ISSN 1345-3041

Dec.2022 / Vol.180

Mitsubishi Electric ADVANCE

Information Technology for Realizing Sustainable Societies (The first part)

MITSUBISHI ELECTRIC
Changes for the Better

Dec. 2022 / Vol. 180

Mitsubishi Electric ADVANCE

Information Technology for Realizing Sustainable Societies (The first part)

Mitsubishi Electric Advance is published on line quarterly (in March, June, September, and December) by Mitsubishi Electric Corporation.

Printed in Japan.

CONTENTS

Technical Reports

Precis

Mitsubishi Electric has established four business areas to achieve sustainable management: “Infrastructure,” “Industry and Mobility,” “Life,” and “Business Platform.”
This special feature covers “Information Technology for Realizing Sustainable Societies (The first part),” which is the foundation that supports all four areas.
The second part will be published in ADVANCE vol. 181.

Overview

Author: *Tetsuo Nakakawaji**

Innovation by IT Shall Realize Sustainable Societies

The progress of IT in recent years has been remarkable and has greatly changed society and life. As digital data and the Internet have spread widely, it has become possible to access various information from anywhere at any time. A closer observation of this trend reveals that it is evolving from IT to DX, or “Digital Transformation.”

The introduction of IT means streamlining the entire system and making it more efficient by converting information contained in tacit knowledge into digital information as explicit knowledge, storing it in a database, and exchanging it via networks. IT has enabled the manufacturing industry to visualize and streamline operations such as material procurement, processing and assembly production, and inventory and sales management, and to produce, ship, and sell large numbers of products with stable quality at low cost.

Although IT makes business more efficient, the effect is limited because the trend of improving business efficiency remains the same. It also tends to encourage mass production and mass consumption. In response, amid diversifying demand and environmental issues, the concepts of mass customization and subscription have emerged. The IT that supports this is what is called DX. The term “transformation” comes from the fact that full-scale use of IT, from consumer needs analysis to value chains, leads to revolutionary innovation that completely overturns existing values and frameworks.

In order to realize sustainable societies, IT is expected to be used not only for IT itself, but also for DX. This is because a sustainable society is hard to attain by conventional approaches such as improvements and reductions. A sustainable society can only be realized by making full use of all kinds of data, optimizing it in cyberspace, promoting the reuse of resources, and responding to diverse demands. For example, digital twin technology makes it possible to optimally build and control systems in physical space by aggregating information in cyberspace to reproduce and simulate events in physical space.

As an example, in the field of agriculture and food, the National Agriculture and Food Research Organization (NARO), to which the author belongs, is working to attain “Society 5.0 in the field of agriculture and food” by making full use of IT. This initiative is not limited to merely introducing IT that optimizes cultivation by measuring and analyzing water quantity and temperature information with sensors. In this initiative, NARO is challenging innovations such as designing crops that are resistant to climate change and disease, and designing food that is both healthy and delicious, by aggregating and integrating crop information such as genes and genomes, environmental information such as soil and weather, and information on diseases and pests to analyze them cross-sectionally using AI supercomputers before simulating and optimizing them in cyberspace.

**Vice President, The National Agriculture and Food Research Organization(NARO)*

Such efforts need to be made in all fields, including energy, transportation, disaster prevention and mitigation, medical care, education, and manufacturing. The key is big data. The word “big” here means both quantity and type; innovation is born by collecting a large amount of diverse data and integrating, analyzing and linking it cross-sectionally.

In order to realize a sustainable society through IT-based innovation, it is also important to integrate IT into organizations and businesses themselves. Since information is invisible, it is difficult for users to appreciate its cost. Companies and industries also need to reform their organizations and profit structures for information collection, integration, management, and utilization in order to achieve both economic development and to solve social issues. This is because without a structure that generates value as a business, the efforts of companies themselves will not be sustainable before attaining a sustainable society.

The more IT is used, the more value it creates. It is important to implement information processing functions without distinguishing between hardware and software, collaborate with various players through open innovation, and pursue value through agile innovation. It will also be necessary to deal with the negative aspects of the information society, such as security, privacy, and the digital divide.

Digital Transformation Piercing from Developments to Services

Author: *Noriyuki Minegishi**

1. Introduction

Digital transformation (DX) was proposed by Professor Erik Stolterman in 2004 in his comment, "Digital technology changes people's lives for the better in every aspect." Mitsubishi Electric's DX provides optimal solutions for customers and society and helps to create a sustainable society. DX has attracted much attention in recent years, and as the discussions have progressed, key terms such as "model-based development" and "digital twin" have become widely used. Both are introduced as a means of DX, but model-based development is sometimes introduced as DX itself. However, both model-based development and digital twin are concepts and are interpreted in various ways.

Mitsubishi Electric considers that model-based development and digital twin are closely related to the realization of business DX, and has defined each of them in its own way and clarified the technical concept for realizing DX from development to services by linking them.

This paper describes our model-based development and the definition of digital twin, and how they work together to realize DX.

2. Model-based Development

It is well known that model-based development in which design is performed on the cyber side reduces development man-hours [(1)]. Conventionally, products (mainly equipments) were developed based on experience, prototyping, and evaluation, but by using models to digitally adjust development, rework can be reduced. Most development work uses 3D models, or a logical model (1D model) is used for analyzing one physical phenomenon.

However, Mitsubishi Electric positions everything from requirement & function analysis to detailed design as model-based development, and defines model-based development as realizing holistic optimization that includes not only equipments but also systems.

2.1 Scope of model-based development

Mitsubishi Electric has set the scope of model-based development as model-based system engineering (MBSE), which performs requirement & function analysis and functional design, and model-based design (MBD), which performs logical design and detailed design (Fig. 1).

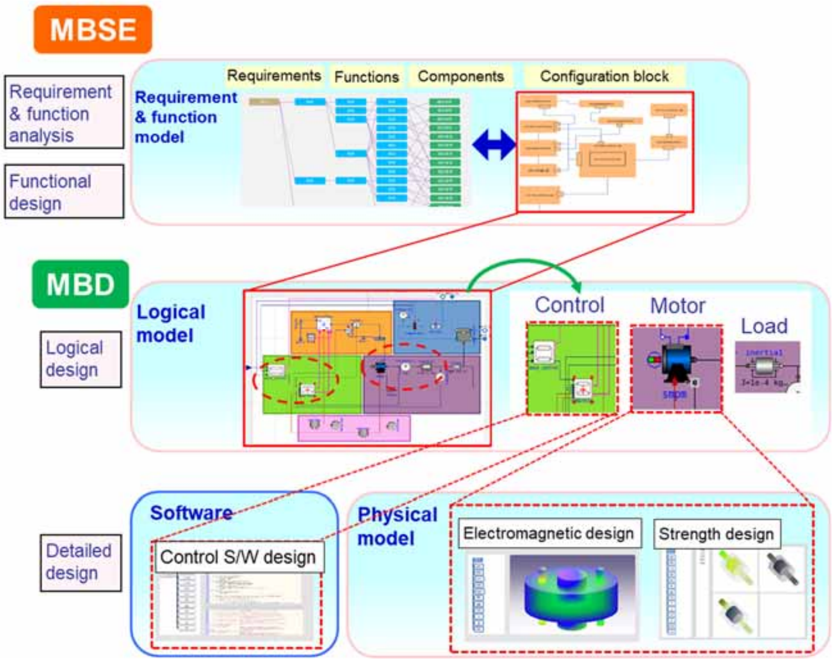


Fig. 1 Scope of our model-based development

*Information Technology R&D Center

The origin of product development is the requirements for the product. These requirements are analyzed in consideration of the life cycle and stakeholders of the product to be developed. The requirements are defined not only for product operation and performance, but also for cost and quality.

Next, all the functions that realize the requirements are extracted, and the components that realize the functions are assigned. Functions and components are represented by several means such as tree diagrams, block diagrams, and orthogonal arrays to prevent omission.

These are the requirement & function analysis and functional design processes, which are implemented using the MBSE method. The "requirement & function model" created in this process is shared between the discipline engineers that combine individual elements after the logical design and the systems engineers, and is used to solve problems that arise in individual element design.

Once the components are determined, the process proceeds to logical design. Logical design uses a "logical model" including a 1D model that abstracts physical phenomena. The 1D model is expressed by mathematical formulas representing related physical phenomena. Detailed design embodies the logical model expressed in mathematical formulas into something tangible to satisfy physical requirements. Logical design and detailed design are defined as MBD.

The design results of each process are verified, such as requirement & function analysis, functional design, logical design, and detailed design, and consistency across different functions and different physical elements in the processes is also verified, thereby eliminating the major rework of reconsidering requirements and functions from the test results after prototyping. Starting with the requirements based on the product life cycle, the level of abstraction is lowered step by step for verification, and problems are also solved in the downstream process with an overview of the scope of impact, thereby realizing more efficient development beyond the scope of adjustment based on pre-prototype models. The scope of our model-based development is the combined scope of model-based system engineering (MBSE) and model-based design (MBD).

2.2 Holistic optimization by model-based development

Our model-based development, including MBSE and MBD, not only improves development efficiency but also realizes holistic optimization (Fig. 2).

In conventional MBD, the discipline engineers read and define the required specifications of their own parts from the higher-level specifications and proceed with the design so that only the requirements of their own field are satisfied. This causes them to lock themselves in their own work and lose sight of their surroundings. As a result, in some cases the requirements of target products may not be achieved in tests and evaluations that combine multiple element designs, or consistency with other elements may not be achieved.

On the other hand, in Mitsubishi Electric's model-based development, the model created by MBSE is used to comprehensively understand the relationship between the individual element design in charge and other elements. After that, logical design is performed using a 1D model, and verification is performed at the higher-level in combination with other elements, thereby making it possible to confirm that the requirements of the higher-level are satisfied, beyond the responsibility of the person in charge. Furthermore, when it is difficult for the person in charge to satisfy the requirements within their responsibility in the logical design or detailed design, the requirements of the object can be satisfied with the optimal solution by returning to the requirement & function model, and discussing and coordinating with the higher-level person in charge and other element designers. Holistic optimization can be realized by allowing the element designer to maintain an overview of the whole.

