ICES-2017-82

Axiomatic Design of Space Life Support Systems

Harry W. Jones[1]
NASA Ames Research Center, Moffett Field, CA, 94035-0001

Systems engineering is an organized way to design and develop systems, but the initial system design concepts are usually seen as the products of unexplained but highly creative intuition. Axiomatic design is a mathematical approach to produce and compare system architectures. The two axioms are:

- **Maintain the independence of the functional requirements.**
- **Minimize the information content (or complexity) of the design.**

The first axiom generates good system design structures and the second axiom ranks them. The closed system human life support architecture now implemented in the International Space Station has been essentially unchanged for fifty years. In contrast, brief missions such as Apollo and Shuttle have used open loop life support. As mission length increases, greater system closure and increased recycling become more cost-effective. Closure can be gradually increased, first recycling humidity condensate, then hygiene waste water, urine, carbon dioxide, and water recovery brine. A long term space station or planetary base could implement nearly full closure, including food production. Dynamic systems theory supports the axioms by showing that fewer requirements, fewer subsystems, and fewer interconnections all increase system stability. If systems are too complex and interconnected, reliability is reduced and operations and maintenance become more difficult. Using axiomatic design shows how the mission duration and other requirements determine the best life support system design including the degree of closure.

Nomenclature

DP = Design Parameter
DSM = Design Structure Matrix
EVA = ExtraVehicular Activity
FR = Functional Requirement
KISS = Keep it simple, stupid
QFD = Quality Function Deployment

I. Introduction

THIS paper applies the axiomatic design method to space life support systems. Axiomatic design uses a coupling matrix to control the relations between requirements and subsystem functions. Using an axiomatic mathematical model to develop a systems concept is a surprising endeavor. It is usually assumed that developing a system concept and preliminary design requires “great creativity.” (Recthtin, 1991) The creative design process is poorly understood and seems to be intuitive, even visionary. A suggested design concept mysteriously appears to put an end to the confusion of requirements, stakeholder constraints, and technical uncertainty. There is no accepted methodology for system concept development. According to *The Art of Systems Architecting*, it requires great talent, knowledge, and judgment. (Rechtin and Maier, 1997)

This intuitive understanding of systems concept development is usual in the United States and the United Kingdom. However, Europeans favor “a systematic generation of solutions using various levels of abstraction.” (Wallace and Blessing, 2000) The most widely used process is the “Systematic Approach” developed by Pahl and Beitz, which was first published in German in 1977 and frequently extended and republished. (Pahl and Beitz, 2007) It provides detailed operational guidelines for the entire top-down engineering process. It is widely used in engineering design and education, but not always completely implemented. (Kannengeisser and Gero, 2014)

---

[1] Systems Engineer, Bioengineering Branch, Mail Stop N239-8.

The European systematic approach of Pahl and Beitz has a sequence of four phases: (1) task clarification, (2) conceptual design, (3) embodiment design, and (4) detail design. Each phase has many steps. Clarification involves the development of the requirements and a system specification. Conceptual design identifies the basic principles leading to the best design concept. This conceptual design phase identifies the the essential problems, establishes a function and subfunction structure, finds working principles to fulfil the subfunctions, combines these working principles into working structures, evaluates the various combined solutions, and selects the best design concept. Most of the Pahl and Beitz systematic approach concentrates on the next phase, embodiment of the design. (O'Shaughnessy and Sturges, 1991)

The American intuitive approach and the European systematic approach both theoretically begin by defining the requirements without considering or constraining the conceptual design. This idealistic attempt rarely succeeds in practice. “(W)e rarely know enough to write the requirements without exploring concepts, building models and prototypes, or performing analysis and trade studies.” (Hooks and Farry, 2001) The supposedly pure requirements development later becomes design by stealth. The conceptual design rabbit is first hidden in the requirements hat, and then pulled out to general amazement by the visionary and inspired systems architect. In reality, “iteration between requirement definition and design is inevitable.” (Hooks and Farry, 2001) In contrast, the axiomatic design approach does not try to develop the requirements before and independently of the design concept, but rather develops requirements and design together in a top-down process. This is fundamental to achieving a good design concept.

II. Axiomatic design

Axiomatic design theory was developed by Suh at MIT in 1990 and has been extended and republished. (Suh, 1990) (Suh, 2001) (Suh, 2005) The axiomatic design approach is described, including the mapping of requirements to design concept. The axioms are stated and their implementation using a design matrix is described.

A. The axiomatic design approach

Axiomatic design uses matrix methods to analyze the transformation of functional requirements (FRs) into design parameters (DPs). This is shown in Figure 1.

$$\begin{bmatrix} FR1 \\ FR2 \end{bmatrix} = \begin{bmatrix} A11 & A12 \\ A21 & A22 \end{bmatrix} x \begin{bmatrix} DP1 \\ DP2 \end{bmatrix}$$

Figure 1. Functional requirements (FRs), design parameters (DPs), and the design or coupling matrix A.

