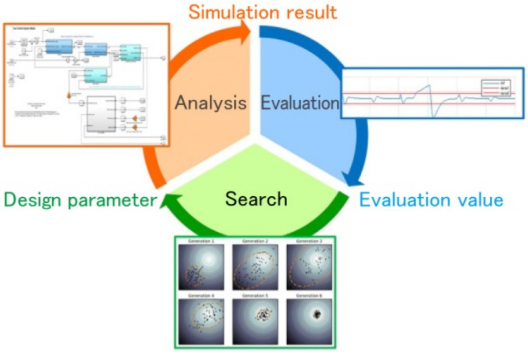
Automated Parameter Synthesis by Falsification Technique



ATSUYOSHI SAIMEN[*1] KENJI TAKAO[*2]

ICHIRO HASUO[*3]

Falsification technology expresses complicated requirements with a temporal logic formula and then searches for input patterns and operating conditions that do not satisfy the requirements using a simulator and an optimization solver. When applying this technology, there are issues of it being impossible to search for design parameters that meet multiple requirements at the same time and it being impossible to handle both quantity and time requirements such as in cases where both amount and time by which and for which the threshold is violated should be smaller. Mitsubishi Heavy Industries, Ltd worked on these issues in a joint research project with the National Institute of Informatics and as a result developed a method that can search for design parameters comparable to human design in a short time. As an application case, this method was used in the design of fuel control parameters when gas turbine load is shut off and this report presents the developed method.

1. Introduction

In setting design parameters for product design, simulation under multiple conditions is performed and the parameters that best meet the requirements such as performance are selected. However, depending on the product, it is necessary to determine a large number of design parameters so as to satisfy multiple conflicting requirements and a large number of man-hours are required for adjustment by experts through trial and error. As an automated adjustment technology for design parameters, methods using optimal solution search algorithms such as stochastic optimization and evolutionary computation have been proposed, but it is difficult to formulate objective functions and constraints in optimization calculations when complex requirements that contain the concept of time are included, such as "the threshold shall be violated once or more," "the threshold shall not be violated for T seconds or more," "after the variable X violates the threshold A, the variable Y shall always violate the threshold B," etc.

On the other hand, as a parameter search method that can handle complicated requirements, there is a method using falsification technology. Falsification technology expresses complex requirements in temporal logic and searches for input patterns and operating conditions that do not meet the requirements using a simulator and an optimization solver. This technology is also applied to search for design parameters that meet the requirements. However, parameter search technology that applies conventional falsification technology has issues of it (1) being impossible to search for design parameters that meet multiple requirements and (2) being impossible to handle requirements related to both amount and time period of violation, such as in cases where "both amount and time by which and for which the threshold is violated should be smaller". MHI developed a technology to solve these issues and used it to design fuel control parameters at the time when the load of the gas turbine is shut off as a test. As a result, it was confirmed that design parameters comparable to the results designed by experts could be automatically extracted in a short time.

Chapter 2 below presents an outline of falsification technology, chapter 3 covers the issues of

[*1] CIS Department, ICT Solution Headquarters, Mitsubishi Heavy Industries, Ltd.

[*2] Chief Staff Manager, CIS Department, ICT Solution Headquarters, Mitsubishi Heavy Industries, Ltd.

[*3] Associate professor, National Institute of Informatics

conventional technology, chapter 4 describes the developed technology, chapter 5 gives numerical examples using the developed technology and chapter 6 is the conclusion.