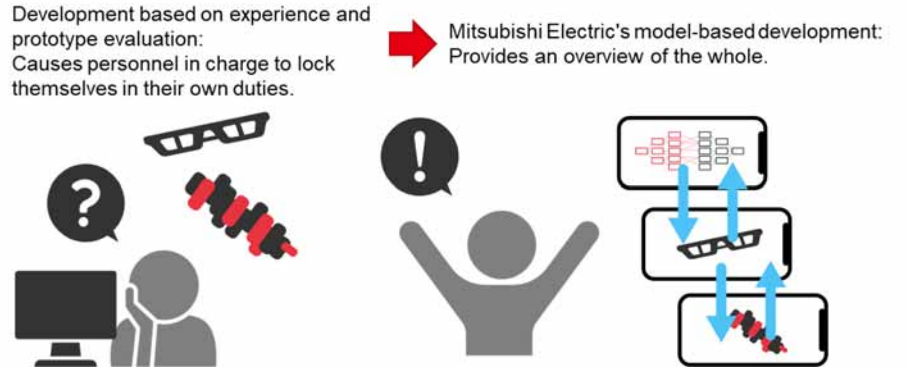

Development based on experience and prototype evaluation:
Causes personnel in charge to lock themselves in their own duties.
Mitsubishi Electric's model-based development:
Provides an overview of the whole.
?
!

Fig. 2 Holistic optimization by model-based development

2.3 Model-based development from systems to elemental technologies

The processes described so far, from requirement & function analysis to detailed design, are not limited to the processes of equipments. The development is carried out step by step from systems (System of Systems (SoS) at a higher-level) to elemental technologies such as parts and software (Fig. 3).

Figures 1 and 2 show the case of equipments as examples. However, the source of the value delivered to customers is the "systems" in which equipments are linked together, or the SoS in which systems are linked together. Figure 3 shows how model-based development is applied to stages from systems to parts and software. First, in the system stage, processes from requirement & function analysis to logical design are performed, and the equipments that make up systems are defined. In the next stage, processes from the requirement & function analysis of equipments to logical design are performed, leading to the detailed design of parts and software. Parts and software design results are verified by equipments, and equipment design results are verified by systems. In this way, starting from the customer's requirements, the systems, equipments, and ultimately the parts and software are studied, and the customer's environment is envisioned and verified, thereby enabling the reliable provision of continuous customer's value.

3. Digital Twin

A digital twin is defined as a digital representation of an equipment or system; the digital twin itself does not create value.

Mitsubishi Electric defines a digital twin as "a means of reproducing, at a remote location, the invisible situation of a phenomenon at the operating site." This is a method, so it is necessary to clarify the purpose, that is, the "value," and to build a digital twin that can realize the value. Values can include:

(1) Realization of services such as operation and maintenance optimization
(2) Realization of manufacturing and construction as planned
(3) Confirmation of phenomena occurring in tests at the development stage

In order to precisely reproduce phenomena that cannot be seen at the operating site, Mitsubishi Electric's models are utilized of its products obtained through the model-based development described in Chapter 2, and also utilizes necessary models other than its products based on its abundant knowledge in the industry to create a digital twin.

Maintenance optimization is a well-known example of a digital twin [(2)]. Also, the manufacturing industry [(3)] and construction industry [(4)] use digital twins for realizing manufacturing and construction as planned, and for confirming prototypes through testing [(5)].

As a use case, this chapter describes a digital twin for the purpose of optimizing operations in a temperature-controlled warehouse warehouse.

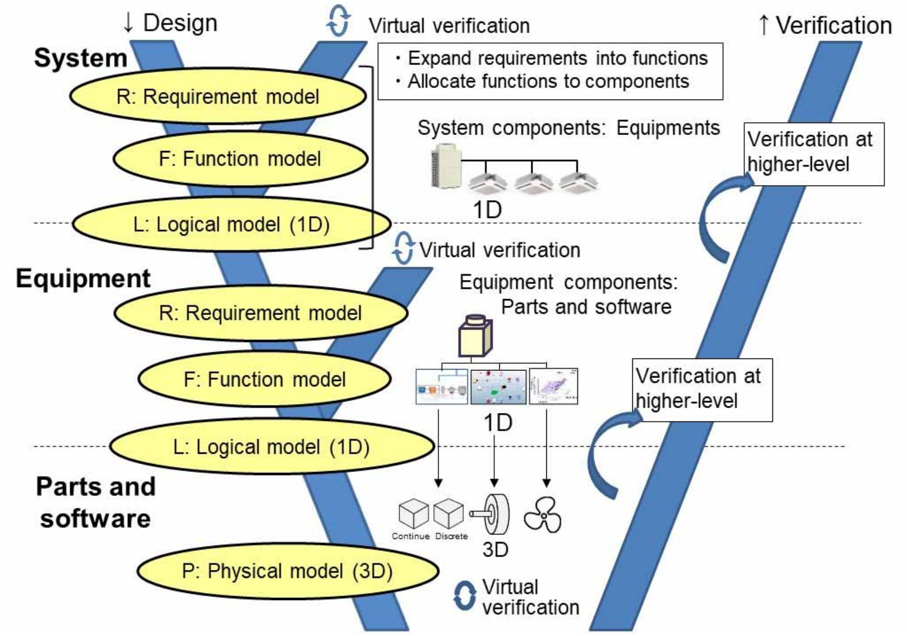


Fig. 3 Model-based development from systems to parts and software

3.1 Optimization of the operation of a temperature-controlled warehouse

When building a digital twin for optimizing the operations of a temperature-controlled warehouse, the value, means, and services to be provided must be considered (Fig. 4).

(1) Customer value (purpose)

The assumed customer is the owner of the warehouse. In order for the owner to make a profit, it is necessary to (i) make the most of the warehouse space, (ii) ensure product quality (ability to control the temperature suitable for products), and (iii) reduce operating energy costs.

(2) Means of achieving customer value

Possible means to achieve the value (purpose) of (1) are: (i) optimization of product placement (positioning that maximizes space utilization while improving temperature control efficiency), (ii) optimal control of refrigerators (control that does not reduce the quality of products and does not cool them too much), and (iii) provision of refrigerators that consume less power.

(3) Services that need to be provided in addition to equipments

In order to deliver value to customers, it is necessary to provide business optimization consulting services or business optimization programs (control simulators), in addition to equipments.

3.2 Digital twin for optimizing temperature-controlled warehouse operations

A digital twin has been built to optimize the temperature-controlled warehouse operations described in Section 3.1 (Fig. 5). By linking IoT data with models created by model-based development, the digital twin is realized as a means of creating value.

The warehouse model, which is the core of customer value, consists of a wall model, an airflow model, and a product model. When considering value, the temperature of the products in the warehouse is of concern. The temperature of the products is “the invisible situation of a phenomenon at the operating site,” and as a component necessary for reproduction, a model was derived through the process of model-based development. Regarding the temperature inside the warehouse, the wall model is used to derive the inner wall temperature from the outside air temperature and the amount of solar radiation on the wall, and the airflow model is used to estimate the temperature changes of the products.

Equipments and their control are the main elements for “achieving both product quality and minimizing energy costs” provided by Mitsubishi Electric. The models created in the design process are used for the indoor unit and refrigerator models (control model and physical model) of the refrigeration equipment.

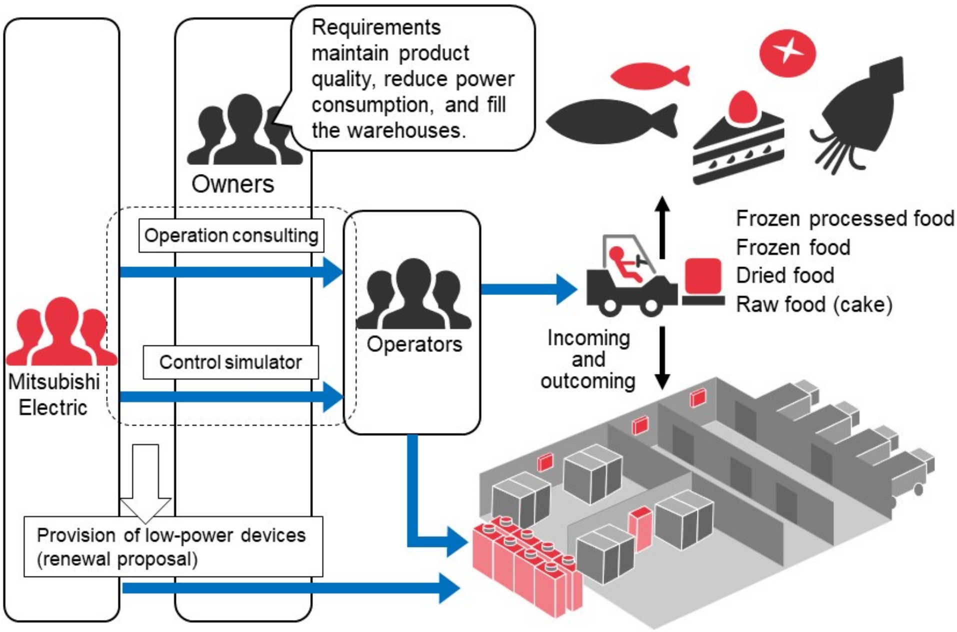


Fig. 4 Optimization of operation of temperature-controlled warehouse

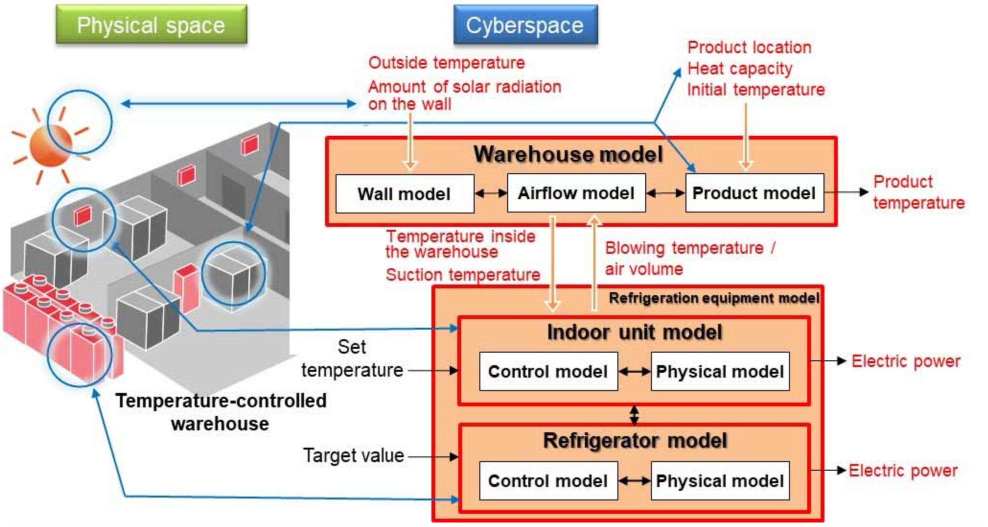


Fig. 5 Digital twin system for optimizing operation of temperature-controlled warehouse

Models configured in this way are linked with IoT data to reproduce the warehouse situation with high accuracy. The weather and temperature at a certain point in time, as well as the state of products (heat capacity of products, location, temperature at the time of delivery) are input into the warehouse model, and the sensor data and setting values of equipments are input into the equipment models. From this initial state, control is performed to minimize power consumption without impairing the quality of products, and the resulting power is obtained as an output. The warehouse state at the time the sensor data is input into the model is reproduced, and the future state is predicted based on optimal control thereafter.

