UNISEC Space Takumi Journal for Practical Study of Problem Finding and Solving in Space Systems
UNISEC Space Takumi J. Vol.2, No.3 pp.27-38, 2012 Published online 1 June 2012 in Space Takumi Online Library(http://space-takumi.unisec.jp)

A Method of Quantitative Evaluation of Architecture and Systematic Resource Allocation in Definition Phase of Microsatellite Development

Ryosuke ISHII [1], Yukiya HANADA [1], Toru SUGIYAMA [1], Sota INOMATA [1], Hironori SAHARA [1,*,†]

[1] Department of Aeronautics and Astronautics, Tokyo Metropolitan University,
6-6 Asahigaoka, Hino, Tokyo 191-0065, Japan

SUMMARY

In this paper, we pay attention to a quantitative architecture evaluation of satellite in Definition Phase, and build a way for a systematic resource allocation method by applying Game Theory. We propose Design Complexity for the quantitative architecture evaluation as a clear index and apply it to CanSats for example, so that we become able to compare a satellite to others with respect to architecture or improve it to be simpler in terms of Design Complexity. Also, we propose a systematic resource allocation method by application of Game Theory, where a group of subsystems in a satellite are a set of players having their own strategies, and obtain Gain of the satellite by a combination of them. This significantly indicates that a satellite development should be progressed with a definite philosophy, where the proposed Design Complexity and Systematic Resource Allocation serve in the role of information for making the philosophy. Eventually, the way will newly construct Reasonable Reliability Engineering (so-called Hodoyoshi Reliability Engineering) and microsatellite can enter a new stage for industrialization and business in the future.

KEY WORDS: Hodoyoshi Reliability Engineering; Design Complexity; Resource Allocation; Game Theory; Microsatellite

[*] Corresponding author. Associate Professor, Member UNISEC.
[†] E-mail: sahara@sd.tmu.ac.jp.
Received and presented December 10th, 2011 in 1st UNISEC Space Takumi Conference, Kitakyushu, Japan.
Revised March 27th, 2012. Accepted March 30th, 2012.

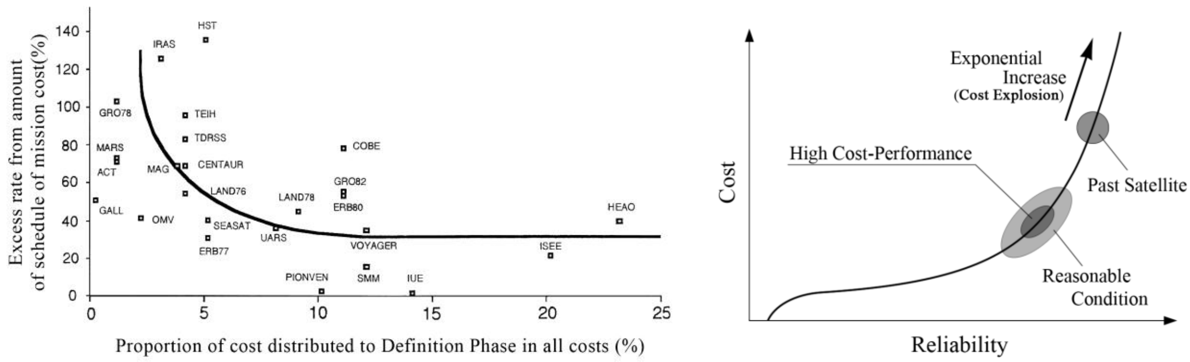
NOMENCLATURE

a	=	number of member variable
A_i	=	j-th class member variables
C_{des}	=	*Design Complexity*
C_{oup}	=	Coupling
C_{ohe}	=	Cohesion
C_i	=	cost for a strategy by i-th subsystem
C_N	=	total budget of a combination of all players
F	=	Number of Functions
G	=	number of independent graphs
L	=	number of links between components
m	=	number of methods
N	=	number of node (components) for calculation of Cyclomatic Complexity
N	=	player set in resource allocation
R_{sat}	=	reliability of whole system
$R_{H/W}$	=	reliability of whole hardware
$R_{S/W}$	=	reliability of whole software
R_{des}	=	reliability of design
R_{fab}	=	reliability of fabrication
R_{test}	=	reliability of test
R_{comp}	=	reliability of components
R_{sw}	=	reliability of each software
R_{op}	=	reliability of operation
R_{elm}	=	reliability of elements
S_i	=	strategy of i-th subsystem
S_N	=	strategy of a combination of all players
T_i	=	i-th solution
x_i	=	reliability of i-th subsystem
x_N	=	total reliability of a combination of all players
$\mu(A_i)$	=	number of methods accessing member variable A_i
ψ	=	factor of experience
π	=	Cyclomatic Complexity

1. INTRODUCTION

In recent years, a growing desire for industrialization of satellite development has highlighted a certain type of “tolerance” in project management, where the problems of frequent excesses in cost and irrelevant resource allocation must be solved for a sustainable and self-consistent development of the satellite industry. Much the same is true on microsatellite; an excessive resource allocation disturbs such advantages of microsatellite as low-cost and short-term development, even if we intend microsatellite for a contributor to the wider range of industrialization. Thus, an adequate quantitative evaluation of microsatellite architecture and an appropriate resource allocation are strongly expected there. The most importance for the future is a decision of resource allocation with quantitative architecture evaluation of microsatellite at the initial stage of project, known as “Definition Phase.” According to the practical accomplishment of past satellites shown in Fig. 1 left), you can see that there is a relation of one up to the excess rate of cost and the development cost spent on Definition Phase, and find that careful consideration in Definition Phase is very important in a project management. In fact, it can be said that the decision of appropriate resource allocation in Definition Phase is indispensable to prevent the cost excess and extension of development period in the project.