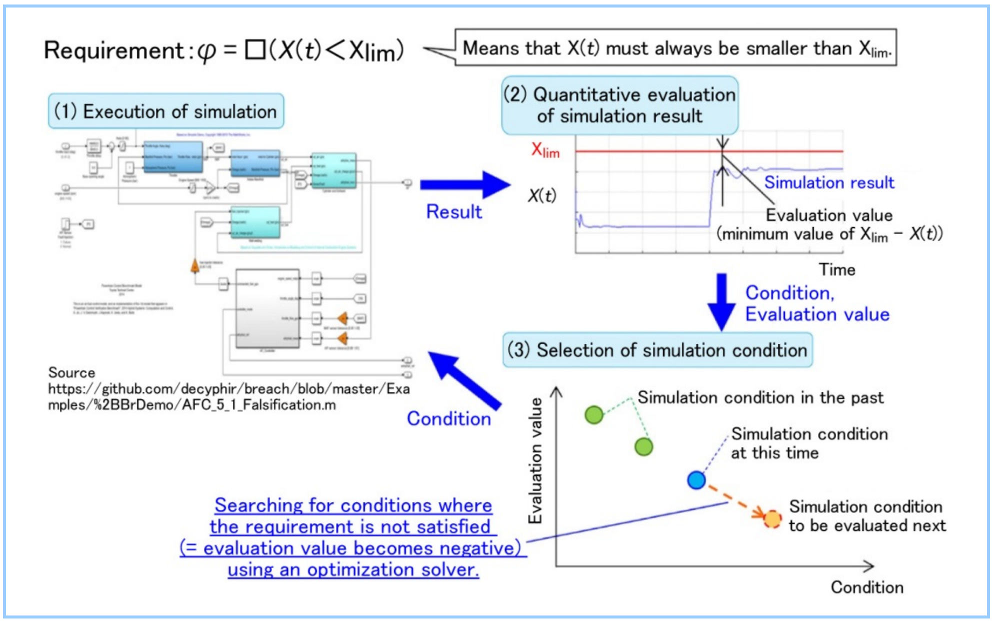
2. Falsification technology

2.1 Overview

Falsification technology, as shown in **Figure 1**, first formulates the requirement and then repeats the flow of (1) executing the simulation, (2) quantitatively evaluating the simulation result from the perspective of how well the requirement is satisfied and (3) selecting simulation conditions that reduce the margin for the requirement, to search for input patterns and operating conditions for which the evaluation value (margin) is negative, that is, for which it does not satisfy the requirement. Since the simulation target is dealt with as a black box, this technology can be applied even when the model is large and complicated and any simulator can be used as long as the acquisition of input and output is possible. For the selection of simulation conditions that reduce the margin with respect to the requirement, the conventional method applies an optimum solution search algorithm such as simulated annealing method.

Falsification technology also describes requirements according to a logical system called signal temporal logic. By using signal temporal logic, it is possible to handle requirements including the concept of time as described in the previous chapter. Furthermore, a method called Robust Semantics, which quantitatively evaluates how well the simulation result satisfies the requirements described in the signal temporal logic, has been proposed, which makes it possible to efficiently search for conditions that do not satisfy the requirements.

Falsification technology has already become widespread as a tool and in the automotive and aircraft industries, there have been multiple reports of cases where it was used to detect failure cases at the design stage.


Requirement : $\varphi = \Box(X(t) < X_{lim})$
Means that X(t) must always be smaller than X_{lim}.
(1) Execution of simulation
(2) Quantitative evaluation of simulation result
X_{lim}
$X(t)$
Result
Simulation result
Evaluation value (minimum value of $X_{lim} - X(t)$)
Time
Condition, Evaluation value
(3) Selection of simulation condition
Condition
Simulation condition in the past
Simulation condition at this time
Simulation condition to be evaluated next
Evaluation value
Condition
Searching for conditions where the requirement is not satisfied (= evaluation value becomes negative) using an optimization solver.

Source
https://github.com/decyphir/breach/blob/master/Examples/%2BBrDemo/AFC_5_1_Falsification.m

Figure 1 Overview of falsification technology

2.2 Application to parameter search

In Figure 1, the input pattern and operating conditions may be design parameters of the simulation model. Here, by setting the negation ¬φ of the requirement φ as a requirement and performing automatic verification, it is possible to search for a design parameter that does not satisfy ¬φ, that is, a design parameter that satisfies φ.

3. Issues of conventional technology

There are the following two issues in applying parameter search utilizing falsification technology.