Maisart's[6] latest AI technology is used for estimating the detailed internal state, predicting the future state, and optimization based on the future state. In addition, ClariSense performs the collection of IoT data, data processing such as removing noise contained in data and data correction/coordination, connection with models, and library creation when reusing models.

4. Conclusion

This paper described Mitsubishi Electric's model-based development, which not only improves development efficiency through digital adjustment, but also optimizes systems and equipments as a whole. It also described our digital twin, which reproduces the invisible situation at the operating site through estimation and prediction based on models created during the development process and IoT data, and showed the digital twin of a temperature-controlled warehouse as a specific example. Both the model-based development and digital twin are DX from development to service based on customer value. With our DX, we will continuously deliver value to customers through our products, both equipments and systems.

References

(1) Model Based Development Issues for Popularization in Japan, FOURIN, Inc. (2020)
(2) GE Reports: "Data" will support the future of aviation https://www.gereports.jp/airline-of-the-future/
(3) SIEMENS: Digital Twin https://new.siemens.com/jp/ja/company/stories/research-technologies/digitaltwin/digital-twin.html
(4) Gateway: Digital transformation in the construction production process https://gateway.smartconstruction.com/
(5) Ansys: Creating a DIGITAL TWIN for a Pump https://www.ansys.com/content/dam/product/systems-embedded-and-integrated/twin-builder/creating-a-digital-twin-for-a-pump-aa-v11-i1.pdf
(6) H. Mishima: Current Status and Future Prospects of AI Technologies in Mitsubishi Electric Corporation, Mitsubishi Denki Giho, 94, No.6, 318–323 (2020)

Model Based Design Adjustment Technique for System Design

Authors: *Yasuhiro Omori*, Masakatsu Toyama**, and *Masazumi Okada**

1. Introduction

Systems engineering is a systematic method for developing large-scale, complex systems. The method involves clarifying requirements based on the purpose and background of the system and proceeds with design while integrating multiple specialized fields [(1)]. Systems engineering includes model-based systems engineering (MBSE), which is performed using models [(2)].

The development of systems and devices in MBSE also requires adjustment in detailed design to meet the system requirements. In many cases, the adjustment is performed using the results of detailed design utilizing calculation formulas that convert 1D/3D-CAE models and know-how into explicit knowledge in Excel *[1].

With the aim of facilitating adjustment in system design, this paper defines a modeling method that specifies the dependencies between designs in different fields such as mechanical, structural, electrical, electronic, and software when constructing a system model using MBSE, and a method for linking the system model with the detailed design results of the 1D/3D-CAE model used in each design. It also describes the results of confirming the feasibility of analyzing the impact of each design result on the entire system using the defined method.

2. Method Design

2.1 Method overview

Systems engineering is a method of proceeding with the detailed design of the target system by examining the entire lifecycle of the system to be designed (development, manufacturing, transportation, installation, operation, maintenance, disposal, etc.), and by clarifying roles from the perspective of interactions with surrounding systems, stakeholders, environmental conditions, etc. [(2)] This makes it possible to reduce omissions in examining the functions to be provided by the system, thereby reducing rework from the post-process.

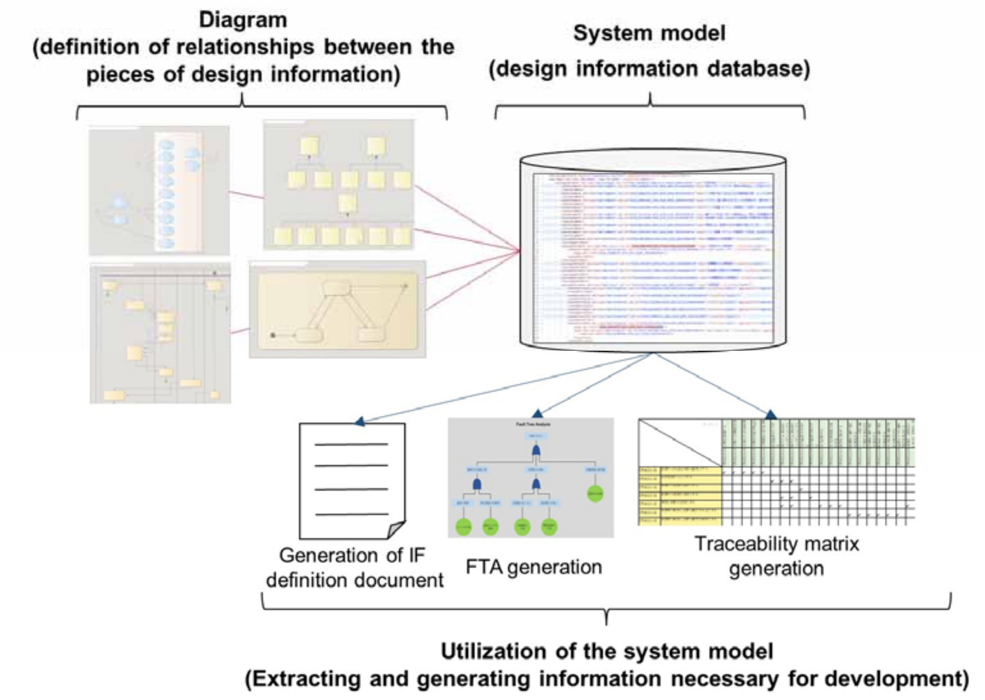


Diagram
(definition of relationships between the pieces of design information)
System model
(design information database)
Generation of IF definition document
FTA generation
Traceability matrix generation
Utilization of the system model
(Extracting and generating information necessary for development)

Fig. 1 System model and its utilization in MBSE

[1] Excel is a registered trademark of Microsoft Corp.

*Information Technology R&D Center.

In systems engineering, an enormous amount of information must be handled, so the design work should not be managed manually using documents, but should be implemented in a format that can be managed as digital data. This is used to create a database of design information so that the necessary information can be searched, and consistency can be checked using a computer. This database of system design information is the system model in MBSE which extracts the information necessary for proceeding with the design from the database and generates the format required for making design decisions, thereby promoting efficient, effective development [(2)].

The system model can be created by using various diagrams (Block Definition Diagram, Requirement Diagram, Parametric Diagram, etc.) based on notation such as SysML[(3)] *[2] in the MBSE tools to create relationships between the pieces of design information (Fig. 1).

In addition to the system model, models used in development include detailed design models by 1D/3D-CAE used in detailed design. The models are simulation models for optimization in each design field (mechanical, structural, electrical, electronic, software, etc.), and by adjusting the design results using the detailed design models, specifications that satisfy system requirements are derived. When making adjustments between detailed designs, design elements with dependencies that affect other designs are adjusted, but it takes time and causes omissions in examination because the adjustments are based on the tacit knowledge of each designer.

For this reason, a mechanism is created to extract the dependencies between detailed designs from the system model, which holds information about the entire system design, and to link the results of the detailed design models. As shown in Fig. 2, this makes it possible to visualize the impact of detailed design results on the entire system, thereby facilitating adjustment.

In order to realize this mechanism, definitions are made for a system model construction method that can specify dependencies between designs, a method of extracting the dependencies from the system model, and a linkage method that calculates the impact by passing the dependency information to detailed design models.

2.2 Definition of the system model

Some of the design elements (control cycle, component rating, power consumption, etc.) in each design field affect other design fields. For example, when designing an air conditioner, if the control cycle is changed to lower the fan speed to reduce power consumption, an impact event may propagate in which a change in heat dissipation causes a change in the performance value of the cooling and heating capacity in the design of the refrigerant circuit.

In order to be able to extract the propagation of impact across designs from the system model, the diagrams described in Section 2.1 are used to define relationships between the pieces of design information.

This method uses SysML Block Definition Diagrams and Parametric Diagrams to define the dependencies between the design elements of various designs and the target performance. In addition, performance targets, which are system requirements, are defined in SysML requirement diagrams. Figure 3 shows an example of relationships between the Requirement Diagram, Block Definition Diagram, and Parametric Diagram for an air conditioner. In this paper, Sparx Systems' Enterprise Architect is used as MBSE tools.

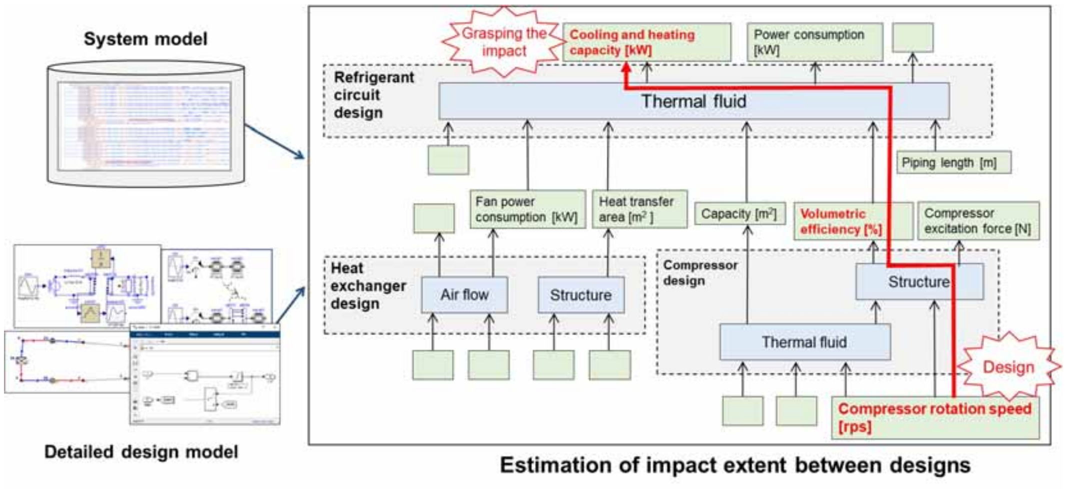


Fig. 2 Overview of adjustment facilitation method using system model

[2] SysML is a registered trademark of Object Management Group, Inc.