Additionally, although many of satellite development projects strongly depend on developer's experience and ability at the present, it disturbs the new entry to satellite industry as a result. To solve this, a more systematic project management is necessary. A result of resource allocation depends largely on indices used there. For instance, one of the indices to make a resource allocation in satellite development is reliability of hardware, which tends to be set high from the following factor particularly, that is, there is a difficulty of reliability evaluation especially of microsatellite due to a small number of productions and requirement of maintenance free after the launch. However, achievement of industrialization is never realized without the decision of reliability and cost in business. Then, the defect of resource allocation based on reliability should be solved, and it comes in two main types of inappropriate reliability criterion and powerful addiction on developer experience. A relation between reliability and cost of a satellite development denotes the same tendency of Fig. 1 right) in general, which suggests that cost performance lowers rapidly from a certain range. Therefore, in recent years, the discussion about reliability achieving a high cost performance for industrialization in microsatellite, is expected to produce *Reasonable Reliability Engineering*, or so-called *Hodoyoshi Reliability Engineering*.



Figures 1 Relations
left) between excess rate from budget and cost ratio distributed to Definition Phase [1]
right) between cost and reliability in satellite development

In this paper, we pay attention to a quantitative architecture evaluation of satellite in Definition Phase, and propose a systematic resource allocation method by applying the way of Game Theory [2]. As the result, we purpose an achievement of project management without depending on developer's experience and ability.

2. QUANTITATIVE ARCHITECTURE EVALUATION

An existing reasonable design method has been based just on component's reliability and chance failure, that is, supposed that we design a satellite perfectly and it breaks only because of chance failure in orbit. However, the actual failure of a satellite in orbit is caused by more "design failure" than chance failure [3]. In addition, accumulated know-how is very important in satellite development to avoid the design failure. No matter if the same satellite is developed, reliability is swayed by proficiency in design. Therefore, *Hodoyoshi Reliability Engineering* will include not only the conventional factors but also additional ones inspired in Definition Phase.

2.1 Reasonable Reliability Engineering Model

Though some models for *Hodoyoshi Reliability Engineering* were proposed in Japan, we here originally created and implemented our own model [4] as represented by Eqs. (1) – (3).

$$R_{sat} = R_{H/W} \times R_{S/W} \tag{1}$$

$$R_{H/W} = H(R_{des}) \times I(R_{fab}, R_{test}) \times J(R_{comp}) \times \psi \tag{2}$$

$$R_{S/W} = K(R_{sw}) \times L(R_{op}) \tag{3}$$

where a satellite-level reliability model is described. Additionally, we show the reliability model in increments of component in Eq. (4).

$$R_{comp} = H(R_{des}) \times I(R_{fab}, R_{test}) \times J(R_{elm}) \tag{4}$$

The above rate functions, *H*, *I*, *J*, *K* and *L*, are resolved by *Design Complexity* in our model. Alternatively, we deal mainly with hardware not including attendant software and operation in this paper.

2.2 Design Complexity

Though *Design Complexity* itself is not directly appeared in Eqs. (1) – (4), it affects microsatellite reliability indirectly. As a microsatellite system gets more complexity, its *Design Complexity* becomes higher and its reliability more decreases. *Design Complexity* consists of the four parameters of *Cyclomatic Complexity* [5], *Coupling* [6], *Cohesion* [6], and *Number of Functions*. These parameters are often used in software engineering and we alternated them for this proposition. We should use Function Flow Block Diagram (FFBD) [7] for visualizing the relationship among each components and functions. FFBD is used when they diagrammatically show system's signal flow and class structure for designing system architecture. While FFBD is definitely different from system diagram, we supposed that an element had just a function in order to provide module, function, and signal flow to subsystem, element, and wiring line in this paper. As the result, system diagram is regarded as FFBD approximately, and computing *Design Complexity* with an example system architecture shown in Fig. 2 is available.

