2009-01-2463

Development of Simulation Tool for Life Support System Design Based on the Interaction Model

Hiroyuki Miyajima
Tokyo Jogakkan College

ABSTRACT

In recent years, with increased size of the manned space program and systems used in the programs, the role of computer simulations has increased. I have long used and developed independently several simulation tools for the design, operation, analysis, and optimization of the Life Support Systems (LSS). I recognized that the designer makes his/her own idea certain while building a simulation model on a computer. However, conventional simulation tools are not designed so that the interaction between the designer and a model building support environment is dynamically used to bring out a designer's idea. Therefore, in this paper, I consider the development of a conceptual design support tool in the design of the LSS, while focusing attention on the interaction between the simulation tool and the designer.

INTRODUCTION

Since the launch of Vostok-1 spacecraft with Yuri Gagarin on board in 1961, various manned space programs have been conducted. In recent years, with increased size of the manned space program and systems used in the programs, the role of computer simulations has increased. Of the simulations, a simulation tool of a Life Support Systems (LSS) is the focus.

In the 1980s or later, simulation tools for the manned space program have increased with the development of computer technology. Cases of the analyses of Environmental Control and Life Support Systems (ECLSS) in Europe and the United States are provided in the literature [1, 2]. Especially, the use of simulations of the International Space Station (ISS) program has been described in detail in document [3]. In document [1], analysis technologies are classified into trade studies, process models, subsystem/system models; for the subsystem/system model, G-189A, SINDA'85/FLUINT, CASE/A, and TRASYS have been introduced as ECLSS analysis tools. The CASE/A developed in the 1980s includes a graphical user interface, which is nowadays commonly used, for the model construction and data management. Other than the cases described in documents [1, 2], Alan Drysdale has actively pursued research in modeling [4, 5] and system analysis [6, 7], and proposed Equivalent System Mass (ESM) as an index of the trade study. Furthermore, a trade study tool has recently been developed using MS-Excel, which is versatile spreadsheet software [8].

In the 1990s or later, there were the construction projects of ground experiment facilities for the Controlled Ecological Life Support Systems (CELSS) / Advanced Life Support Systems (ALSS), which material is more reproduced than the ECLSS. Representative examples are the Lunar Mars Life Support Test Project (LMLSTP) [9] and Bioregenerative Planetary Life Support Systems Test Complex (BIO-Plex) [10, 11, 12] of the United States, and the Closed Ecology Experiment Facilities (CEEF) [13, 14] of Japan. For the designs and operation of such facilities, studies using simulations were pursued. There are simulation studies for system design and optimization [15, 16], search method of design solution space [17], optimization frameworks [18], and conceptual designs [19].

I have long used and developed independently several simulation tools for the design, operation, analysis, and optimization of the CELSS/ALSS. Through experience gained in this work, I found the two points as follows. (1) The use of a conventional tool enables the development of a simulation model in a short time as compared with one's own development, but no product can be obtained which is more than what is given by a model building support environment and an analysis tool established on the basis of the conventional tool. (2) Alternatively, when making one's own development, the degree of freedom in modeling is large, but high programming technology and a considerable amount of time are required. As a secondary effect for the above, while debugging software, a programmer gains significant understanding of a target system and occasionally comes up with a new idea for the design in the process of the debugging work. That is, while building a simulation model on a computer, the designer makes his/her own idea certain. It is considered that this yields a great effect in a concept formation process on the conceptual design. However, conventional simulation tools are not designed so that the interaction between the designer and a model building support environment is dynamically used to bring out a designer's idea.

Therefore, in this paper, I consider the development of a conceptual design support tool in the design of the LSS, while focusing attention on the interaction between the simulation tool and the designer. In the following sections, I describe the role of a simulation in the development process, the use of the interaction to the development of the conceptual design support tool, the comparison between an operation system and a representation system of the simulation tool, and the development of the conceptual design support tool using an interaction model.

ROLE OF SIMULATION IN DEVELOPMENT PROCESS

First, I observe the role of computer simulation with reference to the development process of the NASA ECLSS. According to [1], this process consists of the following four phases: a concept study and preliminary analysis (Phase A), a preliminary definition and design (Phase B), and design/development and operations (Phase C/D). In Phase A, a concept study and a preliminary analysis are pursued to evaluate the feasibility of a program using a computer. In Phase B, the possibility of a preliminary concept created in Phase A, and a system requirement are compared using trade studies technology. Of Phases A to D, in Phases A and B, a simulation is used for the conceptual design.