A functional requirement (FR) is what we want to achieve, what the system must satisfy. A design parameter (DP) is how the FRs will be achieved, the key variables that characterize design solution.

The design matrix A indicates that the functional requirement FR1 is satisfied by a combination of design parameters DP1 and DP2. FR1 = A11 DP1 + A12 DP2, and similarly for FR2.

The two axioms of axiomatic design are formally stated as:

Axiom 1: The Independence Axiom. Maintain the independence of the functional requirements (FRs).

Axiom 2: The Information Axiom. Minimize the information content (complexity) of the design. (Suh, 2005)

The kind and degree of mutual dependency between the FRs is determined by the functional coefficients Aij of the design matrix A. Maximum independence of FRi is achieved if Aii = 1 and all other Aij = 0.

B. Top-down mapping of requirements to design concepts

In Figure 1 the functional requirements (FRs) are related to design parameters (DPs) at a single level, but much of the power of axiomatic design is gained by mapping requirements to design first at the top level and then at successively lower levels, in an iterative process. Figure 2 shows the FR decomposition process of developing detailed requirements and concepts by moving back and forth, zigzagging, between the functional and physical domain.

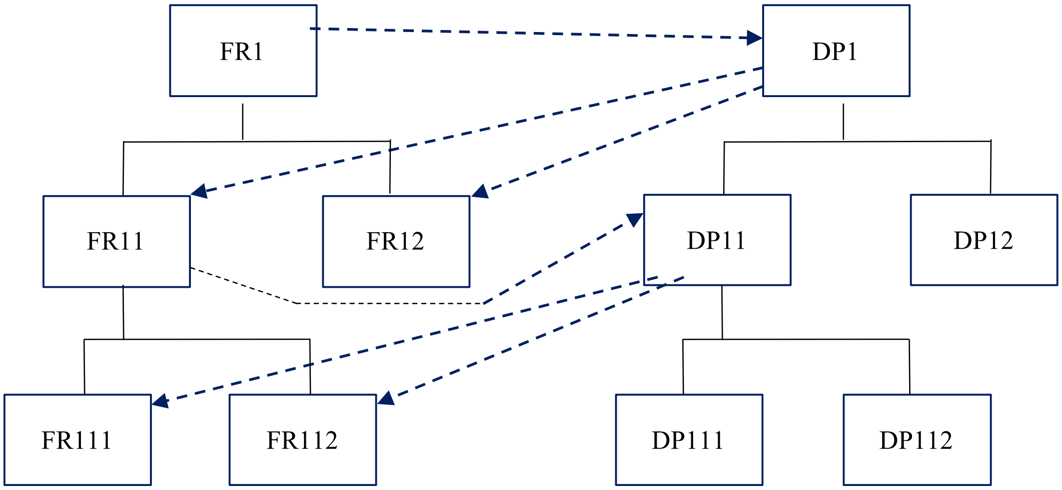

Figure 2. Decomposing FRs and DPs by zigzagging between successively lower levels.

The axiomatic design process decomposes the highest-level design to develop lower level design details that can be implemented. To decompose the FR and DP one dimensional matrix vectors, we must zigzag between the requirements and design domains. This is illustrated in Figure 2. From FR1 in the functional domain, we go to the physical domain to conceptualize a design and determine its corresponding DP1. Then the process comes back to the functional domain to create FR11 and FR12 at the next level down. Together FR11 and FR12 satisfy the highest level FR1. FR11 and FR12 are the FRs for the highest-level DP1. Then in the physical domain, DP11 is found to satisfy FR11. It in turn is used to create FR111 and FR112 at the third level. The process of decomposition is continued until the highest-level FR can be satisfied without further decomposition.

To analyze the design decision, the design equation FR = A x DP is examined at each level of decomposition. For example, in Figure 2, after FR1 and DP1 are decomposed into FR11, FR12 and DP11, DP12, the design equation describes the design concept at this level. At the higher levels of the process, the concept lacks detail, but the design matrix can be examined to see how well it satisfies the first axiom, independence. (Suh, 2005)

C. Design matrix coupling

A design is described as uncoupled, decoupled, or coupled, according to the pattern of zero and nonzero entries in the design matrix. According to the independence axiom, an uncoupled design is best and a decoupled design is less good, while a coupled design is the least satisfactory. (Suh, 2005)

Figure 3 shows the design matrix of an uncoupled design.

$$\begin{array}{|c|}\hline FR1 \\ \hline FR2 \\ \hline\end{array} = \begin{array}{|c|c|}\hline X & O \\ \hline O & X \\ \hline\end{array} \times \begin{array}{|c|}\hline DP1 \\ \hline DP2 \\ \hline\end{array}$$

Figure 3. An uncoupled design matrix, diagonal entries only.