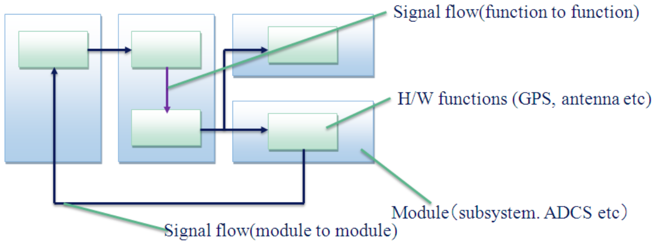


Figure 2 An example system architecture

2.2.1 Cyclomatic Complexity

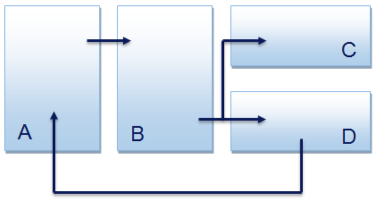
Cyclomatic Complexity is a barometer of architecture's redundancy and the number of feedback loop in a system, and is expressed in Eq. (5).

$$\pi = L - N + 2 \times G \tag{5}$$

If the equation's result is high, the signal flows of system have many forks, which is a cause of a lot of test cases and cost increase, and associated with a high probability of design miss. We compute *Cyclomatic Complexity* of the system shown in Fig. 2 to be $\pi = 4 - 4 + 2 \times 1 = 2$.

2.2.2 Coupling

Coupling is a barometer of strength between a component and the others. It is expressed by index of centrality. Index of centrality is a barometer of the number of preferred components and meliority of a network constructed by the components. We obtain 16.5 of *Coupling* in the example system as shown in Fig. 3.



	A	B	C	D	SUM	R
A	0	1	2	1	4	4
B	1	0	1	1	3	5.3
C	2	1	0	2	5	3.2
D	1	1	2	0	4	4
SUM					16	16.5

Figure 3 Coupling of the example system

2.2.3 Cohesion

Cohesion is a barometer of concentration in each component's function. If functions are not concentrated in a system, it is not a good system because circuit diagram's readability and test cost become exacerbated. We propose to use LCOM* [8] to express *Cohesion*. LCOM* is a method of expressing *Cohesion* in software engineering and defined as Eq. (6).

$$\mathrm{LCOM*} = \frac{\left|(1/a)\sum_{j}^{a}\mu(A_j)\right| - m}{1 - m} \tag{6}$$

Generically, LCOM* is close in value to 1 if there are small number of methods accessing each member variable. Adversely, LCOM* is close in 0 when a lot of methods access each member variable. In fact, low LCOM* system is good because method in modules cooperates with module's function. In this paper, we assume member variable as satellite subsystem's output, and method as its function. However, if there is a function group which has no output in a subsystem to another, we assume fringe of FFBD as a member variable. In addition, a subsystem having only one function lets the value least. The average LCOM* of each subsystem is defined as *Cohesion* of a whole satellite system. In keeping with these, we define satellite system's *Cohesion* as Eq. (7). Since LCOM* is the least at zero, we add 1 to the LCOM* average in Eq. (7).

$$C_{ohe} = \frac{1}{s}\left(LCOM * + 1\right) \tag{7}$$

2.2.4 Number of Functions

This parameter is very simple. What we have to do is only to count *Number of Functions*, *F*, in FFBD. In the case of the example system, the *Number of Functions* is 5.

2.2.5 Design Complexity

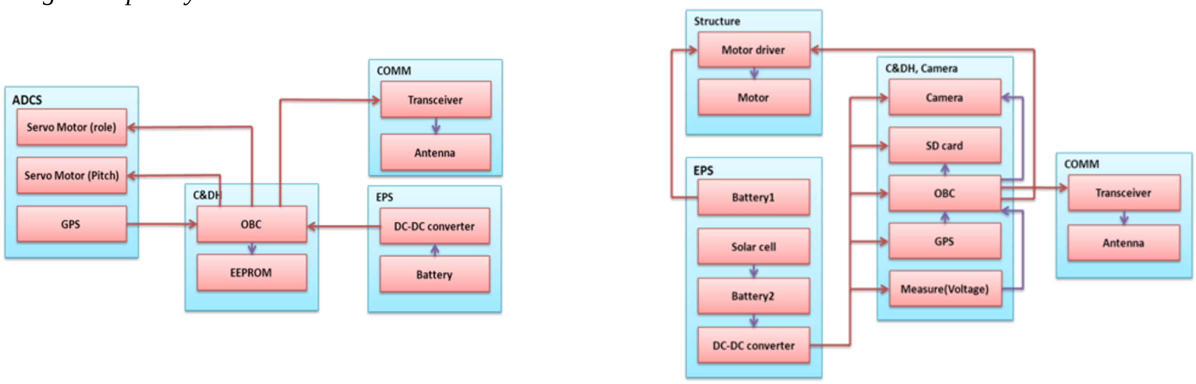
Summarized from the above, we propose to unity *Cyclomatic Complexity*, *Coupling*, *Cohesion* and *Number of Functions* to *Design Complexity* by Eq. (8).

$$C_{des} = \sqrt[4]{\pi} \times \sqrt[4]{C_{oup}} \times \sqrt[4]{C_{ohe}} \times \sqrt[4]{F} \tag{8}$$

C_{des} is affected by the four parameters equivalently. In addition, when all of the four parameters become *n*-times, C_{des} also becomes *n*-times.