3.1 Searching for design parameters that meet multiple requirements at the same time

When it is required to satisfy multiple requirements $\varphi_1, \cdots, \varphi_n$ at the same time, the requirement φ is described as $\varphi = \varphi_1 \wedge \cdots \wedge \varphi_n$. For design parameter search, as described in section 2.2, a design parameter that does not satisfy $\neg\varphi = \neg\varphi_1 \vee \cdots \vee \neg\varphi_n$, which is the negation of φ, is searched for. In the above expressions, $\wedge$ means an AND condition and $\vee$ means an OR condition. The quantitative evaluation result (hereinafter referred to as the evaluation value) of the simulation result for $\neg\varphi$ is defined as the largest of the evaluation values for each of $\neg\varphi_1, \cdots, \neg\varphi_n$. When multiple requirements are connected by $\vee$ as shown in **Figure 2**, even when a parameter that can make the evaluation value for $\neg\varphi_2$ smaller is found, the overall evaluation value is dominated by the evaluation value of $\neg\varphi_1$. As a result, it becomes impossible for the optimization solver to determine how to select the parameters to make the evaluation value of $\neg\varphi_2$ smaller and the search does not proceed. Such a problem is called a scale problem.

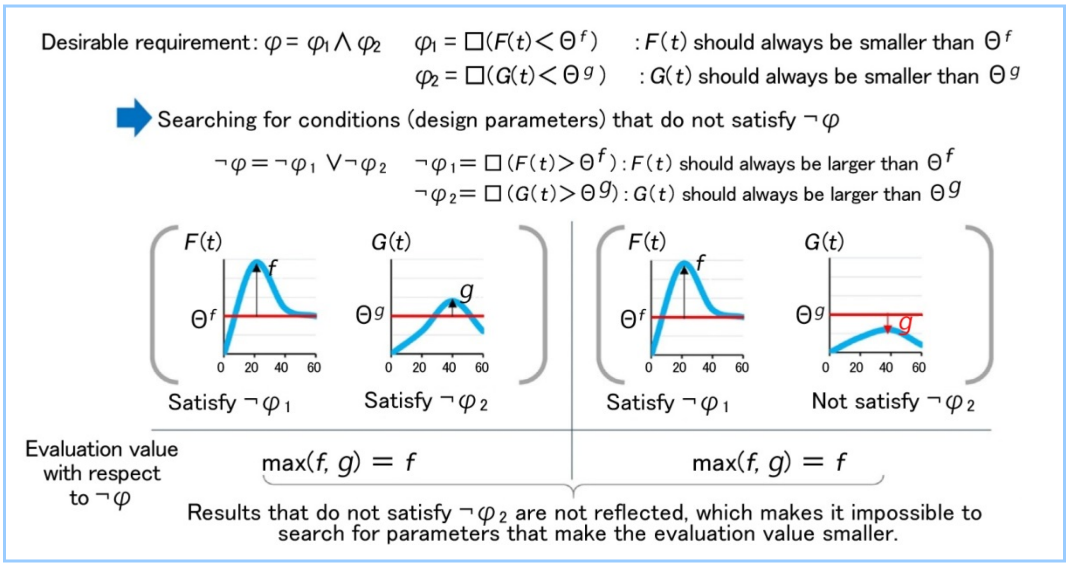

Desirable requirement: $\varphi = \varphi_1 \wedge \varphi_2$
$\varphi_1 = \square(F(t) < \Theta^f)$: $F(t)$ should always be smaller than Θ^f
$\varphi_2 = \square(G(t) < \Theta^g)$: $G(t)$ should always be smaller than Θ^g
Searching for conditions (design parameters) that do not satisfy $\neg\varphi$
$\neg\varphi = \neg\varphi_1 \vee \neg\varphi_2$
$\neg\varphi_1 = \square(F(t) > \Theta^f)$: $F(t)$ should always be larger than Θ^f
$\neg\varphi_2 = \square(G(t) > \Theta^g)$: $G(t)$ should always be larger than Θ^g
$F(t)$
f
Θ^f
0 20 40 60
$G(t)$
g
Θ^g
0 20 40 60
Satisfy $\neg\varphi_1$
Satisfy $\neg\varphi_2$
$F(t)$
f
Θ^f
0 20 40 60
$G(t)$
g
Θ^g
0 20 40 60
Satisfy $\neg\varphi_1$
Not satisfy $\neg\varphi_2$
Evaluation value with respect to $\neg\varphi$
$\max(f, g) = f$
$\max(f, g) = f$
Results that do not satisfy $\neg\varphi_2$ are not reflected, which makes it impossible to search for parameters that make the evaluation value smaller.