2.3 Analysis of the system model

The system model shown in Section 2.2 is expressed as a SysML diagram, but as described in Section 2.1, it is actually a database of design information. The diagram shown in Fig. 3 is created using notation such as SysML, and the necessary pieces of information are distributed over multiple diagrams. When trying to use diagrams for development, all developers must be familiar with the notation and be able to understand diagrams without any discrepancies to find the necessary pieces of information from multiple diagrams. Therefore, it will be difficult for the people involved in development to use the diagrams as they are.

Many MBSE tools hold the details of each element defined on the diagrams and information about relationships with related elements in the format of eXtensible Markup Language (XML). By constructing a mechanism for extracting the necessary pieces of information from the contents of the diagrams expressed in XML, it becomes possible to make extensive use of the system model in the development process.

Here, to be able to extract the necessary pieces of information from XML, it is necessary to define relationships between the pieces of information defined in multiple diagrams. Considering the example of an air conditioner, if relationships between the pieces of information are not defined, it is not possible to trace on a computer that the target cooling capacity defined in the Requirement Diagram will be the performance target value of the refrigerant circuit design defined in the Parametric Diagram. As a requirement for the system model that enables tracing on a computer, it is essential to relate the pieces of information in the diagrams to each other.

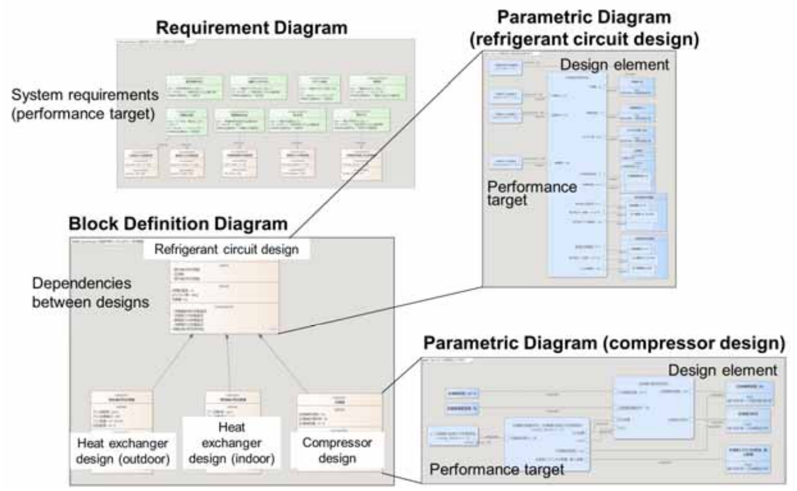

Requirement Diagram
System requirements (performance target)
Parametric Diagram (refrigerant circuit design)
Design element
Performance target
Block Definition Diagram
Refrigerant circuit design
Dependencies between designs
Parametric Diagram (compressor design)
Design element
Heat exchanger design (outdoor)
Heat exchanger design (indoor)
Compressor design
Performance target

Fig. 3 Contents of system model

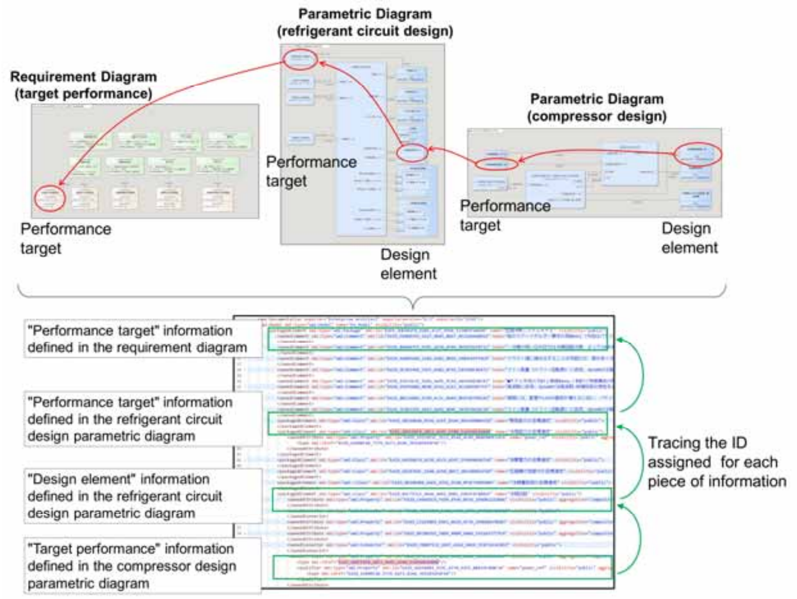

Parametric Diagram (refrigerant circuit design)
Requirement Diagram (target performance)
Parametric Diagram (compressor design)
Performance target
Performance target
Performance target
Design element
Design element
"Performance target" information defined in the requirement diagram
"Performance target" information defined in the refrigerant circuit design parametric diagram
"Design element" information defined in the refrigerant circuit design parametric diagram
"Target performance" information defined in the compressor design parametric diagram
Tracing the ID assigned for each piece of information

Fig. 4 Analysis of system model

Figure 4 shows that relationships between the pieces of information in the diagrams shown in Fig. 3 are defined, and the relationships can be traced on XML. The performance targets defined in the Requirement Diagram are used as the performance targets in the Parametric Diagrams, and the Parametric Diagrams relate each design element to each performance target. Also, a performance target shown in one Parametric Diagram is related to a design element in another Parametric Diagram. Looking at the relationships in terms of the XML structure, a unique ID is assigned to each piece of information, and the ID of each related piece of information is defined. This makes it possible to trace each related piece of information by referring to the ID, and to extract the relationship between the design element of one design and the performance target of another design. In this way, the pieces of information in the diagrams are related with each other, thereby enabling computer processing to extract desired information from XML.

2.4 Linkage method with detailed design model

The method described in Section 2.3 enables the extent of design impact to be traced from the system model. On the other hand, to estimate the impact quantitatively, it is necessary to link with the detailed design simulation models.

There are two methods for linking with these models: a method that links MBSE tools and simulation tools(4), and a method that converts the execution results of models for detailed design into approximate formulas to give the design values to the approximate formulas, and returns the results to the system model.

This paper defines the latter linkage method. This method takes a lot of time and effort to create approximate formulas, and it is less accurate than the method of linking with simulation tools. However, Mitsubishi Electric’s products are diverse, and the simulation tools used for each business domain are different, and our own in-house tools and Excel calculation formulas are often used. Therefore, we have decided to use this method, which is easy to link with various tools. Figure 5 shows the linkage method between the system model and detailed design models according to this method.

As shown in Fig. 4, the design element and the performance value related to the design element are extracted from XML of the system model. The quantitative impact of the performance value when the value of the design element is determined is then calculated from the approximate formula of the relationship between the design element and the performance value. If the performance value for which the impact is calculated is a design element of another design, the performance value with further impact is extracted from XML and the magnitude of the impact is calculated from the approximate formula. By repeating this process it is possible to express the extent and magnitude of the impact that propagates when the value of a certain design element is determined.

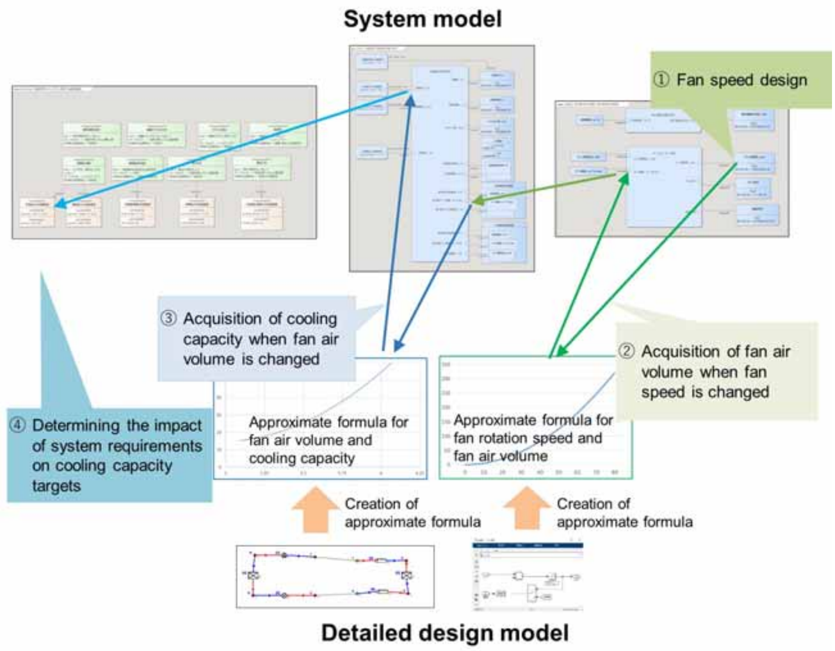

System model
① Fan speed design
③ Acquisition of cooling capacity when fan air volume is changed
② Acquisition of fan air volume when fan speed is changed
④ Determining the impact of system requirements on cooling capacity targets
Approximate formula for fan air volume and cooling capacity
Approximate formula for fan rotation speed and fan air volume
Creation of approximate formula
Creation of approximate formula
Detailed design model