An uncoupled design is described by a diagonal design matrix. Each of the FRs is satisfied by a single DP without being affected by any other DP. Figure 4 shows a decoupled design.

$$\begin{array}{|c|}\hline FR1 \\ \hline FR2 \\ \hline\end{array} = \begin{array}{|c|c|}\hline X & O \\ \hline X & X \\ \hline\end{array} \times \begin{array}{|c|}\hline DP1 \\ \hline DP2 \\ \hline\end{array}$$

Figure 4. A decoupled design matrix, all entries on or below the diagonal.

In this decoupled but not fully uncoupled design matrix, FR1 is satisfied by DP1, but FR2 is affected by DP1 even though it may be largely satisfied by DP2. In the design process, DP1 can be designed independently to satisfy FR1 and then DP2 can be designed to satisfy FR2 while also considering the effect of DP1. Independence is desirable and coupling is to be avoided because any change in the design or operation of DP1 will affect the performance of DP2 as well as of DP1. Also, any problems in designing or operating DP2 may force changes in DP1 that would make it less optimal in meeting FR1. Figure 5 shows a coupled design.

$$\begin{bmatrix} FR1 \\ FR2 \end{bmatrix} = \begin{bmatrix} X & X \\ X & X \end{bmatrix} x \begin{bmatrix} DP1 \\ DP2 \end{bmatrix}$$

Figure 5. Coupled design.

The design in Figure 5 is fully coupled. Even though DP1 can be designed largely to meet FR1, and similarly for DP2, both DPs affect both FRs and the design process must balance their interactions. In the extreme worst case of a fully coupled design matrix, any change in the design or operation of any of the DPs will affect all the FRs. Any later adjustment or failure will perturb the entire system. A fully independent design characterized by an entirely uncoupled design matrix would be best. Meeting the independence axiom requires maintaining the independence of the FRs. In the ideal case, the design matrix is square and the number of DPs equals the number of FRs. A good design must be either uncoupled or decoupled, and therefore, the intended design must have either a diagonal or a lower triangular pattern of entries. (Suh, 2005) If the design matrix is not square, the number of DPs is more or less than the number of FRs. If the DPs are fewer than the FRs, the design must necessarily be coupled. If there are more DPs than FRs, the design can be uncoupled, decoupled, coupled or redundant. (Suh, 2005) A few complex multiple purpose DPs can cause problems but redundant DPs can be useful.

III. Discussion of the axiomatic design approach

Axiomatic design is claimed to be better than design decisions based on experience acquired by trial-and-error optimization. The plausibility and justification of the axiomatic design approach is considered.

There are several key ideas in the axiomatic design approach. These include:

A. The axiomatic approach
B. Justification for Axiom 1, maintain the independence of the functional requirements
C. Matrix design methods
D. Define the requirements and design top-down, together, level by level
E. Make the number of design parameters (DPs) equal to the number of functional requirements (FRs)
F. The best design minimizes the information content (complexity) of the design (Axiom 2)

A. The axiomatic approach

Suh states that, “Axioms … are truths that cannot be derived but for which there are no counter-examples or exceptions. … Design axioms are presumed to be valid if they lead to better designs that satisfy the functional requirements and are more reliable and robust at low cost. … The design axioms were created by identifying common elements that are present in all good designs.” (Suh, 2005)

Designers are asked to accept the algorithmic approach on faith. There is no counter proof that the independence axiom does not produce better designs, but most designs are usually coupled to some extent. Many mature and evolved designs are strongly coupled and are considered good designs by engineers and operators. A logical rationale for the axioms would be preferred.

B. Justification for Axiom 1, maintain the independence of the functional requirements

Axiom 1 is: Reduce system complexity by maintaining the independence of the requirements. This seems intuitively correct and even very familiar. It is generally understood that reducing interconnections and dependencies can help improve reliability and reduce integration and test problems.

Many engineers are taught the KISS principle. “KISS is an acronym for "Keep it simple, stupid" as a design principle noted by the U.S. Navy in 1960. The KISS principle states that most systems work best if they are kept simple rather than made complicated; therefore simplicity should be a key goal in design and unnecessary complexity should be avoided.” (Wikipedia, KISS principle)

Systems engineering is more explicit. “(A) separation of a system into noninteracting subsystems is an extremely important technique known to all developed sciences – and to systems theorists as well.” (Weinberg, 1975) “In partitioning, choose the elements so that they are as independent as possible.” (Rechtin, 1991)

C. Matrix design methods

The Design Structure Matrix (DSM) uses matrices in design. The DSM relates the systems elements such as hardware subsystems to each other. The subsystem names are placed down the side of the matrix as row headings and across the top as column headings in the same order. An all ones diagonal indicates the identity of the row and column headings. The relations between the subsystems are uncoupled, decoupled, or coupled, as in axiomatic design. The objective of system design is to create subsets of DSM elements, clusters, that are not interacting or only minimally interacting. The objective is to maximize interactions between elements within clusters while minimizing interactions between clusters. Minimizing the size of the clusters has also been suggested. Clustering can be considered an art, but algorithms and heuristics can be used. After clustering, the clusters contain most, if not all, of the interactions. (Yassine, 2004) (Browning, 2001) Minimal interaction of the subsystems is desired but complete uncoupling is not intended. The DSM has been used in life support analysis. (Do and de Weck, 2014-118) (Perry et al., 2106-90)