Figure 2 Example of scale problem

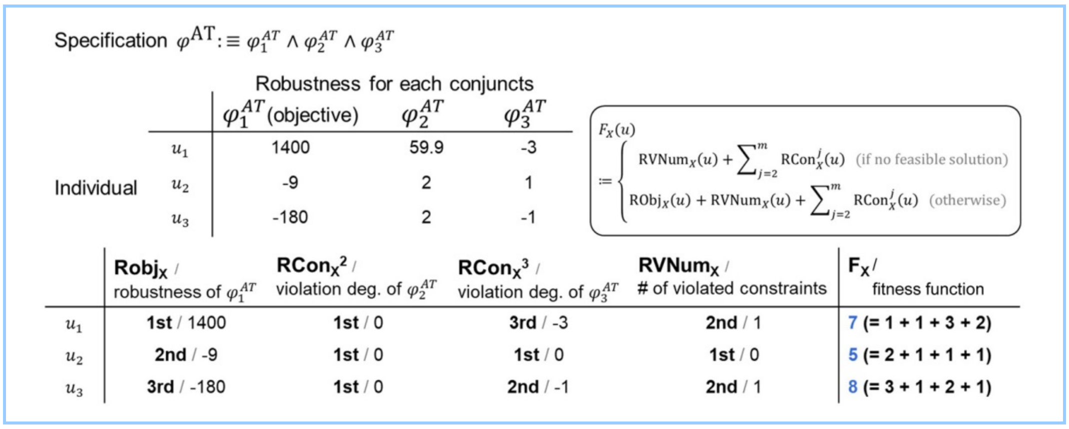
3.2 Consideration of requirement related to quantity and time

In falsification technology, it must be possible to describe requirements using signal temporal logic. When the requirement is related to both quantity and time, such as "both amount and time by which and for which the output value violates the threshold should be smaller", it cannot be described in ordinary signal temporal logic and the signal temporal logic needs to be extended.

4. Developed technology

4.1 Constrained optimal solution search using MCR (Multiple Constraint Ranking)

To handle the issue described in section 3.1, rather than minimizing the evaluation of all requirements, first the requirements are divided into objective functions and constraints to make them constrained optimization problems, which avoids scale problems. Then MCR (Multiple Constraint Ranking)(1) is applied in the solution search of the constrained optimization problem. **Figure 3** gives an overview of the MCR. The MCR is used together with evolutionary computation methods for the selection of superior individuals from many, such as PSO (Particle Swarm Optimization) and CMA-ES (Covariance Matrix Adaptation Evolutional Strategy), to determine the fitness, which is an index for individual selection, based on the ranking among individuals. This solves the scale problem between multiple requirements. Individuals that satisfy more constraints are determined to be superior and individuals that satisfy all the constraints and minimize the objective function are searched for around the individual.



Specification $\varphi^{\mathrm{AT}} :\equiv \varphi_1^{AT} \wedge \varphi_2^{AT} \wedge \varphi_3^{AT}$

Robustness for each conjuncts

		φ_1^{AT} (objective)	φ_2^{AT}	φ_3^{AT}
Individual	u_1	1400	59.9	-3
	u_2	-9	2	1
	u_3	-180	2	-1

$$F_X(u) := \begin{cases} \mathrm{RVNum}_X(u) + \sum_{j=2}^{m} \mathrm{RCon}_X^j(u) & \text{(if no feasible solution)} \\ \mathrm{RObj}_X(u) + \mathrm{RVNum}_X(u) + \sum_{j=2}^{m} \mathrm{RCon}_X^j(u) & \text{(otherwise)} \end{cases}$$