Fig. 5 Linkage method between system model and detailed design model

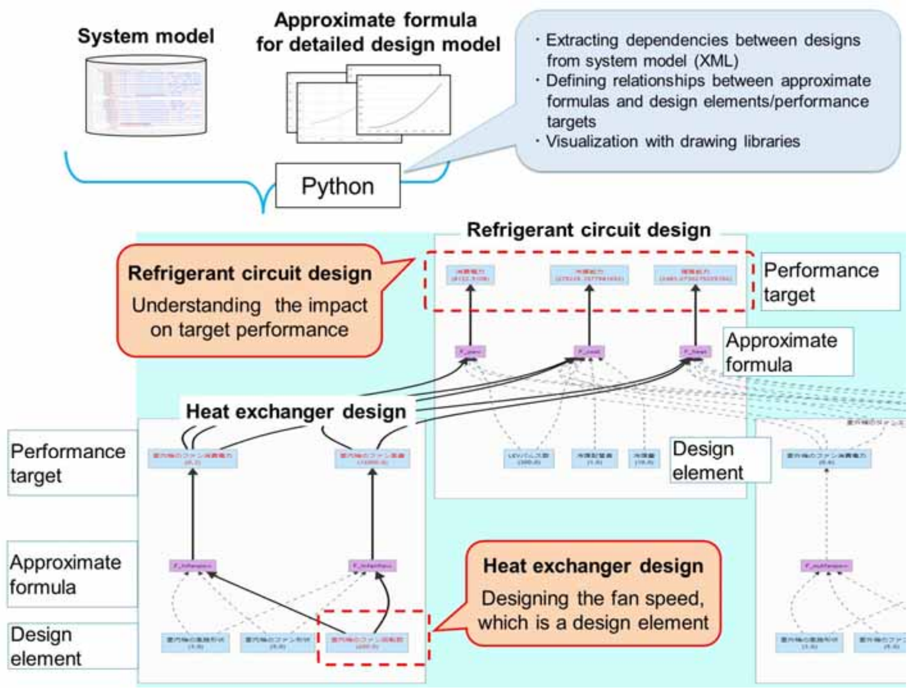

System model
Approximate formula for detailed design model
· Extracting dependencies between designs from system model (XML)
· Defining relationships between approximate formulas and design elements/performance targets
· Visualization with drawing libraries
Python
Refrigerant circuit design
Refrigerant circuit design
Understanding the impact on target performance
Performance target
Approximate formula
Heat exchanger design
Performance target
Design element
Approximate formula
Heat exchanger design
Designing the fan speed, which is a design element
Design element

Fig. 6 Prototype of impact analysis across design

In order for this method to be widely used in actual development, the information extracted from the system model must be in a format that is easy for each designer in charge to understand. In examining the format, we aimed to make it possible to use the method without understanding special notation, to give an overview of the relationships between the designs of the entire system, and to easily understand the extent of the impact of changes.

To meet this goal, a prototype of this method was created using Python *3, which has a complete set of XML parsing and drawing libraries. Figure 6 shows the results of the prototype. By inputting the results of each design into the design elements by GUI (Graphical User Interface) operation, we have confirmed that the extent of the impact on the entire design is indicated by an emphasized line and the magnitude of the impact can be quantitatively indicated.

This prototype uses a format that enables intuitive understanding of the impact of design results using natural language and block diagrams. We believe that all developers will be able to grasp the impact of their own design content, and that it will be possible to easily adjust the system design.

3. Conclusion

By linking the system model that defines relationships between designs to the detailed design models, we have designed a method that enables anyone involved in development to easily perform an impact analysis across design fields. We have confirmed the possibility of using the prototype for development.

In future actual development, we will apply this method to cases where there are problems in adjustment between design fields and show that development can be made more efficient.

References

(1) INCOSE Systems Engineering Handbook, A Guide for System Life Cycle Process and Activities 4th Edition, International Council on Systems Engineering (2015)

(2) K. Ishibashi: “Misunderstandings” and “expectations” of development sites surrounding Model-Based Systems Engineering (MBSE), Ansys Executive Forum (2021)

(3) H. Nishimura: A Practical Guide to SysML, The Systems Modeling Language, Tokyo Denki University Press (2012)

(4) Cybernet Systems Co., Ltd.: MBD-CAE solutions starting from MBSE, Automotive Engineering Exposition (2019)

[3] Python is a registered trademark of the Python Software Foundation.

Framework for Evaluating Sensor Attack Resistance

Authors: *Koichi Shimizu*, Hisashi Mori*,*
*Ryo Muramatsu*** and *Daisuke Suzuki****

1. Introduction

Autonomous systems such as self-driving cars use various sensors to understand the surrounding environment and control the system accordingly. The AEB-equipped car described in this paper uses radar, cameras, and LiDAR to detect the object in front, measure its position and velocity, and if necessary, warn the driver or automatically apply the brakes. However, there have been many reports of attacks on sensors, and there are also known examples of attacks affecting not only individual sensors but also autonomous systems that use sensors. Therefore, it is essential to ensure the safety of autonomous systems against sensor attacks.

In the case of self-driving cars, it is not realistic to conduct real-world driving tests over tens of billions of kilometers(1) required for safety verification, so evaluations are generally conducted by simulating various driving scenarios. However, the challenge is to narrow down the scenarios to be evaluated from among the countless possible scenarios. Furthermore, there have been few reports on simulators that can evaluate the impact of sensor attacks at the system level.

This paper proposes a framework for exhaustively identifying sensor attack scenarios that should be evaluated using techniques based on STAMP/STPA analysis(2) and evaluating the impact of sensor attacks on autonomous systems through simulation. It also describes the results of developing and evaluating a prototype simulator.

2. Evaluation Framework based on SOTIF

International standards related to automotive safety include the functional safety standard ISO 26262(3) and the SOTIF (Safety Of The Intended Functionality) standard ISO 21448(4). Functional safety aims for a state in which there is no risk caused by defects such as failures. SOTIF is a relatively new safety concept that supplements functional safety, aiming for a state in which there is no risk due to intended functionality or performance limitations. ISO 21448 takes into consideration sensors that advanced functions such as autonomous driving rely on but does not include security perspectives. ISO/SAE 21434(5) is an international standard for automotive security, but it focuses on security risk management in the development process and does not address specific threats such as sensor attacks.

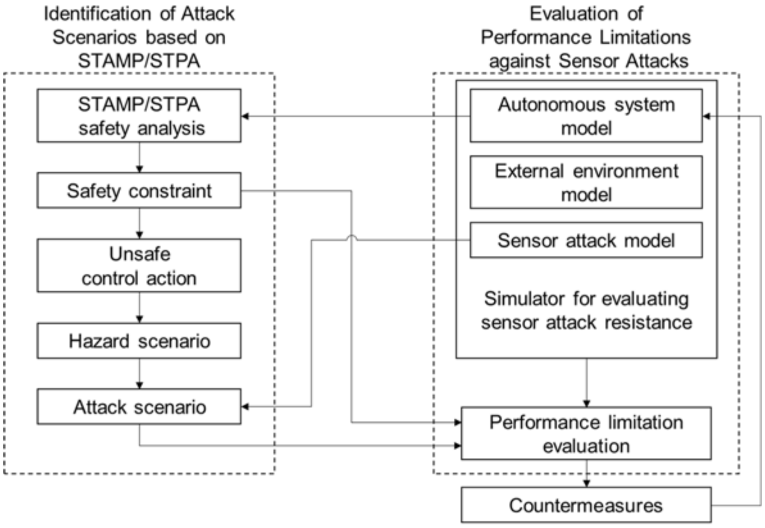
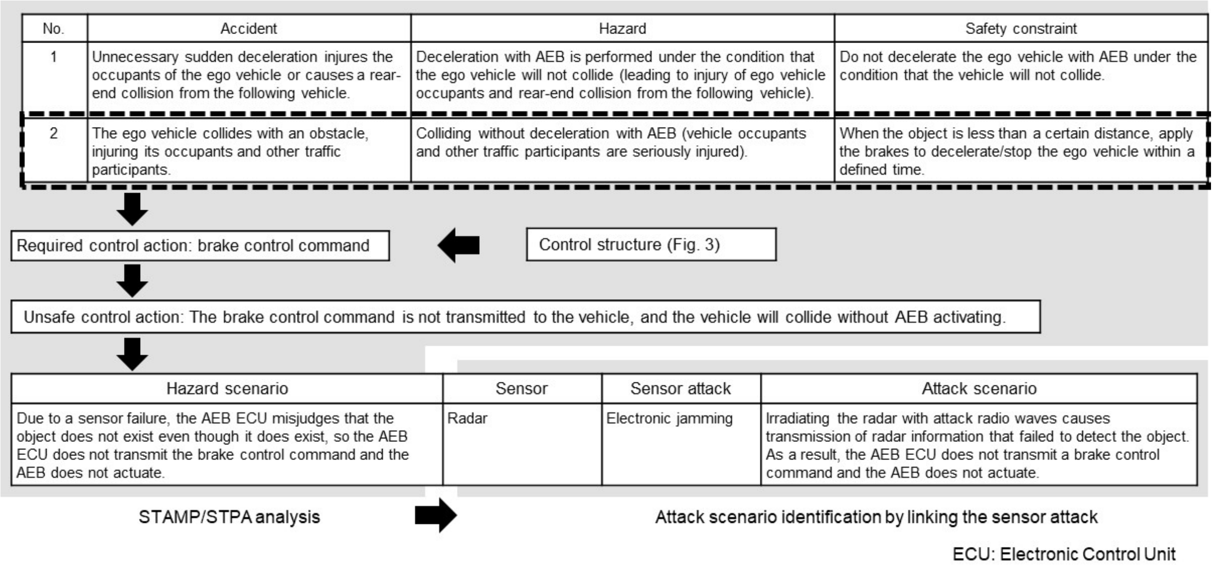

Identification of Attack Scenarios based on STAMP/STPA
STAMP/STPA safety analysis
Safety constraint
Unsafe control action
Hazard scenario
Attack scenario
Evaluation of Performance Limitations against Sensor Attacks
Autonomous system model
External environment model
Sensor attack model
Simulator for evaluating sensor attack resistance
Performance limitation evaluation
Countermeasures

Fig. 1 Framework for evaluating sensor attack resistance

**Information Technology R&D Center **Automotive Electronics Development Center*
****Corporate Information Security Div.*

ISO 21448 stipulates a process for correcting specifications and improving SOTIF by inputting system specifications and then identifying and evaluating specification deficiencies and performance limitations that may lead to injury or other damage. The process is suitable for evaluating the extent to which an autonomous system that incorporates sensors can withstand sensor attacks and reflecting the evaluation results in sensor design. Therefore, this paper proposes an evaluation framework for sensor attack resistance based on the improvement process of SOTIF (Fig. 1). By identifying scenarios that lead to damage using STAMP/STPA analysis and linking the results to sensor attacks, the sensor attack scenarios that should be considered in evaluating performance limitations are exhaustively identified. When evaluating performance limitations, a simulator for evaluating sensor attack resistance that incorporates sensor attacks and the external environment into the autonomous system is used to evaluate the performance limitations against sensor attacks under each attack scenario. The details of each are described in Chapters 3 and 4.