2.3 CanSat System Evaluation

We apply *Design Complexity* for CanSat system. Indeed CanSat is a simple system compared to a satellite, but it has many basic functions required also for satellite such as communication, GPS, battery, and so on. Here, we evaluate two types of CanSat, the one is ours developed in Tokyo Metropolitan University (TMU) and another is a CanSat developed by Univ. -. Their system diagrams are drawn in Figs. 4, respectively, and the parameters and *Design Complexity* of them are listed in Table 1.



Figures 4 left) CanSat by Tokyo Metropolitan University, right) CanSat by Univ. -.

Table 1 Parameters and Design Complexity of the CanSats

	By TMU	By Univ. -	**By TMU (Improved)**
Number of subsystems	4	4	4
Number of functions	9	13	10
Number of lines between Subsystems	5	8	3
Number of lines within a subsystem	2	8	5
Cyclomatic Complexity	3	6	2
Coupling	16.8	16.5	16.8
Cohesion	1.3	1.3	1
Number of Functions	9 (do.)	13 (do.)	10
Design Complexity	**4.9**	**6.4**	**4.3**

The TMU CanSat has high *Cohesion* because functions in ADCS are installed in a subsystem in despite of their independence. If we add its own CPU having all interfaces from/to another subsystem as shown in Fig. 5, *Cohesion* becomes lower and its *Design Complexity* decreases from 4.9 to 4.3. This is a warrant of distributed system, which can be better in rather large system, that is, satellite. Indeed a small system like CanSat obtains little benefit by introduction of distributed system because small system's *F* has more strong influence rather than C_{ohe}, but functionally-distributed architecture can be helpful even in the larger system such as microsatellite.

Furthermore, we can also obtain the result that BUS type architecture is valid compared to STAR type one from *Design Complexity* as *Number of Functions* increases, especially in terms of *Coupling*, and a redundant system as much as possible is recommended in terms of *Cyclomatic Complexity*, as shown in Fig. 6.

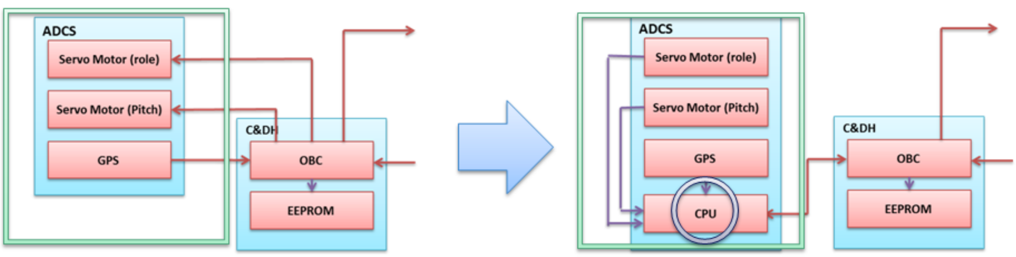


Figure 5 Improvement of the TMU CanSat

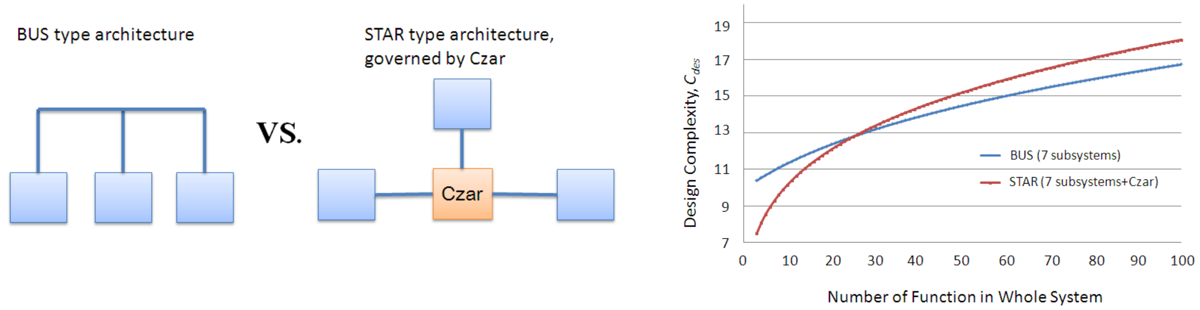


Figure 6 Comparison of BUS and STAR type architectures

As shown above, the combination of elements, components, and subsystems directly affects *Design Complexity*, C_{des}, and we can arrange C_{des} by changing the combination. Our instinct as engineers suggests that C_{des} shifts the parameters of R_{des}, R_{fab}, R_{test}, and so on, eventually a set of cost and reliability on the realization of the parameters. In other words, we can devise plural strategies indicated by a set of cost and reliability with the estimation using C_{des}. Unfortunately, no one has established influences of C_{des} to cost and reliability so far, that is, twice of C_{des} cannot mean twice of cost or reliability. However at least, C_{des} can determine the degree of difficulty for realization of a system. We are now collecting the data of various architectures and concerned parameters such as ex-ante and ex-post cost and reliability estimations in order to establish the relation between them soundly, so that we always welcome every piece of information in practical cases of satellite projects.

