..

The team has investigated what is required to integrate controls for the types of non-linear dynamic systems encountered in advanced life support. To this end, a water processing bioreactor testbed is being developed which will enable prototyping and testing of integration strategies and technologies. Although systems such as the water bioreactors exhibit the complexities of interactions between control schemes most vividly, it is apparent that this behavior and its attendant risks will manifest itself among any set of interdependent autonomous control systems. A methodology for developing integration requirements for interdependent and autonomous systems is a goal of this team and this testbed.

This paper is a high-level summary of the current status

of the investigation, the issues encountered, some tentative conclusions, and the direction expected for further research.

INTRODUCTION

There are subsystem requirements above and beyond the functional requirements of a subsystem. These requirements derive from the need to integrate subsystems into an operational vehicle, and integrate that vehicle into an operational program. When there are dependencies between subsystems, functional requirements and interface descriptions are insufficient to constrain intersystem dynamics. This is the driving rationale behind system engineering.

The more dynamic or autonomous subsystem activities are, the greater the need for system engineering analysis and design methods. Even simple components can provide unexpected interactions when arranged improperly. Much of practical engineering experience is geared to recognizing these hidden interactions and developing design approaches to eliminate or at least mitigate the risks.

Control of non-linear, time-varying dynamic processes offer some of the most difficult system engineering challenges. Control of the Water Recovery Subsystem bioreactor preprocessors is an illustrative example of this.

System engineering of the control system requires understanding the system in context, to determine what architecture and infrastructure is needed, and to understand the operational dependencies of the systems. Our investigation has developed some conclusions about the type of testing, modeling and analysis required to determine that context.

Integrated Testing

The importance of integrated testing is historically well-known. The importance of designing for integration is not always as explicitly understood, especially for complex or interdependent systems.

The development methodology of stand-alone systems is relatively well understood. But when you put systems together, interesting and unpredictable things can happen. Simplistically put, the integrated system behaves differently than the stand-alone systems behave separately. On previous integrated testing projects, integrating of Advanced Life Support (ALS) subsystems has always driven out requirements and identified technology gaps in the ALS program [1].

Based on that experience, both the Advanced Integration Matrix program and its parent, the Bioastronautics program, have identified control system architecture and integration as a critical technology gap for exploration missions [2]. As a result of an internal working group analysis [3] and a NASA workshop [4], it was proposed that an integrated test be developed to explore the design constraints and integration requirements of ALS control systems.

Test Objectives

AIM Test is intended to:

- Stress interfaces
- Identify information flows
- Explore operations concepts and dependencies
- Investigate architecture capabilities and requirements

In order to characterize the architecture requirements, it is important to determine the capabilities needed during a mission, particularly to determine what types of data and autonomous capabilities will be required by crew, vehicle and ground control during complex mission scenarios.

Test Components

Because of the scope of the investigation, development and testing of the controls of ALS systems is only one component of the integrated test. In order to determine top-level architecture requirements, the test was initially proposed as an exploratory test. Because of this, the test is in two parts. The first part of the test was scenario-driven, involving two bio-reactors, a simulation

of the Air Revitalization Subsystem (ARS) from another project, and a scenario designed to elicit onboard and ground-based task capture. Human factors will be instrumental in defining the approach to task capture and analysis tools and methods.

Observing and documenting the controls design methodology is as much a part of the test as data collected during the actual integrated test. Scenario development, task capture, operations concepts and data flow management prior to the test run itself, will also generate test products.

By observing the decision points associated with each task, the data required for decisions, the origin and data-flow paths associated with that data, and its latency and reliability, the initial capabilities of the control architecture can be bounded.

Using task analysis based on operational scenarios as the framing presentation allows investigation into process and methodology, and into separation of concerns between requirements and design.