Axiomatic design has been combined with the DSM. The axiomatic design matrix maps the FRs to the DPs while the DSM is used to structure the detailed system development. Alternating between the two matrices allows the definition of the system architecture to interact with subsystem and interface design. (Guenov and Barker, 2005)

Quality Function Deployment (QFD) often uses the house of quality tool which contains a matrix relating requirements to engineering characteristics. QFD uses a systematic iterative mapping process. The central matrix is the body of the house of quality and is similar to the matrix used in axiomatic design. This planning matrix relates customer needs, the “Whats,” to the product system capabilities, the “Hows.” The entries in the matrix indicate whether the interaction of the specific item is a strong positive, a strong negative, or somewhere in between. The intent is more to discover and understand the interactions rather than to reduce or eliminate them. As with the DMS, QFD is based on an understanding that a system must have a hierarchy of interconnected subsystems. Eliminating interactions is not the intent of QFD. As with axiomatic design, house of quality analysis can also be cascaded, with "Hows" from one level becoming the "Whats" of the next lower level. (Wikipedia, House of Quality) The house of quality has been used in life support analysis. (Lee, 2014-178)

D. Define the requirements and design top-down, together, level by level

The top-down design process is very well known. “The whole idea of systems engineering is the implementation of a top-down process.” (Liu, 2016) The requirements are initially defined at the top level and then are expanded top-down. The requirements are usually implemented in a hierarchy of system, subsystems, and components.

“Broadly defined, system engineering is the effective application of scientific and engineering efforts to transform an operational need into a defined system configuration through the top-down iterative process of requirements definition, functional analysis, synthesis, optimization, design, test, and evaluation.” (Blanchard, 1991)

What is different in axiomatic design is that, instead of the complete requirements tree being developed before design and then the subsystems being designed to the final requirements, the top level and intermediate level design implementations are considered at the same time as the requirements for those levels are developed. This is the zigzag process of Figure 2, going back and forth between design and requirements, level by level. The objective is to keep the FRs and DPs as independent as possible, in accordance with Axiom 1.

E. Make the number of design parameters (DPs) equal to the number of functional requirements (FRs)

The objective of making the number of DPs exactly equal to the number of FRs is to allow uncoupled design, which is best according to Axiom 1. Having a larger number of DPs could allow them to be in groups each independently satisfying one of the FRs, which would satisfy Axiom 1. At the lowest level of design, many components are needed to satisfy an FR, but they can do so independently of other FRs. An uncoupled or decoupled design cannot always be achieved.

F. The best design minimize the information content (complexity) of the design (Axiom 2)

Suh’s Axiom 2 is simply, “Minimize the information content of the design.” The information axiom is used to select the best design. Suh defines information using the familiar bits of data communication and storage, but it is computed based on the estimated probability that all the FRs will be satisfied. A design is considered complex if its probability of success is low. (Suh, 2005) Estimating the probability of success seems difficult and subjective.

"Axiom 2: Minimize Information Content is difficult to understand and apply. There are many approaches to interpreting Axiom 2. Some designers use it to mean complexity of parts, others use it to mean reliability of parts, still others have considered it to refer to the ability to maintain the tolerances on parts. Axiom 2 has not been used by the design community as much as Axiom 1, leading to questions about its usefulness, or about the axiomatic approach in general." (Engineering Design)

It is suggested instead that alternate systems produced using Axiom 1 be compared using the systems engineering trade-off approach. Performance, cost, reliability and many other factors should be considered. Axiomatic design assumes that satisfying Axiom 1 should have higher priority than other systems considerations.

IV. Axiomatic design of life support

The two basic systems architectures used in space life support are well known. Brief missions in low Earth orbit such as Apollo and shuttle have used open loop life support, with atmosphere gas, water, and food directly supplied rather than recycled and with carbon dioxide removed by lithium hydroxide (LiOH) rather than by a regenerative chemical processor. The closed atmosphere and water systems architecture now used on the space station is similar to the recycling systems architecture first developed in the 1960's. The top level FRs and DPs are similar for both direct supply and recycling systems, but the coupling between systems becomes stronger as the life support closure increases.

A. Level 1 and 2 life support requirements

The axiomatic design process will be used to consider the design of space life support. The zigzag design process of Figure 2 is described using a table that shows the top-down process of expanding the level 1 requirement into level 2 requirements and the system implementation of the different requirements. A table shows the process better than going between one tree for the requirements and another identical tree for the systems. Table 1 shows the two top levels of space life support requirements and systems.