Next, the role of a simulation on the LSS is viewed from two aspects: task and domain. In document [3], the task of system analysis software of the ISS ECLSS is divided into a plan, a concept design, engineering, construction, operation, and troubleshooting. In this study, the task of LSS development is divided into a plan, a design, production, operation, control, diagnostics, and training. I focus particular attention on the design.

As to a peripheral area for the LSS, there are a chemical plant, and a production process. These are represented as shown in Fig. 1. Versatile scheduling software becomes a task oriented tool. Furthermore, there are multiple pieces of software with which design, operation, and task are handled in a versatile manner. For example, Matlab/Simulink™ and LabVIEW™ are representative examples, which are applicable to the design and control of a system. For the analysis of the material circulation of the CEEF, I used WITNESS™ in the past, which is a combined discrete-continuous versatile simulation tool. This simulation tool, WITNESS, is applicable to production processes and chemical plants. CASE/A, which is used for the design of ISS, can be classified as a domain oriented-tool specialized for the LSS domain. Furthermore, there is also a software program that works with liberally with the ECLSS, while it is compatible with various domains of discrete-continuous systems as in EcosimPro developed at the European Space Agency (ESA) for the analysis of the ECLSS. All of the tool functions for system analysis and optimization, and for control system design work well, but the tools do not have sufficient functions to support the conceptual design with each one of which a sketchy mental model in one's brain at the stage of a conceptual design is transformed into a simulation model so that the conceptual design is supported. I focus attention not only on the analysis function of a simulation tool but also on the use of an emergence function thereof by interaction made with the designer.

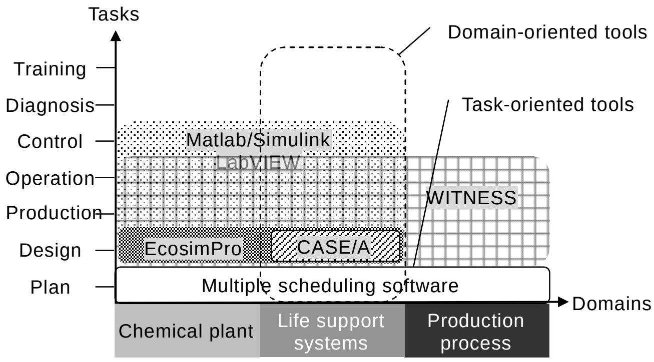


Fig. 1 Classification of simulation tools using task and domain

USE OF INTERACTION TO DEVELOPMENT OF CONCEPTUAL DESIGN SUPPORT TOOL

Figure 2 shows interactions between a tool and a designer in a conceptual design support using a simulation tool. Figure 2 is a diagram showing a flow in which, in the process of the development of an LSS model by the designer, the designer performs modeling or debugging while viewing the screen, or makes a change to parameters while viewing a calculation result. Here, I focus attention on an accelerated concept formation by the interaction, forming this loop, between the designer and a simulation tool.

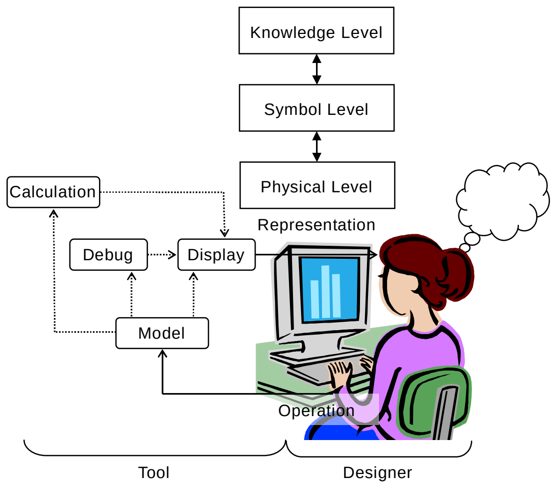


Fig. 2 Interaction between tool and designer in conceptual design support

An interaction design is used when developing an application tool to support the designer. The interaction design determines the representation and operation systems of the tool working from the perspective of the thought and action processes that the user experiences with the application tool. At this time, a model of thought and action is referred to as an interaction model. In order to develop an application tool enabling the designer to think without difficulty in the support of a creative activity, Nakakouji et al. cite the following four prerequisites for externalizing the interaction design [20]:

1. Representation system enabling representation of ambiguity;
2. Representation system enabling representation of solution and problem;
3. Representation system enabling simultaneous overview of halfway finished part and of whole coming to successful completion; and
4. Operation system enabling intuitive operation for the above.