No.	Accident	Hazard	Safety constraint
1	Unnecessary sudden deceleration injures the occupants of the ego vehicle or causes a rear-end collision from the following vehicle.	Deceleration with AEB is performed under the condition that the ego vehicle will not collide (leading to injury of ego vehicle occupants and rear-end collision from the following vehicle).	Do not decelerate the ego vehicle with AEB under the condition that the vehicle will not collide.
2	The ego vehicle collides with an obstacle, injuring its occupants and other traffic participants.	Colliding without deceleration with AEB (vehicle occupants and other traffic participants are seriously injured).	When the object is less than a certain distance, apply the brakes to decelerate/stop the ego vehicle within a defined time.

Required control action: brake control command

Control structure (Fig. 3)

Unsafe control action: The brake control command is not transmitted to the vehicle, and the vehicle will collide without AEB activating.

Hazard scenario	Sensor	Sensor attack	Attack scenario
Due to a sensor failure, the AEB ECU misjudges that the object does not exist even though it does exist, so the AEB ECU does not transmit the brake control command and the AEB does not actuate.	Radar	Electronic jamming	Irradiating the radar with attack radio waves causes transmission of radar information that failed to detect the object. As a result, the AEB ECU does not transmit a brake control command and the AEB does not actuate.

STAMP/STPA analysis

Attack scenario identification by linking the sensor attack

ECU: Electronic Control Unit

Fig. 2 Example of attack scenario identification for AEB-equipped car

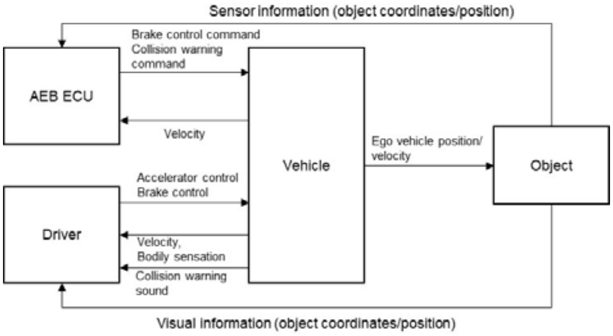

Sensor information (object coordinates/position)
AEB ECU
Brake control command
Collision warning command
Velocity
Vehicle
Ego vehicle position/ velocity
Object
Driver
Accelerator control
Brake control
Velocity,
Bodily sensation
Collision warning sound
Visual information (object coordinates/position)

Fig. 3 Control structure of AEB-equipped car

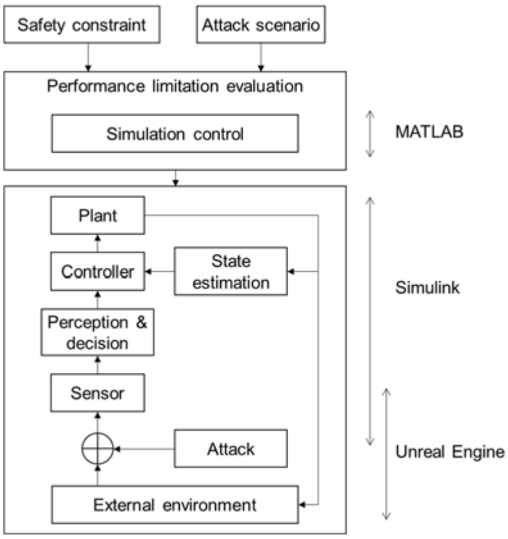
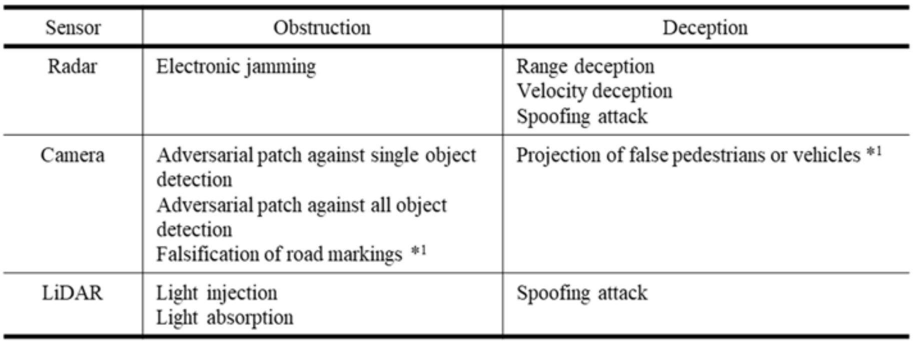


Fig. 4 Simulator for evaluating sensor attack resistance

Table 1 Supported sensor attacks



Sensor	Obstruction	Deception
Radar	Electronic jamming	Range deception Velocity deception Spoofing attack
Camera	Adversarial patch against single object detection Adversarial patch against all object detection Falsification of road markings *[1]	Projection of false pedestrians or vehicles *[1]
LiDAR	Light injection Light absorption	Spoofing attack

[1] Not supported in simulator

3. Identification of Attack Scenarios based on STAMP/STPA Analysis

Damage to be prevented, such as injury and economic loss, varies depending on the system, and is referred to as an accident in this paper. In addition, a system state in which an accident is latent is referred to as a hazard. Fault Tree Analysis (FTA), Failure Mode and Effect Analysis (FMEA), STAMP/STPA, etc. are existing techniques for identifying hazard scenarios in which systems succumb to hazards. FTA and FMEA mainly focus on hazards caused by component failures, whereas STAMP/STPA assumes that hazards can occur due to unintended interactions between components even if there are no component failures. In this paper, STAMP/STPA is adopted because the main focus is not on failures but on the interactions between AEB control and the vehicle based on sensor information.

Figure 2 shows an example of attack scenario identification for an AEB-equipped car. First, as a rough structure for realizing safety in systems, we define accidents, hazards, and safety constraints for controlling hazards. An example of the definitions is shown at the top of Fig. 2. In addition, the components (subsystems and devices) involved in the realization of safety constraints and the interactions between the components (control actions and feedback data) are analyzed and organized into a control structure as shown in Fig. 3. Next, from the control structure, the control actions required to enforce the safety constraints are identified and unsafe control actions leading to hazards are identified. Then, the hazard scenario leading to hazards is identified for each unsafe control action.

Finally, the sensor attack scenario is identified by linking the sensor failure in the hazard scenario with the sensor attack that caused it. The purposes of sensor attacks are classified into two types: obstruction and deception when detecting objects, and the sensor attacks include currently known attacks on radar, cameras, and LiDAR (Table 1).

In the analysis results of Fig. 2, “Brake control command” is identified as a control action necessary for implementing the safety constraint, “When the object is less than a certain distance, apply the brakes to decelerate/stop the ego vehicle within a defined time,” and an unsafe control action leading to hazards is identified, “The brake control command is not transmitted to the vehicle, and the vehicle will collide without AEB activating,” thereby identifying the corresponding hazard scenario and attack scenario.

4. Evaluation of Performance Limitations against Sensor Attacks

4.1 Simulator for evaluating sensor attack resistance

Figure 4 shows the structure of the simulator for evaluating sensor attack resistance. The prototype in this paper implements MathWorks’ MATLAB/Simulink *[1], which is widely used in model-based design, as a platform, and models related to the external environment are implemented using Epic Games’ Unreal Engine *[2]. It is assumed that models developed in each domain will be used for the plants, controllers, state estimation, perception & decision, and sensors that make up the autonomous system. This time, we combined sample models provided by MathWorks to create a model of an AEB-equipped car based on radar, cameras, and LiDAR. The external environment model makes it possible to evaluate the impact of various sensor attacks on a running vehicle by covering the driving scenarios of the AEB test specified in the Japan and European NCAP (New Car Assessment Programme) and combining them with the exhaustive attack scenarios identified in the STAMP/STPA analysis. The attack model is unique to Mitsubishi Electric, and the supported sensor attacks are shown in Table 1.

[1] MATLAB and Simulink are registered trademarks of The MathWorks, Inc.

[2] Unreal Engine is a registered trademark of Epic Games, Inc.

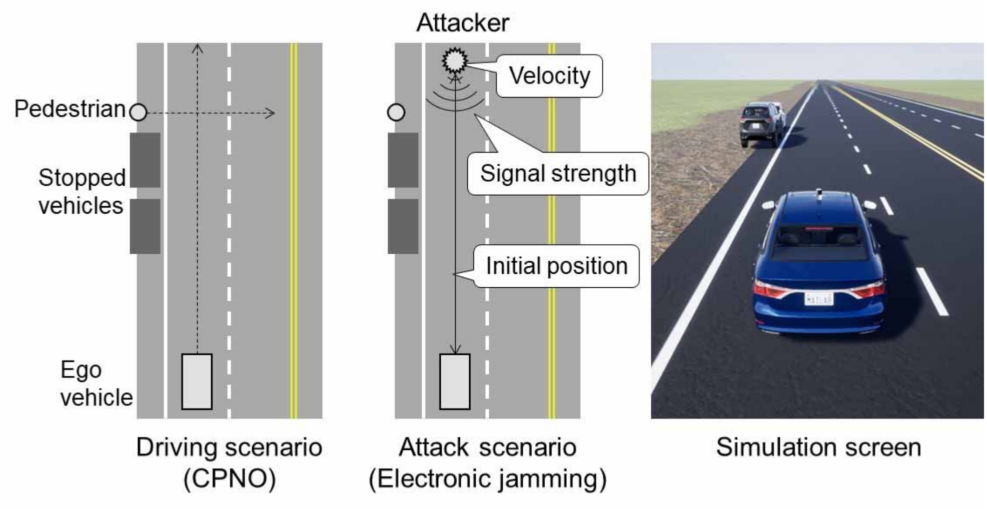


Fig. 5 Example of electronic jamming in CPNO test scenario

4.2 Example of simulation evaluation

An example of evaluating sensor attack resistance is shown for the M-out-of-N algorithm, which is a process for eliminating false detections in object detection with sensors. This algorithm confirms detection when the same object is detected at least M times out of the results of N consecutive times of detection.

Qualitatively, the accuracy of detection results increases as M is increased and brought closer to N, but the accuracy becomes more susceptible to noise. Since (M, N) is considered a design parameter determined by the performance requirements that the system must meet, the safety of the system against sensor attacks should also be considered.