Table 1. Life support level 1 requirements and systems

Level 1	Requirement	FR1: Support human life in space				
	System	DP1: Life support system				
Level 2	Requirement	FR11: Provide atmosphere	FR12: Provide Water	FR13: Handle waste	FR14: Suppress fire	FR15: Provide food
	System	DP11: Atmosphere system	DP12: Water system	DP13: Waste system	DP14: Fire system	DP15: Food system

There is one level 1 requirement, to support human life in space. This requirement is allocated to one level 1 system, the life support system. The level 1 requirement is partitioned into five level 2 requirements. These are derived from and directly traceable to the level 1 requirement. The five level 2 requirements are to provide atmosphere, provide water, handle waste, suppress fire, and provide food. The five level 2 requirements are allocated to five level 2 systems, the atmosphere, water, waste, fire, and food systems.

A key method of the axiomatic design approach is to match each requirement with its design implementation at each stage of the top-down elaboration of the requirements. The lower level requirements are more specific and detailed. This approach is a deliberate direct contrast to the usual method of creating a detailed multilevel requirements tree, freezing it, and then developing the hardware design to meet that set of detailed requirements. It is supposed, but rarely happens, that the requirements are developed without assuming some system design.

While it helps in clarifying requirements, the main purpose of going back and forth between requirements and systems in the axiomatic design approach is to ensure maximum decoupling of each requirement from the systems implementing other requirements. The extent of decoupling obtainable can be limited by the environment and the available hardware systems.

B. The coupling of the level 2 life support requirements and systems

The five life support requirements and the five life support systems are represented by five by one matrices. They are related to each other by a five by five coupling matrix. This is shown in Figure 6.

FRs - requirements								DPs - systems
FR11: Provide atmosphere		X	*Provides water?*	*Provides humidity?*	De- and re-pressurize			DP11: Atmosphere system
FR12: Provide Water		*Provides condensate?*	X	Needs water *Provides water?*		*Needs water?* *Provides water?*		DP12: Water system
FR13: Handle waste	=		Needs water *Provides water?*	X		Food and packaging waste	x	DP13: Waste system
FR14: Suppress fire					X			DP14: Fire system
FR15: Provide food			*Water balance?*			X		DP15: Food system

Figure 6. Life support requirements, systems, and coupling matrix. (Closed system couplings are shown in italics.)

The purpose of DP11: Atmosphere is to meet the specific atmosphere maintenance requirement FR11, and similarly for the other systems. These direct requirement-to-system relations are indicated by the X's on the main diagonal of the five by five matrix. If the requirements and systems are completely uncoupled, as is the ideal case, the five by five matrix would have only the the diagonal X's and no other entries. As in matrix multiplication, the matrix entries indicate the effect on meeting the requirements, in the left five by one column, that is caused by the design and operation of the systems, in the right five by one column. An off-diagonal matrix entry indicates a coupling, where the design and operation of one system impacts meeting another system's requirement, possibly affecting the second system's design and operation. Coupling leads to iterations to refine design and to complex cascade effects if operations are disturbed.

The top row shows how each system affects the atmosphere requirement, FR11. The atmosphere system DP11 maintains the atmosphere. The water system DP12 will probably have little direct effect on the atmosphere but in a recycling system DP12 may be required to provide water for electrolysis into oxygen to meet FR11. Such possible couplings that occur only in a closed recycling, as opposed to an open system, are shown in italics. The waste system DP13 may provide water evaporated into the cabin. The fire system DP14 may use fire suppressant gas and will probably require depressurization and repressurization of the cabin after a fire. This increases the stored atmosphere gas requirement, for an open or closed system. The food system DP15 does not impact the atmosphere directly. Crew metabolism is the source of the FR11 atmosphere requirements to provide oxygen and remove carbon dioxide, including for the the maximum crew metabolism, but the design of the food system should not impact meeting FR11. The atmosphere system DP11 can be largely designed to meet the crew support atmosphere requirement FR11, but its coupling to the other level 2 life support systems generates additional requirements that will require their own design parameters.

The DP12 water system may receive humidity condensate from the DP11 atmosphere system. The DP13 waste system will probably require water for urine flush or other waste processing, but this is an anticipated requirement for the DP12 water system. The DP13 waste system may provide urine and flush water recovered to the DP12 water system for recycling. The DP14 fire system will probably not use water and have no coupling to the DP12 water system. The DP15 food system can have very important couplings with the water system. If the food is dehydrated, it will need significant amounts of water. If the food is normally hydrated, some water may be required in preparation. Hydrated food can improve the water balance of a water recycling system, as the respired or excreted water may be recovered. A simple storage-based life support suitable for a brief mission can be largely uncoupled, but adding recycling to close the loop increases coupling.

The DP13 waste system may be constrained by the DP11 atmosphere system's ability to produce oxygen and remove carbon dioxide. It may also be constrained by the DP12 water system's ability to produce water or its need to receive waste water to recycle. The DP14 fire system may have no coupling with the waste system. The DP15 food system is probably the major source of waste, including food packaging and uneaten food.