Here, we assume that we already obtain the relation and plural options in terms of cost and gain such as reliability, and that we can have a set of strategy, S_i, defining cost, C_i, and gain, x_i, of a subsystem i. It opens the door to come out where the strategy is and where we should point to in the midst of the space defined by cost and gain. In the Section 3, we introduce a way of adequate resource allocation by using the parameters related above.

2.4 Test Manhours Evaluation

Although we know that a number of tests such as unit test of subsystem, integration of subsystems, and integration test are very necessary in the process of satellite development, their evaluations are not really prevalent on ahead. As the result, we often encounter unanticipated matters in the process. It suggests that we should estimate the test manhours of development so that we determine whether or not our system is appropriate system configuration. Here, we propose a way of test manhours evaluation improving the method by Jong, et al. [9]. We count one test manhours for each one of unit test of a subsystem, integration test of subsystems, and overall test of system. Whereas the method by Jong, et al. counts the same test manhours for every subsystem, we count a product of the number of functions and interface ports in each subsystem because test manhours should depend on the scale of subsystem and the number of interface. We conducted the evaluation for the following two systems; the one in Fig. 7 is based on STAR type architecture and the other in Fig. 8 on BUS type architecture. Both of them are based on some actual satellite systems, and we actually built them into FPGA with Verilog-HDL based on their system diagram and conducted the evaluation of test manhours. We summarize their *Design Complexity* in Table 2, and test manhours in Table 3. We can determine that the BUS type architecture is adequate for this scale of system with respect to *Design Complexity* and *Test Manhours*.

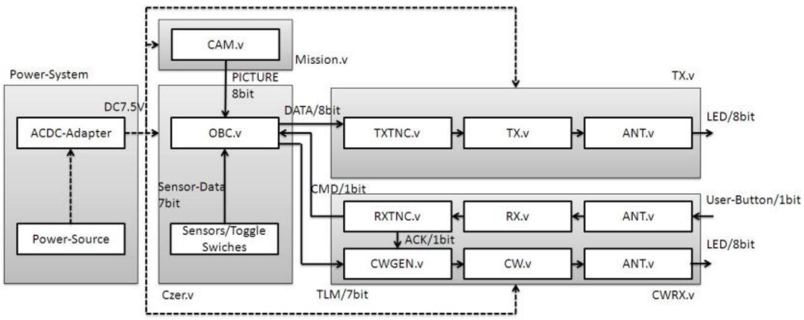


Figure 7 System diagram of STAR type architectures

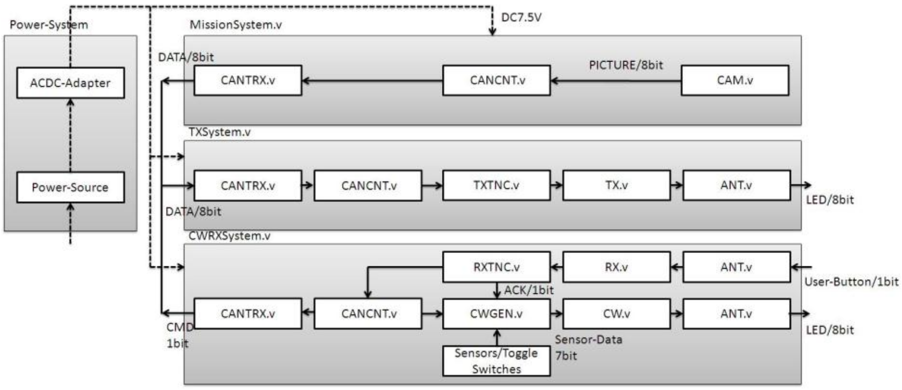


Figure 8 System diagram of BUS type architectures

Table 2 Design Complexity of STAR and BUS type Architectures

	STAR type architecture	BUS type architecture
Cyclomatic Complexity	1	1
Coupling	25.9	16
Cohesion	1.1	1.1
Number of Functions	14	19
Design Complexity	**4.5**	**4.3**

Table 3 Test Manhours of STAR and BUS type Architectures

	STAR type architecture	BUS type architecture
Unit test of a subsystem	20	19
Integration test of subsystems	18	16
Overall test of system	16	10
Sum of test manhours	**54**	**45**

3. RESOURCE ALLOCATION IN SATELLITE DEVELOPMENT

3.1 Application of Game Theory

There are various ones called “resource” in Definition Phase of satellite development, that is, cost, weight of a satellite, power supply, data bus resource, and so on. In this study, we target distribution of project budget to the respective costs for subsystems having respective strategies, for example. In actual satellite development, there is often a case where the final cost for a subsystem development is different from its initially-distributed one. This is derived from the poorly-thought-out resource allocation in the initial stage of Definition Phase, which indicates that it is necessary to achieve an appropriate resource allocation at the project plan stage for avoidance of the budget trade-off during the project's progressing. Here, we aim to solve it by a more systematic trade-off of a budget among subsystems in a satellite development project. Additionally, it will also improve the issues such as cost overruns and the efficiency of project management. In this paper, we construct and analyze a model of allocation of subsystem development cost by applying Game Theory in economics by treating this situation as a problem of game situation. We detail its concrete analysis in the followings.