Prerequisite 1 is a representation system enabling a sketchy flexible representation of a mental image that is not yet clear. Prerequisites 2 and 3 represent the creation of the whole in combination of a part (solution) already completed, and a part (problem) not yet completed. These works are performed in both processes from part to whole (bottom-up), and from whole to part (top-down). Repetition of these works clarifies the halfway finished part while simultaneously providing an overview of the halfway finished part and of the whole coming to a successful completion. Prerequisite 4 is an operation system allowing the designer to feel the operation of a representation, not the operation of a software application.

Hori [21] considers that the level of information in the support of the creative activity is established by a hierarchical architecture (KSP hierarchy) formed of a physical level, a symbol level, and a knowledge level [22]. Here, Hori considers that when changing an architecture at any one of the physical, symbol, and knowledge levels, a change at another level occurs. Consequently, the creative activity can be supported. For example, the generally accepted belief is that in design, a sketched work brings out one's creativity more than CAD-based work. This means, at the physical level, the use of a pencil and a piece of paper are more productive than the use of a computer and a mouse.

In this study, the KSP hierarchy is used for the definition of interactions from the designer to a software application (externalized operation), and from a software application to the designer (externalized representation). At this time, what is most important is how the concepts of the symbol and knowledge levels are represented. In the conceptual design, a system is usually represented using functions. Since a function has a close connection with one's purpose and intention when understanding a system, the designer naturally assumes that the system represents a functional hierarchy in which various devices operate in cooperation with each other to achieve a specific purpose. From this viewpoint, Vicente and Rasmussen represent a target object by a Means-Ends relationship, in five-hierarchy architecture, of "Functional purpose, Abstract function, Generalized function, Physical function, and Physical form [23]." Meanwhile, for the architecture of the system, there is also another system hierarchy using a connection relationship between the whole and part. The assumption is that there is strong correlation between a functional hierarchy and a system hierarchy, which are conceptually different from each other, as in the case where a device achieves a specific physical function. Rasmussen represents this correlation in a two-dimensional space of Fig. 3, and refers to the same as Abstraction Hierarchy (AH). This space provides a mental model useful to the designer performing a conceptual design for the system. At the same time it provides a key important to the description of a relationship between the symbol level and the knowledge level. Here, I assume that the symbol level corresponds to the system hierarchy, and the knowledge level corresponds to the functional hierarchy.

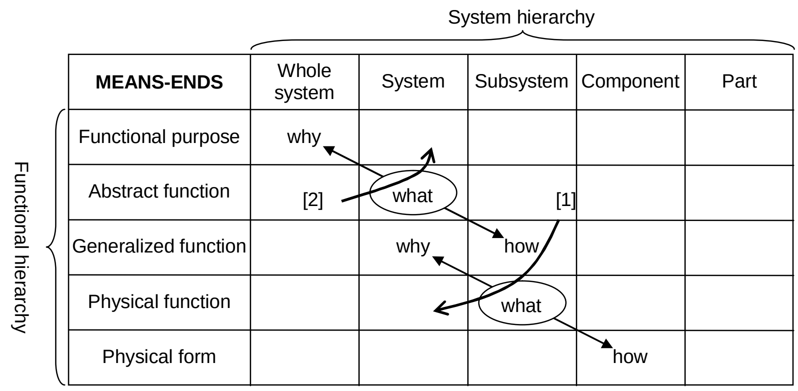


Fig. 3 Abstraction Hierarchy of system

In the meantime, in the support of the creative activity, it is important to consider the functional hierarchy from the system hierarchy. A transformation (arrow [2]) from the system hierarchy to the functional hierarchy represents an analysis, and a transformation (arrow [1]) from the functional hierarchy to the system hierarchy represents a synthesis. That is, a conceptual design process by the designer is considered to correspond to a work in which a mental model is actualized while making the round of the space.