The spacecraft design for fire reduction and fire suppression depends on the oxygen level required by FR11 but is not affected by the implementation of the DP11 atmosphere system. The FR14 fire suppression requirement seems to have little coupling with the water, waste, and food systems.

If the DP12 water system is designed before the food system, the food system will be constrained by the availability of water to rehydrate food or by the need to provide water in the food for ultimate recycling. The food and water systems are strongly coupled through the water balance, if water is recycled. It could be argued that the food system should be designed first to meet the FR15 requirements and the water system designed afterwards to accommodate the food system. If the food is fully hydrated and the water is supplied rather than recycled, the food and water systems are uncoupled. A resupply, non plant growing food system seems to have little coupling with the atmosphere, waste, and fire systems.

C. Reordering and decoupling level 2 life support requirements and systems

Since the requirements interact through the system designs, Figure 6 shows cyclic and reciprocal interactions. The atmosphere system DP11 may require clean water from the water system DP12 to produce oxygen and may provide humidity condensate to the water system DP12 for purification, forming a possible loop. Food hydration affects water balance and influences the water system design and the water system design may enable use of dehydrated food, another loop. The atmosphere – water, food - water, and water - waste couplings form reciprocal interactions with complementary entries accross the diagonal.

In general, an open loop design with oxygen, water, etc., all directly supplied and stored would have minimal coupling. But increased closure and recycling has cost advantages that can justify a more coupled design. The approach of axiomatic design is to retain design options while reducing decoupling and investigating lower levels.

The next step is to reduce coupling by reordering the requirements to simplify coupling, so that the more independent systems are designed first. The order above, atmosphere, water, etc., is the usual order of importance used in space life support descriptions. A reordered and decoupled matrix that has all entries on or below the diagonal allows a sequential design. That is, the requirements on the same level that are listed first can be designed for without the need to change the design after later systems designs are made. The five life support requirements and systems for a fully open loop system are reordered as shown in Figure 7.

FR15: Food		X						DP15: Food
FR14: Fire			X					DP14: Fire
FR13: Waste	=	Food and packaging waste		X			x	DP13: Waste
FR11: Atmosphere			De- and re-pressurize		X			DP11: Atmosphere
FR12: Water				Needs water		X		DP12: Water

Figure 7. Reordered open loop life support requirements, systems, and coupling matrix.

In an open loop system, the food and fire systems can be designed first and independently, the waste system is then designed to handle food and packaging waste, the atmosphere system designed to provide repressurization, and the water system to provide flush water to the waste system.

The closed loop system is much more coupled, and the existence of reciprocal interactions across the diagonal make it impossible to decoupled the matrix so it has all entries on or below the diagonal. The five life support requirements and systems for a closed system are reordered as shown in Figure 8.

FR14: Fire		X						DP14: Fire
FR15: Food			X	*Water balance*				DP15: Food
FR12: Water			*Needs water?* *Provides water*	X	Needs water *Provides water?*	*Provides condensate?*		DP12: Water
FR13: Waste	=		Food and packaging waste	Needs water *Provides water?*	X		x	DP13: Waste
FR11: Atmosphere		De- and re-pressurize		*Provide water?*	*Provides humidity?*	X		DP11: Atmosphere

Figure 8. Reordered closed loop life support requirements, systems, and coupling matrix.

The reordering does not provide complete decoupling but it does help understand the design process. The three open loop requirements-system pairs not in italics remain below the diagonal. The reordered matrix indicates that there are three design loops, similar to those identified in the DSN. There is a food-water loop since dehydrated food require water, hydrated food provides water in the atmosphere and waste for recycling, and this affects water system design. There is a water-waste loop where the waste system requires flush water and provides urine and flush for recycling. And there is a third wider atmosphere-water loop where humidity derived from crew metabolism and perhaps waste is removed from the atmosphere and provided to the water system for recycling.

The open system of Figure 7 meets the independence axiom but the closed system of Figure 8 does not. The closed system obviously is more complex and difficult to design. Its coupling can be reduced, for instance by using fully hydrated food. Such food contains nearly enough water for crew survival and the recycling water system would need much less closure because of the water from food entering the recycling system. If the food was fully hydrated, the next step in opening up the system would be to eliminate oxygen recycling. The amount of oxygen consumed is small relative to the water and oxygen can be produced from surplus water if available. Water recycling is the major mass saving in a recycling system and would be the last recycling to be eliminated to simplify the life support system.

D. The coupling of level 3 life support atmosphere requirements and systems

Figure 9 shows the level 3 atmosphere requirements and systems.