Meanwhile, the resource allocation in satellite development is currently based on nothing but cost estimation of each subsystem by building up the concerning necessary works and resources. However, a clear index of the resource allocation concerning the substance reliability and cost, etc. does not exist so far. Hence, the selection of works and resources being needed in evaluation for requested reliability, are judged by developer's experience at the resource estimation stage in many cases. Generally, the judgment based on the individual experience often causes an inappropriate resource allocation, and remarkably influences cost overrun and excess in development period. In addition, such project management as strongly depending on the experience makes cost estimation and resource allocation difficult for an organization with little experience of satellite development, which disturbs their new entry to satellite development.

Therefore, a method for a Systematic Resource Allocation based on reliability criterion to subsystem is important, and its establishment can solve in-situ barrier in front of the industrialization of satellite development. Of course, the method is difficult to establish. Nevertheless, the most important thing is to find an element necessary to construct the Systematic Resource Allocation by analyzing reliability.

We modeled "distribution of subsystem development cost from the project budget to each subsystem" as a game situation. In this paper, this game situation is called "a project budget allocation game" and is analyzed based on the following settings.

- Player set *N* is defined to consist of seven subsystems. (Number of players , $n = 7$ here)
- Strategy S_i is simply defined as two kinds of budgets, S_{iH}, S_{iL}, distributed to subsystem *i*.
- The gain is defined as reliability x_i of subsystem *i*, and x_i is reflected in payoff function g_i and strategy set S_N of subsystem *i*.

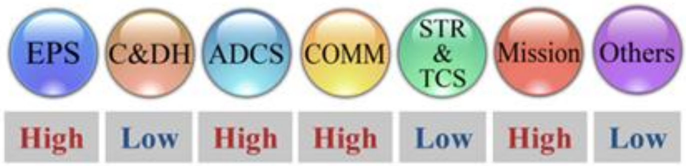


Figure 9 A combination example of strategy of each player

These seven subsystems are EPS, C&DH, ADCS, COMM, STR&TCS, Mission, and Others as shown in Fig. 9. Then, each player can simply have two strategies of "High" (S_{iH}) and "Low" (S_{iL}) with large and small amounts of distribution, respectively. We have an assumption that the amount of distribution to subsystems influences their system redundancy level and quality of components in the subsystem. In other words, "High" and "Low" strategies make the subsystems reliability gets higher and lower, respectively. As the result, total budget (C_N) and reliability (x_N) of the entire satellite development will be determined. In this case, the resource allocation game can be shown as Eq. (9).

$$\begin{aligned} N &= \{1, \quad 2, \quad 3, \quad \cdots, \quad 7\} \\ S_i &= \{s_{iH}, \quad s_{iL}\} \\ C_i &= f(S_N) \\ x_i &= g_i(C_i), \qquad i = 1 \sim 7 \\ C_N &= \sum_{i=1}^{7} C_i \\ x_N &= \prod_{i=1}^{7} x_i \end{aligned} \tag{9}$$

where *f* shows the rule of the budget allocation and g_i shows the reliability curve of each subsystem. A concrete budget allocation is analyzed by providing as the definitions of arbitrary *f* and g_i. Then, we numbered all the solutions obtained as a result of the game, $T_1 \sim T_{128}$. In this case, we analyzed the microsatellite consisting of the subsystems with the strategy and gain shown in Table 4. In an actual example, appropriate information on each subsystem like Table 4 should be expected at the first of a project. We analyzed a case based on the above-mentioned setting. This paper introduces the simplified example. Figure 10 shows the total budget (C_N) and reliability (x_N) of the entire satellite in all solutions ($T_1 \sim T_{128}$). You can find obvious Pareto Surface in Fig. 10, which coincides with widely held relation between cost and reliability as shown in Fig. 1 right) qualitatively.

Table 4 Reliability (Gain) and budget (Strategy) amount of a satellite. [Unit of budget: million yen]

	EPS		C&DH		ADCS		COMM		STR&TCS		Mission		Others	
	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy
High	0.999	3.92	0.999	5.88	0.999	6.49	0.999	3.06	0.999	5.27	0.999	10.78	0.999	7.47
Low	0.900	2.61	0.300	3.92	0.980	4.33	0.950	2.04	0.800	3.51	0.500	7.18	0.980	4.98