Therefore, in this tool, for the operation of the knowledge level, a function is implemented, with which the representation and formation of a concept are made using the relationship between the functional hierarchy and the system hierarchy.

COMPARISON BETWEEN OPERATION SYSTEM AND REPRESENTATION SYSTEM OF SIMULATION TOOL

In this section, prior to the description of the actual development, a comparison is made with respect to operation systems and representation systems of three simulation tools, i.e., Simulink, EcosimPro, and WITNESS. For each one of the KSP, comparison items are classified and shown in Table 1.

Table 1 Comparison between operation system and representation system

Externalization	Evaluation Items	Simulink	EcosimPro	WITNESS
Operation system	**Knowledge Level**			
	Conceptual creation support	✓ Logic circuit knowledge	✓ Model element attributes	✓ Model element attributes
	Symbol Level			
	GUI base modeling	✓ Logic circuit icon	✓ Device icon	✓ Device icon
	CUI base programming	✓	✓	✓
	LSS library		✓	
	Calculation solver	✓	✓	
	Physical Level			
	User-intuitive	✓ Mouse and key board	✓ Mouse and key board	✓ Mouse and key board
Representation system	**Knowledge Level**			
	Display using a table or graph	✓	✓	✓
	Symbol Level			
	Display of whole-part hierarchy			✓
	Display of relationship between parts	✓	✓	✓
	Display of part attributes	✓	✓	✓
	Physical Level			
	Animation			✓

The operation system represents the flow of information from the designer to the simulation tool shown in Fig. 2. In this process, the designer operates the simulation tool at the knowledge, symbol, and physical levels for the representation of a mental model in the designer's brain. At the knowledge level, knowledge on a design target is naturally important; for Simulink, the frame of a logic circuit supports a concept formation. For EcosimPro and WITNESS, an attribute of a model element of the tool supports the concept formation. At the symbol level, all the software programs have GUI base modeling and CUI base programming functions, and only EcosimPro has library, and Simulink and EcosimPro have a computational solver. At the physical level, main operations are performed using a mouse and keyboard.

Next, the representation system represents a flow of information from the simulation tool to the designer shown in Fig. 2. In this process, the mental model in the designer's brain is represented at the knowledge, symbol, and physical levels using the simulation tool. At the knowledge level, a display using a table or graph is not only considered as data but also considered as knowledge having some meaning. This is the function possessed by the three software programs. At the symbol level, a display function of whole-part hierarchy, a display function of relationship between parts, and a display function of part attributes are important. The whole-part hierarchy display function is quite important for the representation of a large-scale system, and the author considers in particular that replacing of elements by a drag and drop operation expedites the concept formation. Moreover, the display function of the relationship between parts is important for checking the flow between parts. The display function of part attribute enables thought on the entire configuration, with details of parts being left ambiguous. At the physical level, an animation function such as one of WITNESS is important.

The belief is that the designer's creativity is enhanced when visually enabling performance of the cycle of modeling, performance, and result display, and when enabling check of a result as a phenomenon using animation. Therefore, different effects other than those of tables and graphs can be expected.

DEVELOPMENT OF CONCEPTUAL DESIGN SUPPORT TOOL USING INTERACTION MODEL

An interaction model of the conceptual design support tool is designed, as shown in Fig. 4, based on the requirements of externalizing, and analysis of an interaction. The relationship between parts is built while, on Model Development Workspace (A1), checking the flow between parts based on Subsystem List Viewer (A2) and Function List Viewer (A3). In the middle of the work, change is made to the relationship between parts while, on Element Tree Viewer (B), checking the role of a part on the whole. Details of a part are set on Model Attribute Viewer (C), and at the same time the role of the part of the whole is again checked in depth on Model Attribute List Viewer (A4). Subsequently, the result is checked on Graph Viewer (D), and changed if necessary on Model Attribute Viewer (C). The concept is formed while the simulation tool and the designer repeat such interactions.