Using Japan's NCAP CPNO (Car-to-Pedestrian Nearside Obstructed) as an example of a driving scenario, the attack scenario by electronic jamming of radar, which is identified in Fig. 2, is superimposed on the driving scenario (Fig. 5). The position of the attacker is fixed in front of the ego vehicle, and the initial position and signal strength are varied to evaluate the safety of the system. This evaluation is performed for different (M, N) and the results are compared. Figure 6 shows the evaluation results for (M, N) = (2, 2) and (9, 12). First, both results show reasonable results that the closer the attacker is to the vehicle and the stronger the signal strength of the attacker, the more likely the vehicle is to collide. Next, regarding sensor design parameters, it is confirmed that (M, N) = (2, 2) is generally safer but there are cases where the brakes are applied too early, that is, the brakes are applied at a timing when there is no risk of collision with a pedestrian. This indicates that the other safety constraint shown in Fig. 2, "Do not decelerate the ego vehicle with AEB under the condition that the vehicle will not collide" should be considered.

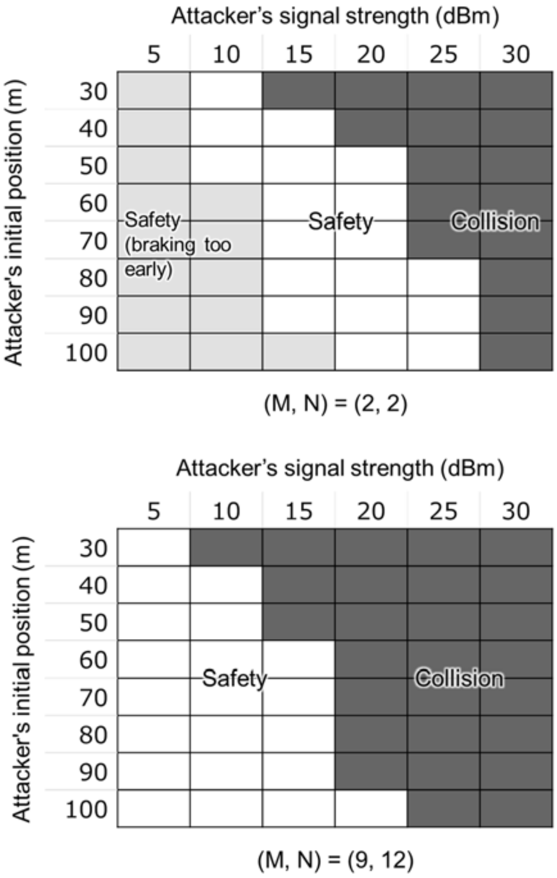


Fig. 6 Evaluation example with respect to attack parameters

5. Conclusion

In this paper, we proposed a framework for evaluating the sensor attack resistance of autonomous systems, and by taking an AEB-equipped car based on radar, cameras, and LiDAR as an example, we identified a sensor attack scenario through STAMP/STPA analysis and showed an example of evaluation using a prototype simulator. As autonomous systems become more complex, the interconnections between different simulators will be important in the future. Therefore, we will consider modularization using techniques such as Functional Mock-up Unit (FMU) to enable the developed sensor attack model to be used for other simulators.

This work is partially based on results obtained from the project (JPNP16007) commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

References

(1) Nidhi, K., Paddock, S.M.: Driving to Safety: How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability?, RAND Corporation (2016)

(2) Nancy, L., John, T.: STPA HANDBOOK (2018)

(3) ISO, Road vehicles - Functional safety. Standard ISO 26262: 2018 (2018)

(4) ISO, Road vehicles - Safety of the intended functionality. Standard ISO/FDIS 21448: 2022 (2022)

(5) ISO, Road vehicles - Cybersecurity engineering. Standard ISO/SAE 21434:2021(E) (2021)

A Digital Twin Approach to Diagnostics in Equipment Maintenance

Authors: *Yusuke Kaneki**, *Takashi Kikuzawa**, *Hitomi Yoshimura**,
*Nobuyuki Miyake** and *Junji Otani**

1. Introduction

To solve diversifying social issues, Mitsubishi Electric is working to provide integrated solutions that combine the strengths of both inside and outside the Group by consolidating its management foundation, which has been cultivated over the 100 years, and transforming its business model. We aim to create value by multiplying our strong components with our extensive field knowledge and advanced digital technology to realize a vibrant and comfortable society.

This paper describes a Digital Twin approach to diagnostics in equipment maintenance that utilizes Digital Twin as an example of technical development that supports integrated solutions. This technique will turn Mitsubishi Electric's field knowledge, such as equipment design knowledge and operational know-how, into formal knowledge through IoT and Digital Twin to achieve consistency and efficiency in diagnosing the causes of failures in equipment maintenance. This will reduce downtime for customers and contribute to solving social issues such as the declining birthrate and aging population, and the declining working-age population.

2. Social Issues Surrounding the Manufacturing Industry

According to the Ministry of Internal Affairs and Communications' WHITE PAPER Information and Communications in Japan(1), Japan's working-age population is expected to decline due to the declining birthrate and aging population. The ratio of the working-age population to the total population is estimated to decrease from 59.1% in 2020 to 53.9% in 2040, which will have a major impact on the Japanese economy and society. In addition, according to "World Population Prospects" of the United Nations Department of Economic and Social Affairs, the working-age population is declining worldwide: Europe, North America, East and Southeast Asia will face the same challenges as Japan.

Under these circumstances, the shortage of human resources and the aging population are having a major impact on the manufacturing industry as well. In particular, maintenance work, such as the inspection of equipment and diagnosis/repair of failed equipment, requires advanced skills and knowledge, but it also relies heavily on the tacit intuition and experience of skilled technicians. Due to the aging of skilled technicians, it has become difficult to pass on their skills to young technicians.

3. Issues at the Equipment Maintenance Site

At the equipment maintenance site, after an equipment failure occurs, maintenance staff go to the site to diagnose external and internal phenomena occurring in the equipment while referring to sensor data and other sources to identify the cause of the failure. In addition to using manuals, etc., the diagnostic work takes into consideration the installation environment and equipment state of each customer, so the design knowledge, experience, and intuition of maintenance staff have a significant impact on work quality. Therefore, although efforts are being made to train maintenance staff and to pass on technical knowledge, various issues arise when considering global service deployment, including differences in distance, language, and customs. As it becomes more difficult to secure human resources in the manufacturing industry, new initiatives utilizing advanced digital technologies such as AI, IoT, and Digital Twin will be necessary to maintain consistent work quality in the global market.

4. Use of Digital Twin for Equipment Maintenance Sites

"Digital Twin" refers to the concept of creating a real-world twin in digital space. The Digital Twin is attracting much attention in all industries, but the manufacturing industry in particular is expected to reap great benefits from it. With the introduction of IoT in recent years, efforts have been made to analyze and utilize the data collected from equipment and production lines in the cloud. By inputting the equipment state collected in real time from the equipment in this way into a model that reproduces the behavior of the equipment, the equipment and its state can be reproduced in digital space, almost the same as in the real world. This can be used to improve the quality of equipment maintenance work, improve work efficiency, reduce prototyping costs, improve after-sales service, and so on. For example, when some

**Information Technology R&D Center*

abnormality occurs in the equipment or manufacturing line during equipment maintenance, the state of the equipment or manufacturing line in digital space can be analyzed to identify the cause of the abnormality in real time. The difference from conventional data analysis is that the internal state of equipment is reproduced. The Digital Twin of equipment uses design knowledge to create a model of the equipment that reflects the real-world state. This makes it possible to analyze and explain in detail why the abnormality occurred using design knowledge. Thus, in the Digital Twin of equipment, it is important to build models using the design knowledge of equipment, thereby enabling solutions to be built that take advantage of the strengths of manufacturing companies. Therefore, we are developing a technique for applying the Digital Twin to equipment maintenance sites.

5. Digital Twin Approach to Diagnostics in Equipment Maintenance

5.1 Overview of diagnosis technique in equipment maintenance

Mitsubishi Electric has developed a technique for diagnosing equipment that utilizes Digital Twin to solve issues at equipment maintenance sites. This technique involves collecting sensor data representing the equipment state from the failed equipment and estimating the cause of the failure based on the sensor data. This technique leads to consistent quality of maintenance work and improves the efficiency of the work itself. Sensor data are values obtained by sensors installed inside and outside the equipment and are used for equipment control and maintenance. For example, sensor data varies depending on the equipment, such as temperature, current value, vibration, and pressure. This technique mainly uses sensor data that can be obtained from outside the equipment to estimate the causes of failures.

The technique consists of a model that reproduces the equipment state and a function for diagnosing equipment (Fig. 1). Using actual sensor data and the model, which reproduces the equipment failure state with Digital Twin, the function for diagnosing equipment calculates the candidate causes of the failure and their likelihoods. Maintenance staff receive these calculated candidate causes and start maintenance service on the equipment from the most probable cause. As a result, the function assists diagnosis based on design knowledge, leading to consistent work quality among maintenance staff. Furthermore, the sensor data from equipment in the real environment is constantly collected and stored in the cloud, making it possible to instantly diagnose the equipment using this data when a failure occurs. By the time maintenance staff arrive at the site, the results of the diagnosis have already been obtained, and they can quickly perform diagnostic work while referring to the results on PCs or tablet devices.

5.2 Equipment state reproduction model

The equipment state reproduction model maintains the relationship between the causes of failure and the sensor data representing the behavior of equipment during the failure (Fig. 2). It consists of three models: a physical quantity propagation model, a sensor value correlation model, and a state analysis model.

The physical quantity propagation model is a formula that expresses how a physical quantity such as the energy that drives equipment propagates between the components of the equipment. This model is used to simulate the behavior of equipment during normal conditions and during failure, and to acquire how the equipment's sensor data behaves during that time.

The sensor value correlation model retains the behavior of sensor data obtained from simulations using the physical quantity propagation model, i.e., the relationship between failures and the pattern of change in values. This model converts such knowledge into formal knowledge, such as the pattern of sensor data values that characterize a failure when it occurs.