The following components were initially proposed as test articles:

- Controls Investigation
- Water Recovery Subsystem (WRS) preprocessor systems with independent control systems for each reactor
 - Aerobic bioreactor
 - Anoxic bioreactor
- ARS simulation
- Scenario development and Task analysis
 - Mapping command and data flows to capabilities
- Narrative Integration

Methodology

The investigation up to this point has uncovered several methodologies that may provide assistance in managing development of complex systems. Structured approaches to developing system requirements have been used by the Department of Defense (DoD) to identify and manage coupling between complex systems, of which our test is a microcosm.

The Joint Capabilities Integration and Development System (JCIDS) [5] details a system engineering methodology that enables discovery and capture of root system requirements by first identifying capabilities required to support the program operations concept. Those capabilities can include organizational infrastructure, manpower, logistics support, and technology products. The JCIDS methodology allows evaluation of an architecture's ability to provide identified capabilities, and allows technology gap analysis.

The JCIDS methodology influenced the decision to use capability definition as the starting point of this investigation. From capabilities identified as needed during the scenario it is intended to capture what decisions must be made, where are they made, what information is needed to make those decisions, how does the information get there, and the reliability of the information. These need to be determined to identify whether infrastructure and architecture can provide necessary mission capabilities. This is the beginning point for control architecture engineering comparison.

Controls Investigation

Most of the efforts over the past year have focused on the controls investigation. This has primarily been due to delays in the construction of the second bioreactor. The importance of the work done so far cannot be underestimated.

Even though the decision to investigate the WRS preprocessor bioreactors was imposed on the test plan, the problem has illuminated a number of understandings that would not have been uncovered otherwise. Some are specific to the particular systems under investigation; others are generically applicable to many other systems that will be needed for long-term exploration missions.

The general question that sparked the initial investigation is: What requirements must be levied on each subsystem to enable integration of the control systems? The investigation proposes that the two bioreactors can “stand in” for any two interdependent subsystems, e.g. ARS and WRS.

The question is important because flight systems are developed independently by separate subcontractor organizations, often at different times in the program. Based on the traditional avionics control system development methodology, subsystems are designed according to independently developed System Requirements Specifications (SRS) with tightly controlled Interface Requirements Definitions (IRD). It is assumed that subsystems can be controlled independently of each other, and with few exceptions, it is assumed that software control requirements are derived from hardware design specifications [6].

It became obvious early on in the controls investigation that the traditional approach to developing control software for life support subsystems was inadequate to the task. Review of previous ALS control development efforts [7] showed that the process was not being controlled, only the test equipment.

The development approach to controlling the bioreactors that was adopted for this investigation is based on control engineering methods employed in the process control industry.

Controls Development Process

Basic control engineering identifies three prerequisites are required to develop a process control system. First, the process must be steady-state stable. Second, the process must be controllable; i.e., there must be controlled (dependent) parameters and manipulated (independent) parameters. Third, the process must be observable; i.e., there must be observable parameters that correspond to the controlled parameters. If these three conditions are met, then there are straightforward techniques available for controlling the process. If not, other techniques must be brought to bear, including redesign of the system.

Two things need to be pointed out here. First, these three prerequisites are design-dependent. If they are not met, additional control points, sensors or fluid pathways can be added to the system design to increase the number of degrees of freedom to satisfy the conditions.

Second, control in the engineering sense refers to maintaining the output of the system within the range of desired output in the face of perturbations of the input values, of environmental (unmeasured) perturbations, and of variations in process dynamics. This is accomplished by controlling various equipment (switches, valves, pumps), but component equipment control in and of itself does not equate to process control.

Controls Models

In order to determine whether these prerequisites are met, several types of modeling and analysis must be performed. In this case, three types of models were required.

Initially, in order to characterize the process at all, a stoichiometric model had to be developed [8]. The basic chemistry of biological systems is inherently complex; however much of chemical engineering is initially based on empirical equations which were available in the literature.

After the basic stoichiometry was worked out, an equilibrium model was developed which models the dynamics of the system in equilibrium. This determines whether the first prerequisite is met: the process must be stable, requiring no controls to maintain equilibrium in the absence of system perturbations. The mathematical techniques to determine stability are well-understood. The details of the model are described elsewhere [9], and contains reaction information and substrate transport information. The model is a system of partial differential equations with appropriate boundary conditions.