FR111: Nitrogen	=	X												x	DP111: Nitrogen
FR112: Pressure			X												DP112: Pressure
FR113: Temperature				X											DP113: Temperature
FR114: Ventilation					X										DP114: Ventilation
FR115: Remove contaminants						X									DP115: Contaminants
FR116: Remove humidity						Clean air	X								DP116: Humidity
FR117: Remove CO2						Clean air	Dry air	X							DP117: CO2 remove
FR118: Reduce CO2?								*Provides CO2?*	*X*						*DP118: CO2 reduce?*
FR119: Accept water, generate O2?									*Provides H2O?*	*X*			*Provides H2O?*		*DP119: O2 generate?*
FR1110: Oxygen										*O2*	X				DP1110: Oxygen
FR1111: Repressurize		N2									O2	X			DP1111: Repress
															DP12: Water

Figure 9. Level 3 atmosphere requirements and systems.

The nitrogen, pressure, temperature, and ventilation requirements can be provided using independent systems. The requirements and systems were ordered to create a decoupled, all below the diagonal matrix. A logical sequence for atmosphere processing is trace contaminant removal, humidity reduction, and carbon dioxide removal. (Wieland, 1994) The trace contaminant removal provides clean atmosphere to the humidity removing condenser and the condenser provides clean dry air to the carbon dioxide removal. Oxygen could be directly supplied from storage. If oxygen is recycled in a closed system design, the carbon dioxide would first be reduced to water and the water provided to an oxygen generator that produces water by electrolysis. Additional water could come from the water system DP12. The repressurization system DP1111 requires nitrogen and oxygen. The requirements, couplings, and systems for recycling are shown in italics. Increasing the system closure by recycling oxygen requires the atmosphere and water loops to be coupled and the systems integrated. (Wieland, 1994)

E. The coupling of level 3 life support water requirements and systems

The water system has only one basic requirement, to provide a sufficient quantity of potable water, but the water has several uses and, if it is recycled, derives from different sources. Figure 9 shows the level 3 water requirements and systems.

FR121: Crew wash and shower water		*X*										*DP121: Hygiene recycle*
FR122: Clothes wash water			*X*									*DP122: Laundry recycle*
FR123: Dish wash water				*X*								*DP123: Dish water recycle*
FR124: Water in food	=				X	X	*X*		*Condensate*		**x**	DP126: Stored water
FR125: Crew consumed water					X		*X*	*X*	*Condensate*	*Flush*		DP15: Food
FR126: Urine flush water					X			*X*		*Flush*		*DP124: Humidity recycle*
												DP125: Urine and flush recycle
												DP116: Humidity
												DP13: Waste

Figure 9. Level 3 water requirements and systems.

These requirements are based on the original space station requirements but are revised to attempt to identify the essential requirements and expand the implementation alternates. The space station crew uses wash water but does not wash clothes or dishes, as originally anticipated for space station. Separate recycling loops provide maximum independence in meeting the requirements, but a combined system is an obvious alternative. The shower, clothes wash and dish wash water would probably not be provided unless they were recycled. Storage is not a realistic option. Since some of the wash water for crew, clothes, and dishes will evaporate, it must be replaced by filtered condensate or alternately distilled urine and flush.

Stored water could be provided for all crew needs, food, drink, and flush. Some water will probably be provided in the food, even if it is not fully hydrated. If recycling is used instead of storage, a small amount of additional food and food preparation water could be provided by humidity recycling.

If there is no recycling, the crew consumed water and flush water must be provided from storage. Urine and flush recycling can provide enough water for urine flush. Both humidity recycling and urine and flush recycling are needed to provide enough drinking water for the crew.

The atmosphere humidity removal system DP116 would provide the condensate for recycling. The waste system DP 12 would provide the urine and flush for recycling. Since the matrix is not square, redundancy and coupling are usual.

The water needed for crew survival can be all, or nearly all provided in fully hydrated food. Any additional survival water should probably be provided by water stored in tanks. The other water consumed by the crew includes normal drinking and food preparation water and the water consumed in strenuous exercise and EVA, which would be curtailed in an emergency.

For brief missions, these other water needs will be small and all of the needed water can be provided from storage. For a long mission, the water needs will be large and some of the recycling processors shown in italics will be needed. The humidity condensate derived from crew perspiration and respiration is relatively easy to purify by filtering but it is not enough to meet the crew consumption demand, so urine derived from crew metabolism must be distilled. For maximum decoupling, the filter and distillation process are separate loops as originally proposed for space station, rather than combined to reduce cost as in the final space station. (Wieland, 1994)

V. Conclusion

Axiomatic design is significant because it is a logical systems design method that provides an alternate to the usual reliance on intuition, experience, and engineering judgment. It is intended to ensure good design by maintaining the independence of the systems designed to meet different functional requirements. The parallel step-by-step development of the lower level requirements and the identification of specific systems to satisfy them helps achieve independence. It avoids developing a complete detailed set of requirements based on an assumed but unconsidered design, or what would be worse, not having a design that meets the requirements. Axiomatic design helps avoid conflicting requirements, awkward design compromises, and unanticipated complexity.