The specific operation system and representation system at the physical, symbol, and knowledge levels are described below. First, as an operation system at the physical level, a drawing function using a pen tablet is implemented to allow the designer to intuitively perform modeling on Model Development Workspace. In the operation system, animation and rewind functions are considered to be useful, but are shelved this time in view of the difficulty of their development. Furthermore, at the symbol level, Model Development Workspace, Element Tree Viewer, Model Attribute Viewer, and Model Attribute List Viewer are implemented, so that modeling becomes possible while simultaneously having an overview of the halfway finished part and of the whole coming to successful completion. Especially, for the above reason, it seems that Model Attribute List Viewer is the most necessary function in the past developments. In addition, at the knowledge level, a display of the analysis of the system and functional hierarchies expedites the designer's thought.

In Fig. 5, the present tool is shown on the left side, and another tool under development is shown on the right side. The present development is made based on the ALS scheduler developed for the support of a habitation experiment of the CEEF. The ALS scheduler has a function with which a model is represented using a simple Planning and Scheduling Language (PSL), and a function with which the material circulation of the LSS is simulated in accordance with an implemented algorithm. The tool under development is described in the Fig. 5 footnote.

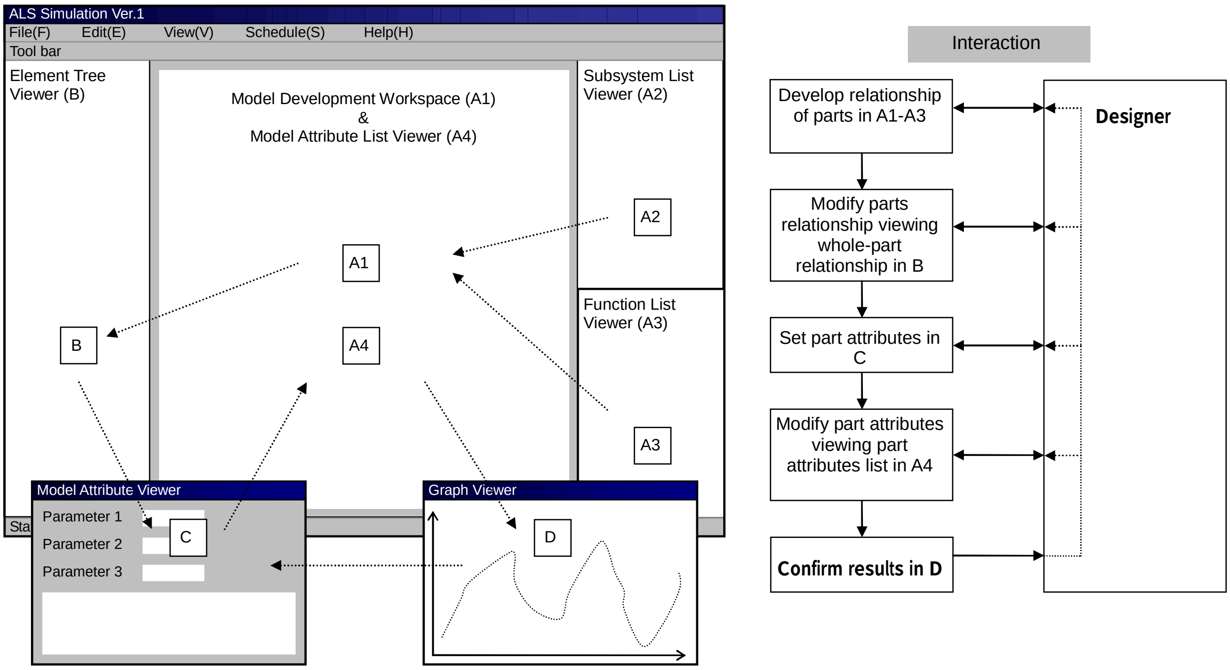
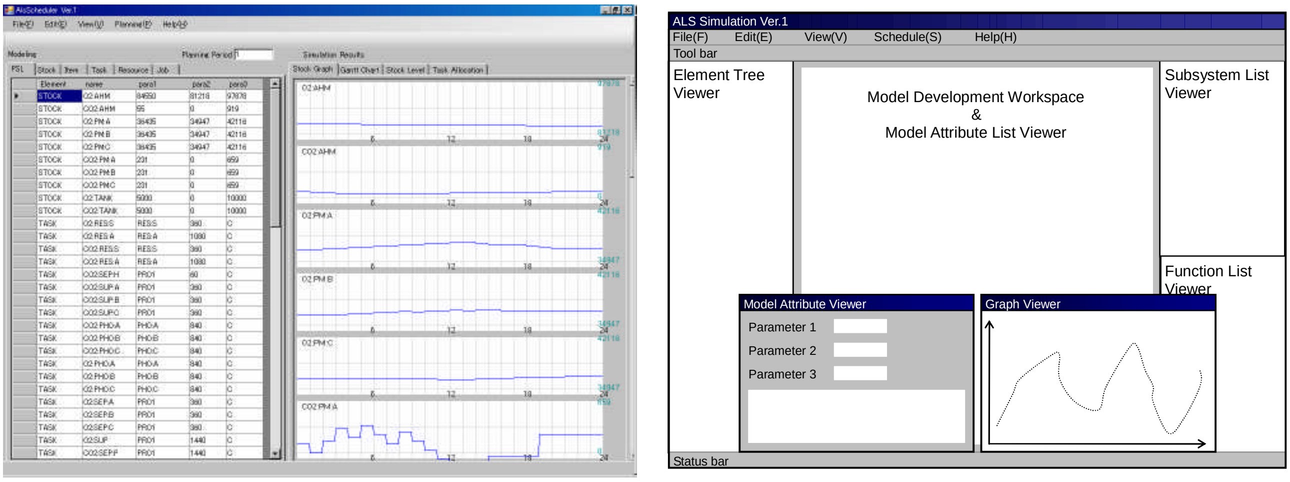