	Robj$_X$ / robustness of φ_1^{AT}	**RCon$_X^2$** / violation deg. of φ_2^{AT}	**RCon$_X^3$** / violation deg. of φ_3^{AT}	**RVNum$_X$** / # of violated constraints	**F$_X$** / fitness function
u_1	**1st** / 1400	**1st** / 0	**3rd** / -3	**2nd** / 1	**7 (= 1 + 1 + 3 + 2)**
u_2	**2nd** / -9	**1st** / 0	**1st** / 0	**1st** / 0	**5 (= 2 + 1 + 1 + 1)**
u_3	**3rd** / -180	**1st** / 0	**2nd** / -1	**2nd** / 1	**8 (= 3 + 1 + 2 + 1)**

Figure 3 Overview of MCR

4.2 Introduction of area function

To handle the issue described in section 3.2, while the simulation result violates the threshold as shown in **Figure 4**, the violating amount is integrated to obtain the area and an area function with this area used as an evaluation value is introduced into the signal temporal logic. Since evaluation values of the area function differ depending on the amount and time by which and for which the threshold value is violated, searching for a solution for which both amount and time by which and for which the threshold value is violated are small is made possible.

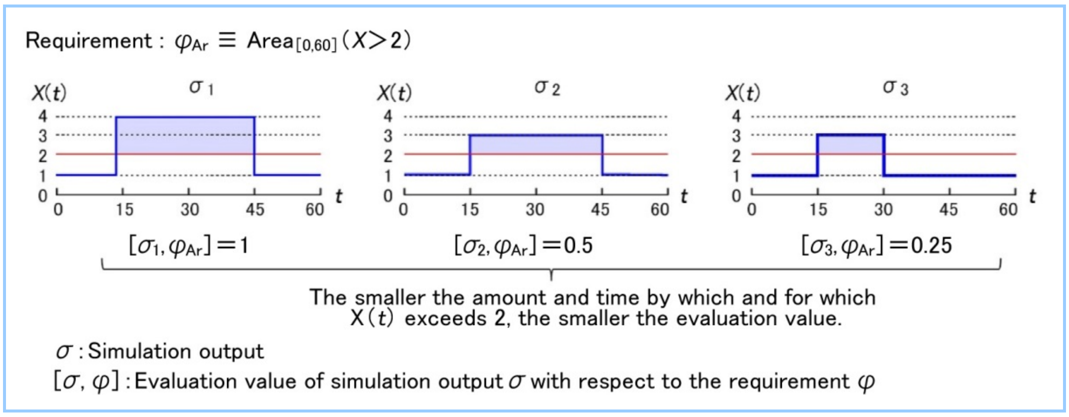
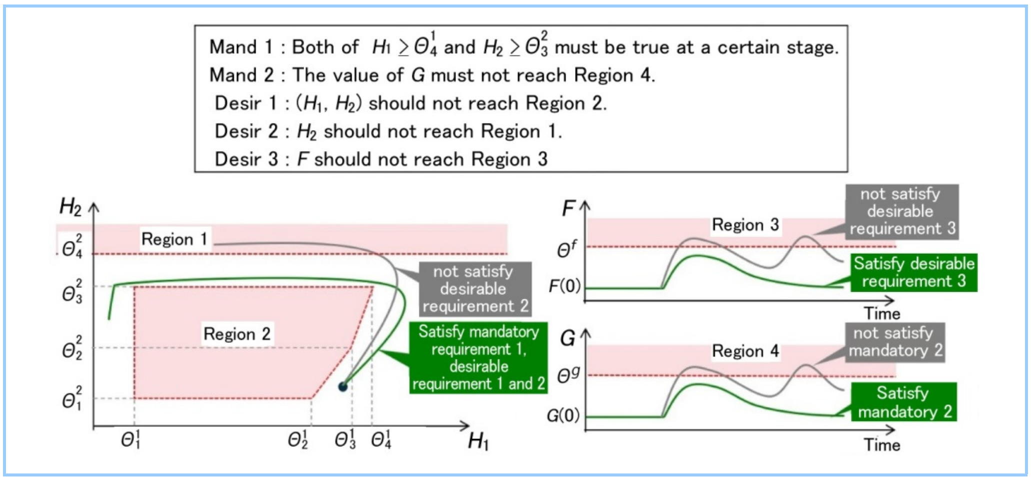