Axiomatic design, like other matrix based methods, provides a clear method to link requirements to subsystems and to understand the relations and interfaces between the subsystems. Beyond the other matrix methods, it emphasizes that increased system robustness, stability, and operability results from uncoupling the functional requirements. This well known force for simplification is often neglected in the interest of improving efficiency and cutting cost. Eliminating duplication, cutting margins, and adding functions to existing systems all improve efficiency, but at the cost of increased complexity, multiplied interfaces, and susceptibility to cascading failures.

The axiomatic design approach provides a rationale to oppose increasing efficiency at the cost of simplicity and resilience. The high degree of closure and the complex integration of the space station life support system were intended to reduce cost but seem to have increased operational difficulties. Axiomatic design provides an important addition to the systems engineering process intended to meet the requirements with the best overall design.

References

Blanchard, B. S., *System Engineering Management*, Wiley, Hoboken, 1991.

Browning, T. R., “Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions,” IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT, VOL. 48, NO. 3, AUGUST 2001.

Do, S., and de Weck, O., “A Grammar for Encoding and Synthesizing Life Support System Architectures using the Object Process Methodology,” 2014-118, 44th International Conference on Environmental Systems Paper, 13-17 July 2014, Tucson, Arizona.

Engineering Design, Chapter 6: Axiomatic design, http://highered.mcgraw-hill.com/sites/dl/free/0073398144/934758/Ch06AxiomaticDesign.pdf, downloaded December 6, 2016.

Guenov, M. D., and Barker, S. G., “Application of axiomatic design and design structure matrix to the decomposition of engineering systems,” Systems Engineering, Volume 8, Issue 1, 2005, pp. 29–40.

Hooks, I. F., and Farry, K. A., *Customer-Centered Products: Creating Successful Products Through Smart Requirements Management*, AMACOM, New York, 2001, p. 145.

Kannengeisser, U., and Gero, J. A., A Comparison Between Pahl and Beitz’ Systematic Approach and the Design Behaviour of Mechanical Engineering Students, *mason.gmu.edu/~jgero/publications/.../15KannengiesserGero.EmpiricalSupport.pdf*, ca. 2014, downloaded Nov. 10, 2016.

Lee, J. M., “A Method for Capturing, Tracking and Ranking the Relative Competitiveness of Competing Alternatives,” 2014-178, 44th International Conference on Environmental Systems, 13-17 July 2014, Tucson, Arizona.

Liu, D., *Systems Engineering: Design Principles and Models*, CRC Press, Boca Raton, 2016, p. 19.

O'Shaughnessy K., and Sturges, R. H., “A systematic approach to conceptual engineering design,” Carnegie Mellon University Engineering Design Research Center, 1991, *A systematic approach to conceptual engineering design.pdf*, downloaded Nov. 10, 2016.

Pahl, G., Beitz, W., Feldhusen J., and Grote, K.H., *Engineering Design: A Systematic Approach*, Third Edition, Translators and Editors Wallace, K., and. Blessing, L. T. M, Springer-Verlag London, 2007.

Perry, J. L., Sargusingh, M. J., and Toomarian, N., “Functional Interface Considerations within an Exploration Life Support System Architecture,” 2016-90, 46th International Conference on Environmental Systems, 10-14 July 2016, Vienna, Austria.

Rechtin, E., and Maier, M. W., *The Art of Systems Architecting*, CRC Press, Boca Raton, 1997, p. 4.

Recthtin, E., *Systems Architecting: Creating and Building Complex Systems*, Prentice Hall, Englewood Cliffs, 1991, p. 15.

Suh, N. P. *Axiomatic Design: Advances and Applications*, Oxford University Press, New York, 2001.

Suh, N. P. *The Principles of Design*, Oxford University Press, New York, 1990.

Suh, N. P., *Complexity: Theory and Applications*, Oxford University Press, New York, 2005.

Wallace, K. M., and Blessing, L. T. M., “Observations on Some German Contributions to Engineering Design In Memory of Professor Wolfgang Beitz,” Research in Engineering Design, Springer-Verlag London, 2000,12:2–7.

Weinberg, G. M., *An Introduction to General Systems Thinking*, Wiley, New York, 1975, p, 10.

Wieland, P. O., Designing for Human Presence in Space: An Introduction to Environmental Control and Life Support Systems, NASA Reference Publication RP-1324, 1994, pp. 35,36.

Wikipedia, House of Quality, https://en.wikipedia.org/wiki/House_of_Quality, accessed Dec. 5, 2016.

Wikipedia, KISS principle, https://en.wikipedia.org/wiki/KISS_principle, accessed Dec. 5, 2016.

Yassine, A. A. "An Introduction to Modeling and Analyzing Complex Product Development Processes Using the Design Structure Matrix (DSM) Method," Urbana, 2004, DSM-Tutorial_English.pdf, downloaded Dec. 5, 2016.