Fig. 4 Overview of conceptual design support tool



On Model Development Workspace, a design target is drawn with a pen using icons for devices selected from the Subsystem List. The devices each have an attribute on detailed parameter and the relationship between input and output. These are set on Model Attribute Viewer. The whole and a part can be checked on Element Tree Viewer while making a model. At the same time, model attribute can also be changed on this screen. Computation result can be displayed on Graph Viewer. Model Development Workspace can be changed to Model Attribute List Viewer. Details of all the model attributes can be displayed on a list.

Fig. 5 Outline of developed system

CONCLUSION

In this paper, the role of the simulation tool of the LSS in a manned space program was outlined, and it was also pointed out that the concept formation function for the conceptual design is poor. To install this function, it is desirable that development be conducted in light of the interaction between the designer and the simulation tool. Next, comparison was made to the operation and representation systems of the three typical versatile simulation tools in terms of the interaction design so as to analyze the functions of the simulation tool necessary for the support of a creative conceptual design. Based on the analysis, the interaction model was created, and the development outline was shown, in which the ALS scheduler is modified to a tool placing importance on the extraction of the designer’s mental model by interactions. In the future, I will discuss the effects and use of this tool in the conceptual design through development of tools and design experiments conducted by a designer.

REFERENCES

1. Paul O. Wieland, Designing for Human Presence in Space: An Introduction to Environmental Control and Life Support Systems, NASA RP-1324, 1994.
2. Peter Eckart, Spaceflight Life Support and Biospherics, Microcosm, 1997.
3. Richard R. Chu, ISS ECLS System Analysis Software Tools - An Overview and Assessment, SAE Technical Paper Series 2002-01-2343, 2002.
4. Alan Drysdale, Mark Thomas, Mark Fresa, and Ray Wheeler, OCAM - A CELSS Modeling Tool : Description and Results, SAE Technical Paper Series 921241, 1992.
5. Alan Drysdale, OCAM-2: A Second Generation Bioregerative Life Support System Model, SAE Technical Paper Series 972291, 1997.
6. Alan Drysdale and Anthony J. Hanford, Advanced Life Support Research and Technology Development Metric - Baseline, CTSD-AD-384, JSC 39503, 1999.
7. Alan Drysdale, Mike Ewert, Anthony J. Hanford, Equivalent System Mass Studies of Missions and Concepts, SAE Technical Paper Series 1999-01-2081, 1999.
8. H. Y. Yeh, Cheryl B. Brown, Frank F. Jeng, Chin h. Lin, and Michael K. Ewert, ALSSAT Development Status and Its Applications in Trade Studies, SAE Technical Paper Series 2004-01-2438, 2004.
9. Jane T. Malin, Enhancing Discrete Event Simulation by Integrating Continuous Models, AAAI Technical Report SS-99-05, 1999.
10. Cory K. Finn, Dynamic System Modeling of Regenerative Life Support Systems, SAE Technical Paper Series 1999-01-2040, 1999.
11. Cory K. Finn, Karen E. Meyers, and Bruce Duffield, Dynamic Model of the BIO-Plex Air Revitalization System, SAE Technical Paper Series 2001-01-2318, 2001.
12. Harry W. Jones, Cory K. Finn, Xianmin Kwauk, and Charles A. Blackwell, Modeling Separate and Combined Atmospheres in BIO-Plex, SAE Technical Paper Series 2001-01-2361, 2001.
13. Hiroyuki Miyajima, Yoshio Ishikawa, Akira Ashida and Keiji Nitta, Development of Simulation Model and its Application to an Integration Test Project of CEEF, 2000-01-2334, 2000.