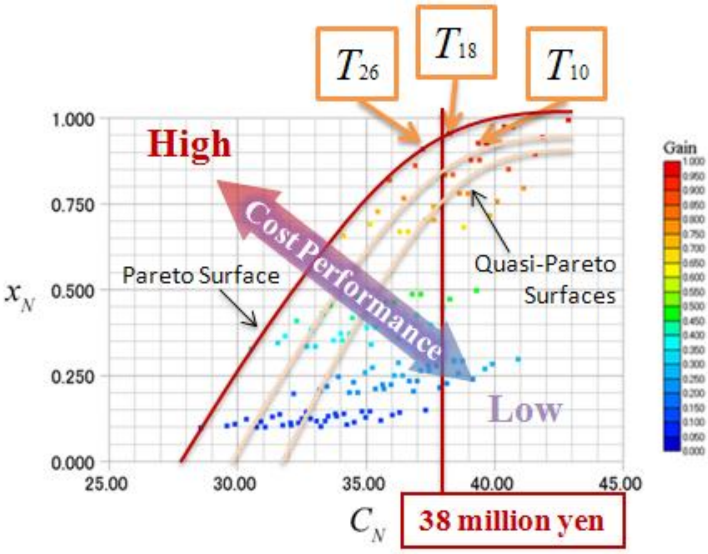


Figure 10 Reliability and total budget

T_{10}, T_{18}, and T_{26} are three abstracted solutions from all the solutions. In relationship between reliability and budget of satellite development, the Pareto Surface of the solutions represented by a red line in Fig. 10 trends the same tendency as the concept that represented in Fig. 1 right). However, some teams of satellite project may land on not Pareto Surface but one of Quasi-Pareto Surfaces shown in Fig. 10, so that they need the more cost to realize the scheduled gain compared to a skillful team stood on Pareto Surface. It suggests that factor of experience, ψ, of each teams may be evaluated by a distance or norm to Pareto Surface at the same level of x_N on the plain in Fig. 10. Moreover, if you choose a strategy among the solutions and are seeking an improvement, you should plot the choice on the plain of Fig. 10. Sacrificing something as the motion leftward or upward means that you already stand on Pareto Surface or Quasi-Pareto Surface and you can be tracking the surface, or, the motion contributes to an improvement in whole system if you are located at the point far from the surface. Eventually, you will arrive in the seemingly insurmountable barrier based on budget limit, 38 million yen for example. There is a point of Reasonable Condition represented in Fig. 1, that is, the solution based on *Hodoyoshi Reliability Engineering*.

Therefore, the application of Game Theory to a satellite development project is appropriate. The visualization of the relation between cost and reliability in a resource allocation of satellite development becomes an index achieving the resource allocation without the dependency on the experience. Moreover, it becomes possible to select the resource allocation to some extent with the policy of development, only by using Fig. 10. For example, if you want to develop more efficient, you have to select the case at the top-left section in Fig. 10 as well as T_{18} of this example. Then, if the budget limit has been established, you have only to select from the solutions below the limit. For instance, when you assume the upper boundary of budget to be 38 million yen, you should select T_{26} positioned on the upper-left corner of Pareto Surface. Three representative solutions, T_{10}, T_{18}, T_{26}, are listed in Table 5.

Table 5 Strategy of each player in each solution of T_{10}, T_{18} and T_{26}

T	EPS		C&DH		ADCS		COMM		STR&TCS		Mission		Others	
	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy
10	0.999	3.92	0.999	5.88	0.999	6.49	0.950	2.04	0.999	5.27	0.999	10.78	0.980	4.98
18	0.999	3.92	0.999	5.88	0.98	4.33	0.95	2.04	0.8	3.51	0.5	7.18	0.98	4.98
26	0.999	3.92	0.999	5.88	0.98	4.33	0.95	2.04	0.8	3.51	0.5	7.18	0.98	4.98

Furthermore, some satellite mission never allows decreasing the reliability of the specific subsystem for its feasibility. In this case, it is possible to adjust by lowering the reliability of "Low" of the subsystem. In fact, it is synonymous with lowering the reliability of the entire satellite greatly to drop reliability by the subsystem that requires high reliability. As the result, solutions where the subsystem takes "Low" are not selected easily. This method enables application to various situations by making other parameters change.

Indeed this is just a simplified model, but we are now trying to prove the probability of the application of Game Theory to a satellite development. Since we have no precise relation between cost and reliability, we may apply satisfaction degree of a mission, for example, instead of reliability at present.

3.2 Evaluation of Practical Project

We adopted the resource allocation with Game Theory to an actual satellite project; the satellite was launched several years ago and operated in orbit for years. First, we prepare the gain and strategy table as shown in Table 6 based on budget plan of the satellite.

Table 6 Gain and Strategy of the satellite. [Unit of budget: million yen]

	EPS		C&DH		ADCS		COMM		STR&TCS		Mission		Others	
	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy	Gain	Strategy
High	0.999	106.70	0.996	95.00	0.994	90.00	0.999	30.00	0.998	37.00	0.999	90.00	0.993	65.00
Low	0.400	64.06	0.650	23.29	0.830	44.09	0.980	20.80	0.750	16.64	0.970	62.40	0.880	45.76

We conducted an evaluation of the resource allocation by means of non-cooperative game and cooperative game with Commons Model (or the commons dilemma), and obtained their solutions, Nash equilibrium, and core. Figure 11 shows all of the solutions, that is, solution by non-cooperative game, Nash equilibrium of non-cooperative game, solution by Commons Model, Nash equilibrium of Commons Model, core of cooperative game with coalition in addition to the scheduled, and the actual points of the satellite. We can see the excessive cost between the ideal solution and the actual point, which means that a project management with high-performance was not attained on the satellite. In addition, we can understand that the conventional cost-plus and allocation was not realized because actual point is an extension of the line from scheduled point and Nash equilibrium by non-cooperative game. Therefore, the preliminary or timely evaluation for resource allocation should be conducted in a satellite development.