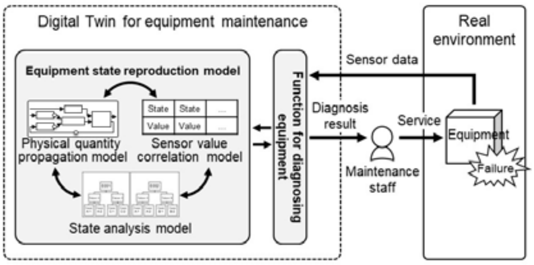

Digital Twin for equipment maintenance
Equipment state reproduction model
State
State
Value
Value
Physical quantity propagation model
Sensor value correlation model
State analysis model
Function for diagnosing equipment
Real environment
Sensor data
Diagnosis result
Service
Equipment
Failure
Maintenance staff

Fig. 1 Overview of diagnosis technique in equipment maintenance

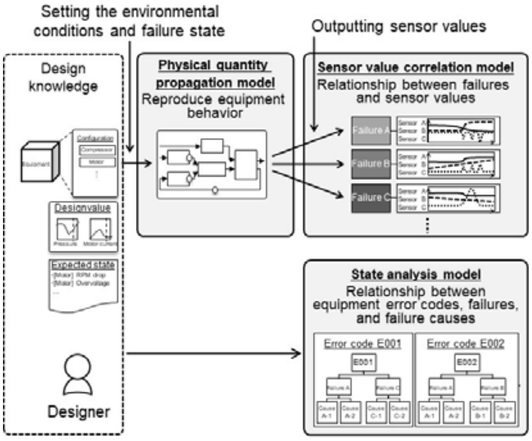


Fig. 2 Framework for equipment state reproduction model

The state analysis model expresses the phenomena that appear when equipment fails, and the relationship between error codes and failures. This model formalizes the knowledge of possible causes of failure when a certain error code occurs.

5.3 Function for diagnosing equipment

The function for diagnosing equipment estimates the cause of failure from the equipment state reproduction model and equipment sensor data (Fig. 3). This function inputs the error code obtained from the failed equipment and the sensor data before and after the failure and estimates its cause using the misdiagnosis reduction function and the sensor value pattern determination function.

The misdiagnosis reduction function uses the error code at the time of failure and the state analysis model included in the equipment state reproduction model to identify several causes of failure that could generate the error code. For example, if the error code is E001, failure A and failure C are extracted as candidate failure causes and passed to the sensor value pattern determination function. In this way, the misdiagnosis reduction function improves the accuracy of the sensor value pattern determination function by narrowing down the causes of failure from the error code using the state analysis model. On the other hand, there are cases where an error code is not generated depending on the failure condition. In that case, the misdiagnosis reduction function does not work, and only the sensor value pattern determination function is used for estimation.

The sensor value pattern determination function inputs the candidate failure causes and sensor data passed from the misdiagnosis reduction function and estimates the failure cause from the sensor value correlation model. Based on the candidates, a pattern of sensor data during failure is acquired for each candidate from the sensor value correlation model. The pattern obtained from the model is compared with the sensor data pattern of the failed equipment to calculate the degree of similarity. The investigation starts from the cause of failure with a higher degree of similarity, which enables more efficient diagnostic work.

6. Prototype System for Diagnosing Equipment

6.1 Overview of prototype system for diagnosing equipment

This cloud-based prototype system passes sensor data collected from equipment to the cloud, and presents the cause of failure and countermeasures to maintenance staff.

6.2 Cloud deployment model

This system consists of a model management platform, a service provision platform, and a data collection platform (Fig. 4). Each platform is built as a separate cloud system and is designed to be deployed globally across countries. The model management platform is a cloud system that manages the equipment state reproduction model. The service provision platform is a cloud system that provides maintenance staff with a function for diagnosing equipment. The data collection platform is a cloud system that constantly collects and stores sensor data from equipment actually in operation. The data collection platform and service provision platform are separated, which makes it easy to connect the service provision platform with existing data lakes and a new data collection platform. The model management platform is deployed in Japan, where models are created, and enables controls such as placing access restrictions on models and controlling export to other countries. The service provision platform and data collection platform are deployed in the country where the service is provided, taking into consideration the cross-border regulations on data that are being written into the data protection laws of each country, such as the General Data Protection Regulation (GDPR).

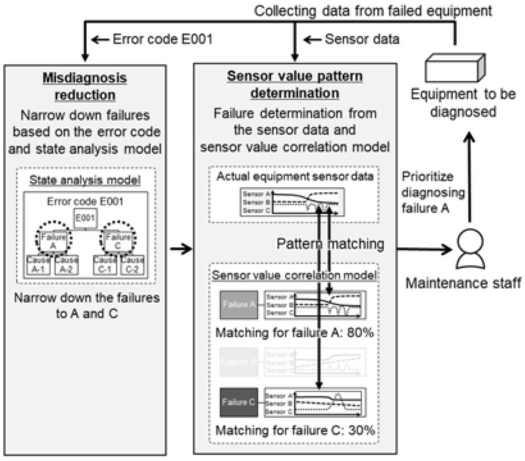

Collecting data from failed equipment
Error code E001
Sensor data
Misdiagnosis reduction
Narrow down failures based on the error code and state analysis model
State analysis model
Error code E001
E001
Failure A
Failure C
Cause A-1
Cause A-2
Cause C-1
Cause C-2
Narrow down the failures to A and C
Sensor value pattern determination
Failure determination from the sensor data and sensor value correlation model
Actual equipment sensor data
Sensor A
Sensor B
Sensor C
Pattern matching
Sensor value correlation model
Failure A
Matching for failure A: 80%
Failure B
Failure C
Matching for failure C: 30%
Equipment to be diagnosed
Prioritize diagnosing failure A
Maintenance staff

Fig. 3 Mechanism of function for diagnosing equipment

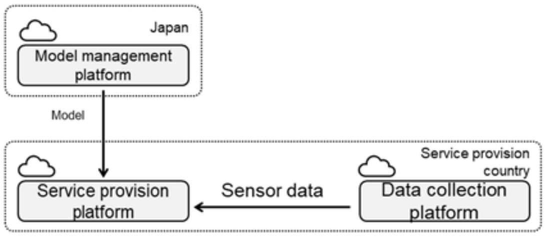

Japan
Model management platform
Model
Service provision country
Service provision platform
Sensor data
Data collection platform

Fig. 4 Cloud deployment model

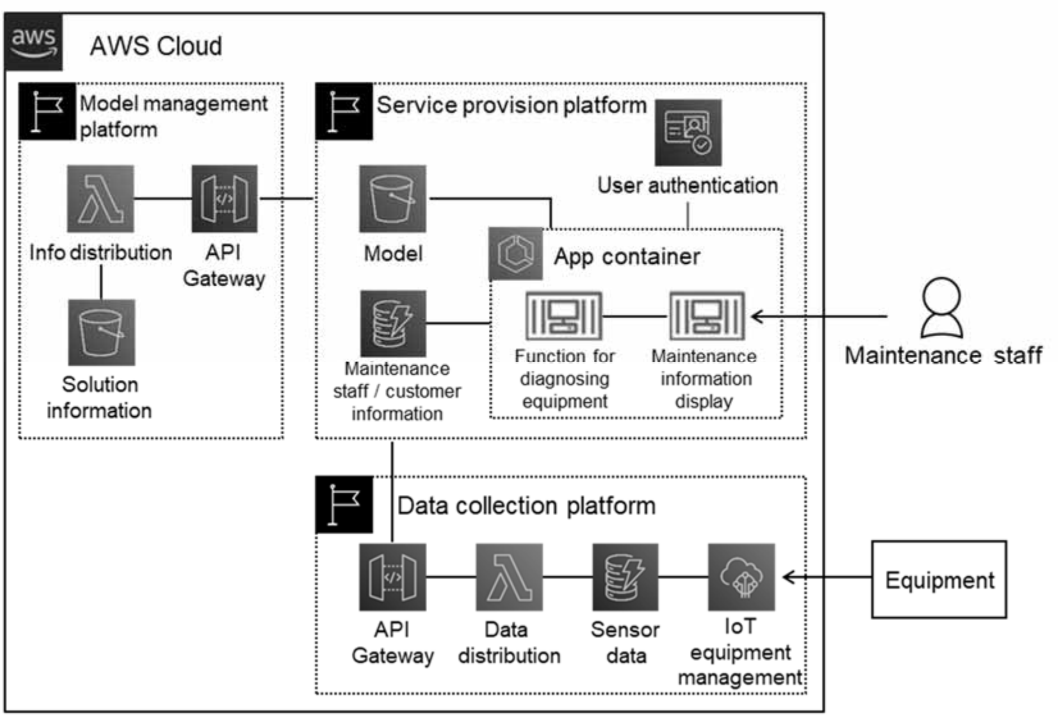


Fig. 5 Cloud architecture

6.3 Cloud architecture

This system is built as a cloud service on AWS *1 (Fig. 5). The model management platform, service provision platform, and data collection platform are each configured as a Virtual Private Cloud (VPC), and Application Programming Interface (API) Gateways are placed between the platforms to separate them. The function for diagnosing equipment and the maintenance information display function, which displays diagnosis results to maintenance staff, are configured as an application container so that they can be easily migrated according to the country where the service is provided. The maintenance information display function is a browser-based application that can be accessed from various terminals such as PCs and tablet devices by maintenance staff.

6.4 Building a prototype system

A prototype system was built for Mitsubishi Electric's equipment at the Information Technology R&D Center. Sensor data was acquired from the equipment and periodically stored in AWS. By installing a mechanism for generating failures in the equipment and diagnosing the equipment by generating specific failures, we confirmed that the causes of the failures can be estimated from the pattern of the actual sensor data and the sensor value correlation model.

7. Conclusion

This paper described a Digital Twin approach to diagnostics in equipment maintenance that utilizes Digital Twin as an example of technical development that supports integrated solutions. This technique turns field knowledge, such as equipment design knowledge and fault diagnosis know-how, into formal knowledge in the form of an equipment state reproduction model and reproduces the behavior of failed equipment as a Digital Twin. Then, sensor data during a failure is collected from equipment through IoT, and the cause of the failure is estimated from the equipment state reproduction model and sensor data. This initiative converts field knowledge into formal knowledge. We believe this technique can help solve industry and social issues such as the difficult of passing on skills due to the aging of skilled workers.

We will continue to promote technical development to realize integrated solutions that make use of our strengths, aiming to realize a vibrant and comfortable society.

References

(1) Ministry of Internal Affairs and Communications: WHITE PAPER Information and Communications in Japan 2021
https://www.soumu.go.jp/johotsusintokei/whitepaper/index.html

[1] AWS is a registered trademark of Amazon Technologies, Inc.

MITSUBISHI ELECTRIC CORPORATION