14. Hiroyuki Miyajima, Yoshio Ishikawa, Ryuji Arai, Yasuhiro Tako, and Keiji Nitta, Considerations of Material Circulation in CEEF Based on the Recent Operation Strategy, SAE Technical Paper Series 2003-01-2453, 2003.
15. M. Czupalla, T. Dirlich, S. I. Bartsev, An Approach to LSS Optimization Based on Equivalent System Mass, System Stability and Mission Success, SAE Technical Paper Series 2007-01-3222, 2007.
16. Michael Lasinski et al, Application of Supply Chain Optimization and Protocol Environment Architecture to ALS Modeling and Visualization of a Mars Surface Habitat, SAE Technical Paper Series 2006-01-2188, 2006.
17. Scott Bell, Luis Rodriguez, and David Kortenkamp, Using Dynamic Simulation and Automated Decision Tools to Design Lunar Habitats, SAE Technical Paper Series 2005-01-3011, 2005.
18. Selen Aydogan, Joseph Pekny, Gintaras Reklaitis, Seza Orcun, and Gary Blau, An Optimization Framework to Design an Integrated ALSS, SAE Technical Paper Series 2006-01-2246, 2006.
19. Jan Osburg, and Ernst Messerschmid, Integrated Simulation of Synergistic Space Station Subsystems During the Conceptual Design Phase, SAE Technical Paper Series 2000-01-2398, 2000.
20. Kumiko Nakakoji and Yasuhiro Yamamoto, Knowledge Interaction Design for Creative Knowledge Work, Transactions of the Japanese Society for Artificial Intelligence, Vol. 19 No. 2, 2004.
21. Koichi Hori, Creativity Support - Theory and Application of AI Approach toward Creativity -, Tokyo, Ohmsha, 2007.
22. Allen Newell, The Knowledge level, Artificial Intelligence, Vol. 18, pp. 87-127, 1982.
23. Jens Rasmussen, Annelise Mark Pejtersen, and L. P. Goodstein, Cognitive Systems Engineering, New York, John Wiley & Sons, 1994.
24. Rámon Pérez, Sergio Mannu, Olivier Pin, and Roland Müller, Overview of European Applications of EcosimPro to ECLSS, CELSS, and ATCS, SAE Technical Paper Series 2003-01-2439, 2003.

CONTACT

Hiroyuki Miyajima
Tokyo Jogakkan College
1105 Tsuruma, Machida-shi, Tokyo 194-0004 Japan
E-mail: miyajima@m.tjk.ac.jp

DEFINITIONS, ACRONYMS, ABBREVIATIONS

AH: Abstraction Hierarchy

ALSS: Advanced Life Support Systems

BIO-Plex:Bioregenerative Planetary Life Support Systems Test Complex

CEEF: Closed Ecology Experiment Facilities

CELSS:Controlled Ecological Life Support Systems

ECLSS:Environmental Control and Life Support Systems

ESA: European Space Agency

ISS: International Space Station

KSP: Physical, Symbol, and Knowledge

LMLSTP:Lunar Mars Life Support Test Project

LSS : Life Support Systems

PSL: Planning and Scheduling Language