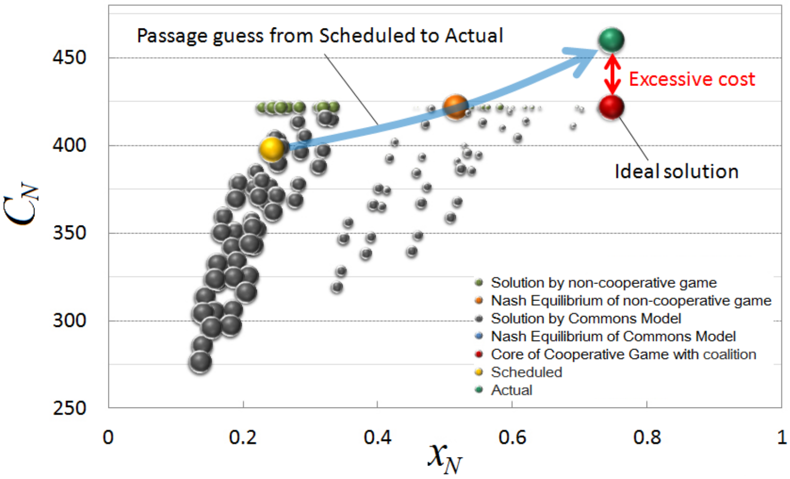


Figure 11 Evaluation result for development of the satellite

4. CONCLUSION

We proposed a quantitative architecture evaluation established with *Design Complexity* and applied it to CanSats for example, so that we became able to compare a specified satellite architecture to others or improve it to be simpler in terms of *Design Complexity* as a single and clear index. We also proposed a Systematic Resource

Allocation method by application of Game Theory, where a group of subsystems were players having their own strategies, and obtained gain such as reliability by a combination of them.

The above significantly indicates that a satellite development should be progressed with a definite philosophy, where the proposed *Design Complexity* and Systematic Resource Allocation serve in the role of information for making the philosophy. Eventually, the way will newly construct *Hodoyoshi Reliability Engineering*, and the microsatellite based on *Hodoyoshi Reliability Engineering* can enter a new era of industrialization and business for the future. Since we have no relation between *Design Complexity* and cost at present, we specifically propose the following procedure at present:

1. each subsystem draws at least two options of its architecture satisfying mission requirement,
2. each subsystem evaluates *Design Complexity* of the options and satisfaction degree or margin for the mission,
3. and the project manager conducts the Game, and should find the way to Pareto Surface or Quasi-Pareto Surface.

It is probable that cost decreases and reliability increases as *Design Complexity* reduces to some extent, and that the decrease of *Design Complexity* reduces reliability around Pareto Surface eventually, so that we regard *Design Complexity* as a concrete and effective index to obtain Pareto solution temporarily.

As a practical application, we are now progressing our ORBIS microsatellite project [10] timely by using the evaluation of *Design Complexity* and *Hodoyoshi Reliability Engineering*, and planned to launch it in 2015.

REFERENCES

[1] Yasaka, T., *Cost Issues in Space Development-Report from Space Cost Group*, Aeronautical and Space Sciences Japan, 45, 1997.

[2] Neumann, J. v, Morgenstern, O., *Theory of Games and Economic Behavior*, Princeton University Press, 1994.

[3] Saito, H., *Secondary Analysis on On-Orbit Failures of Satellite*, Journal of the Japan Society for Aeronautical and Space Sciences, pp. 190 – 196, 2011.

[4] Inomata, S., et al., *Concept of the Suitable Reliability Model for Developing Nano Satellite*, Proceedings of 54th Space Sciences and Technology Conference, pp. 2 – 3, 2010.

[5] McCABE, T. J., *A Complexity Measure, IEEE Transactions on Software Engineering*, vol. SE-2, No. 4, pp. 308, 1976.

[6] Ormandjieva, O., et al., *Measurement of Cohesion and Coupling in OO Analysis Model Based on Crosscutting Concerns*, ISWM 2005, pp. 3 – 4, 2005.

[7] NASA, *Systems Engineering Fundamentals*, pp. 49, 2001.

[8] Henderson-Sellers, B., *Object-Oriented Metrics: Measures of Complexity*, Prentice-Hall, pp. 142 – 147, 1996.

[9] I. S. M. de Jong, R. Boumen, J. M. van de Mortel-Fronczak, J. E. Rooda, *Integration and Test Strategies for Semiconductor Manufacturing Equipment*, IPA Lentedagen, 2006.

[10] Ishii, R., et al., *Development of Standard Bus for Science mission on Binary Black Hole Observation Satellite 'ORBIS'*, 3rd Nano-Satellite Symposium, 2011.