Data generated by this model had to be compared against actual experimental data, in order to validate several assumptions in the theoretical model. This was done using existing WRS testbeds at Texas Tech

University.

Once stability has been theoretically established and validated, a controls-relevant model has to be developed. Controls-relevant modeling is not as straightforward as the previous two models. Relevance in an engineering sense is dependent on the optimization criteria chosen. Choices include performance efficiency, robustness to perturbations, rate of processing, efficiency of processing, amount (or type) of by-products, upstream or downstream constraints, maintenance frequency, cost of operation, etc. There are a number of mature, robust techniques to optimize profit, but it was understood these would be sub-optimal in this domain.

Because this investigation is based on existing WRS designs, the controls relevant model reflects the optimization strategy of performance. The controls relevant model, a system-theoretical model, was used to analyze system response to varying feed rates and feed composition. This model was also validated against experimental data in the Texas Tech University WRS testbed.

The control relevant model enables determination of the second two prerequisites of controllability and observability. Eigenvalue analysis of the system performance of the model in response to small perturbations in the biomass and substrate shows whether the non-linear system is open-loop stable, open-loop state controllable, and open-loop feedback controllable. It also allows determination of observability.

Depending on the results of the analysis a control approach can be chosen. If the system is not controllable, design changes can be made and different approaches can be tested against the resulting model. Of course, these control strategies must still be compared against experimental data to validate the theoretical models and control algorithms.

Test Results

The details of the models and test results are covered in other papers [10]. Several results need to be described here to set the stage for other conclusions. Two reactors are part of the controls investigation. The analysis, modeling and validation testing described above are currently being performed for the packed bed anoxic reactor. The same process is expected to be completed this year for the aerobic tubular reactor. The initial set of models were developed and validated against a similar set of reactors at Texas Tech University.

The packed bed reactor was found to be open-loop stable. The tubular reactor was found to be open-loop unstable. Because of this, the open loop coupled system is not fully state feedback controllable. The open loop coupled system is fully output feedback

controllable and fully observable. Using a Proportional Integral Derivative (PID) controller on the Texas Tech University testbed confirmed this result. The team is in the process of redesigning the tubular reactor to add additional degrees of freedom. Instead of a PID controller, a Model Predictive Controller is being developed.

Because the WRS laboratory has different test objectives, the redesigned system is being developed in the Bioastronautics Laboratory at JSC.

Lessons Learned from Integrated Testing

These lessons are a product of applying straightforward design methods to advanced systems. The lessons are important because the engineering methods used are fundamental to the discipline of process control engineering and are rarely used in aerospace engineering. These methods have not yet been applied to the types of complex systems that will be required for long-duration exploration missions.

Lesson 1 - Systems must be *designed* for controllability

As stated previously, control in this context means bringing the process back into equilibrium in the desired optimization range when the process is perturbed by input or environmental variations.

Such controllability is design sensitive. As seen even in our initial test development, small changes in design can markedly affect the stability, controllability, and observability of the system. The lesson that may not be obvious is that the control design precedes the hardware design of the system. System modeling, analysis, and experimental validation must precede both control and hardware design.

Because of this, controllability and observability dictate the sensor and effector selection and placement. This is likely to be a different set of selections than those chosen for performance testing of a process.

When there are dependencies between systems, the control design dictates the hardware requirements. As control algorithms are usually implemented in software, another way of saying this is that software design must precede hardware requirements for complex control problems. This stands in contrast to the decomposition of functional system requirements into hardware and software requirements and interface definitions, followed by hardware and software design.

Lesson 2 - Control (A) + Control (B) • Control (A+B)

Controllability is not additive for interdependent systems. This is a fundamental axiom from basic control theory. If there is no coupling, then subsystems can be specified, designed and operated independently

of each other. But without appropriate analysis and modeling across systems, it is impossible to determine whether systems are independent or coupled, or what interactions are possible.