Figure 4 Area function

5. Calculation example

This chapter describes the design of fuel control parameters when gas turbine load is shut off, as an application example of design parameter search utilizing falsification technology. When a gas turbine is disconnected from the power system, the fuel is controlled according to a pre-designed time-series fuel supply schedule. In this time, it is necessary to change the flame temperature and rotor speed appropriately over time in order to prevent misfire phenomenon and equipment damage. **Figure 5** lists the requirements for this design. The requirements include those that must be satisfied (Mandatory 1, 2) and those that should be satisfied as much as possible (Desirable 1, 2, 3). H_1 and H_2 represent the flame temperature, F the valve opening, G the rotor rotation speed and Θ the threshold value.

A MATLAB/Simulink-based falsification tool, Breach,[2] was extended so as to use the MCR and area function and applied to the above design problem. **Figure 6** shows the results of simulations using parameters designed by experts through trial and error and parameters obtained by three-hour calculation (more than 7000 searches). The parameters obtained by the search satisfied the requirements for H_1, H_2 and G and only the threshold of F was violated for a short time. In this manner, results comparable to the expert designs could be obtained automatically without expert-level know-how.



Mand 1 : Both of $H_1 \geq \Theta_4^1$ and $H_2 \geq \Theta_3^2$ must be true at a certain stage.
Mand 2 : The value of G must not reach Region 4.
Desir 1 : (H_1, H_2) should not reach Region 2.
Desir 2 : H_2 should not reach Region 1.
Desir 3 : F should not reach Region 3

Figure 5　Requirements in parameter design

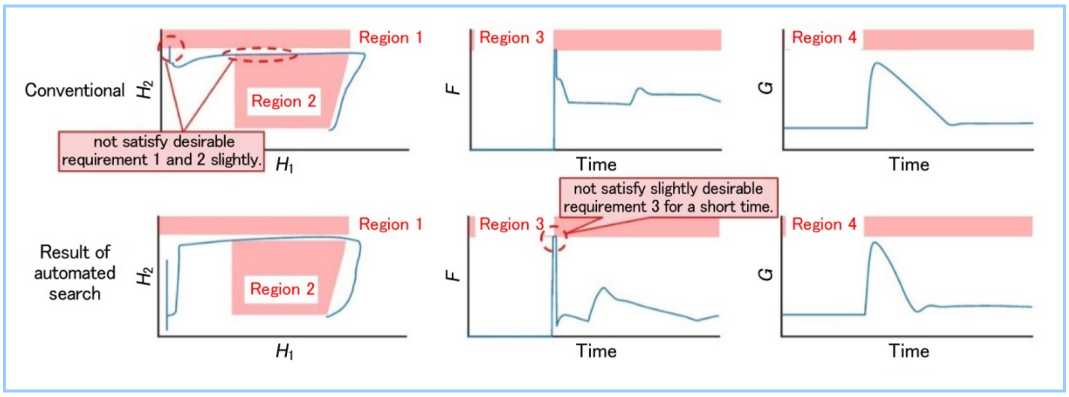


Figure 6　Search result

6. Conclusion

This report presented research on improving the design parameter search method applying falsification technology so that it can be applied even when there are multiple requirements or when requirements related to quantity and time are included. It also demonstrated the fact that the improved method can search for design parameters comparable to design by experts in the design of gas turbine fuel control parameters in a short time, which significantly reduces design man-hours.

For the expansion of the use of falsification technology to our other products, we are proceeding with its application to the design verification of refrigeration equipment for transportation. It is important to verify that there are no violations of the requirements at the design phase in terms of development cost reduction and quality assurance. We will continue to expand the application of this technology to other products in the future.

References

(1) de Paula Garcia et al., A rank-based constraint handling technique for engineering design optimization problems solved by genetic algorithms, Computers and Structures 187 (2017) p77-87

(2) Donzé, A., Breach, A toolbox for verification and parameter synthesis of hybrid systems, Proc. CAV 2010 (2010) p.167–170

(3) Sato S. et al., Hybrid System Falsification for Multiple-Constraint Parameter Synthesis: A Gas Turbine Case Study, Proc. Formal Methods. FM2021 (2021) p.313-329