This conclusion was not obvious when we began our test. The primary initial project goal of the project was to determine what requirements must be levied on individual control systems to assure integration after design. It soon became clear that it is not possible to levy integration requirements on each subsystem in the absence of integrated design.

Hardware and software requirements follow system design for coupled systems. Such engineering analysis and integrated design must encompass the entire system. The requirements thus developed are substantially more than the interface definition and performance and functional requirements levied during traditional aerospace subsystem decomposition.

This is important because once subsystem decomposition occurs, subsequent analysis of system components provides no information about system controllability, stability or observability. That analysis must occur over the entire coupled set of subsystems, not over each of the component subsystems.

Lesson 3 - Interdependence Causes Complexity

The complexity of integrating the bioreactor control systems is not just an attribute of the biology, but also of the interdependence of the processes.

Process in this context refers to a transformation of something to something else. Processes have rates, control variables, and dependent variables. Interdependence means that changes in the parameters of one system necessitate changes in the controls of another system, either automatically (as in the case of the bioreactors) or by intent (a manual or autonomous command).

Conclusion

Although the particular types of modeling and analysis performed are specific to process control problems **it can be generalized that the sequence of modeling, analysis and test must be performed whenever there are dependencies between systems.**

Without these steps the risks associated with instabilities associated with system coupling remain unknown until after integration, sometimes until well after deployment.

These lessons can be applied to all autonomous and automated systems. The possibility of instability is one of the drivers which disallow automation of on-board systems except in exhaustively tested cases. Without proper modeling, analysis and testing, system dependencies may not be discovered until after

deployment. Automated control may enable those dependencies. Cross-system automation added after subsystem design will also generate dependencies between subsystems that were designed independently.

Importance

The Constellation Program has automation requirements different in kind from previous programs. The design of automated systems requires risk mitigation and engineering strategies different from previous programs. Subsystem requirements must be derived from integrated design, in contrast to deriving design from subsystem functional requirements and interface constraints.

References

1. "Lunar-Mars Life Support Test Project: Phase III Final Report," CTSD-ADV-341, Engineering Directorate Crew and Thermal Systems Division, 2000
2. "Bioastronautics Critical Path Roadmap (BCPR)," JSC 62577, 2004
3. "Investigation of Advanced Control Architecture Issues And Technology Gaps," CTSD-AIM-0027, Engineering Directorate Crew and Thermal Systems Division, 2004
4. "Final Report - Workshop on Advanced System Integration and Control for Life Support (ASICLS)," JPL document D-30410, 2005
5. Chairman of the Joint Chiefs of Staff (CJCS) Instruction 3170.01C and CJCS Manual 3170.01, both dated 24 June 2003
6. "NASA Systems Engineering Handbook," SP-6105, 1995
7. Bonasso, P., Kortenkamp, D., Thronesbery, C., "Intelligent control of a Water-Recovery System: three years in the trenches" *AI Magazine*, Spring, 2003
8. Hoo, K., Overland, D., "A Fundamental Mathematical Model of a Microbial Predenitrification System," Internal JSC publication, 2004
9. Zhang, X., Hoo, K., Overland, D., "Modeling and Control Studies of an Integrated Biological Wastewater Treatment System," 35th International Conference on Environmental Systems and 8th European Symposium on Space Environmental Control Systems, paper ICES2005-01-2963, Rome, Italy, 2005.
10. Ciskowski, M., Hoo, K., Overland, D., "Process Control for Advanced Life Support, Biological Test Bed," 36th International Conference on Environmental Systems, 06ICES-38, 2006

Definitions, Acronyms, Abbreviations

AIM: Advanced Integration Matrix

ALS: Advanced Life Support

ARS: Air Revitalization Subsystem

PID: Proportional Integral Derivative

DOD: Department of Defense

SRS: System Requirements Specification

IRD: Interface Requirements Document

WRS: Water Recovery System

JCIDS: Joint Capabilities Integration and Development System

JSC: Johnson Space Center