

Automated Test Scheduling Using Vehicle Routing Problem Methodology

Atsushi ARISAKA¹

Takeshi MIYAZAKI¹

Kazuki FUKUI²

ABSTRACT

In order to shorten the development period and help reduce development costs for new vehicle models, a method for automated scheduling of actual vehicle tests was devised that helps to minimize the number of test vehicles and calculate the shortest test schedule. Use of mathematical optimization methodology for collision test scheduling and use of the vehicle routing problem solution in particular in mathematical optimization methodology for function test scheduling enabled incorporation of the constraints into the automated calculations to be considered. Simple applications were created and verified using the applications with a spreadsheet software. As a result, a scheduling task that previously took an expert 90 hours could be accomplished in 6 hours.

1. Introduction

Continuing advances in technologies such as CAE and performance design technology are enabling theoretical examination of many performance items⁽¹⁾⁻⁽³⁾. In addition to enhancing performance, automakers are also working to shorten development periods and use fewer test vehicles in order to suppress development costs.

However, it is not easy to estimate all performances, and some items are required by law to be confirmed using actual vehicles, and thus confirmation tests with actual vehicles are conducted prior to mass production.

The number of confirmation tests continues to rise due to tightening of laws and regulations and the diversification of specifications and equipment such as destinations, engines, transmissions, and control.

Confirmation tests using collision test facilities (hereinafter, collision tests) and performance confirmation tests that do not use collision test facilities (hereinafter, function tests) have been conducted with different vehicles.

However, in order to use fewer test vehicles, there are demands to use vehicles that have undergone function tests in collision tests.

Furthermore, adjustment of collision test facilities is becoming an issue in collision tests due to increasing numbers of models being developed at the same time, and adjustment of schedules is becoming an issue in function tests due to restrictions on the timing and order of various confirmation items.

Therefore, the test schedule is prepared in advance, all the function and collision tests are assigned to the test facilities and test vehicles, and whether the tests can be completed within the set time period (hereinafter, test deadline) is examined.

However, in actual vehicle development, some 1000 confirmation tests are conducted with around 50 test vehicles even for a full model change with relatively few variations, and testers often make individual requests for some function tests. As a result, it takes two weeks to a month for even an experienced person to create a test schedule, and test scheduling is an issue in new model development.

In existing studies, some methods have been reported to automatically create the shortest collision test schedule by giving the number of test vehicles, the number of confirmation tests, and the number of days and the order needed for each test as constraints^{(4), (5)}.

These methods use the mathematical optimization methodology in Operations Research (OR) to create a test schedule automatically. Figure 1 shows an example of a schedule for conducting four collision tests and some function tests with seven test vehicles.

In these methods, simple conditions can be considered such as conducting tests A and D first, but function tests cannot be conducted using the same vehicles after collision tests, and thus it is necessary to plan the function test schedule with different test vehicles.

In addition, complex conditions need to be considered in

¹ Honda Motor Co., Ltd.

E-mail: atsushi_arisaka@jp.honda

² FLECT Co., Ltd.

function tests, such as tests that need to be conducted first, tests with a decided order among multiple tests, and tests with timings that can be consolidated to shorten the test period.

This makes it a challenge to automatically create an overall schedule that includes function tests in addition to collision tests in actual development.

Therefore, this study examined a method to automatically create the shortest test schedule that takes into account all the tests, using the number of collision test facilities and the number of test vehicles as conditions for calculation, and given the respective numbers of collision tests and function tests, the number of days needed for each test, and the test order.

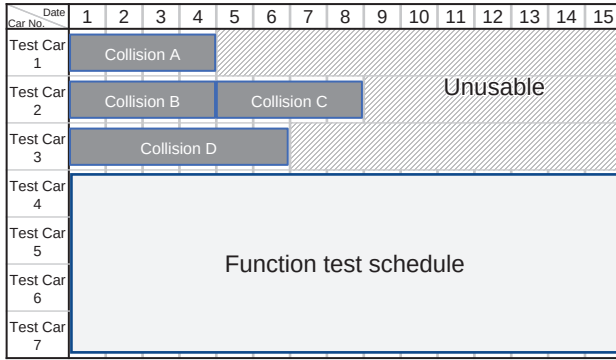


Fig. 1 Example of collision and function test schedule

2. Scheduling of Collision Tests

2.1. Helping Prevent Test Overlap

Multiple tests cannot be conducted at the same time with the same test vehicle. Therefore, the constraint to prohibit overlap of the periods of the tests conducted on each test vehicle is expressed by Eq. (1).

$$\text{Non-overlap}(\{I(s_t, P_t, x_{t,v})\}_{t \in \mathcal{T}}), \forall v \in \mathcal{V} \quad (1)$$

where $I(s, P, x)$ is an interval variable representing the period of the number of test days P from the test start time s , t is the test, s_t is a variable representing the start date and time of test t , P_t is a constant representing the number of test days of test t , $x_{t,v}$ is a binary variable that is 1 when test t is conducted with test vehicle v and 0 otherwise, $\mathcal{T}$ is the set of tests, and $\mathcal{V}$ is the set of test vehicles. Non-overlap($\cdot$) is a special constraint called a global constraint⁽⁶⁾, which does not permit overlap of the given interval variables.

This constraint helps to prevent multiple tests from being assigned to a single test vehicle at the same time.

2.2. Setting the Test Start Timings

Some tests can be conducted immediately after the test

vehicle is assembled, whereas others cannot be conducted until the necessary parts are ready.

If the date and time that test vehicle v becomes available is assumed to be the delivery date Q_v , Q_v needs to satisfy the following equation.

$$x_{t,v} = 1 \Rightarrow s_t \geq Q_v, \quad \forall t \in \mathcal{T}, \quad \forall v \in \mathcal{V} \quad (2)$$

The symbol " $\Rightarrow$ " in Eq. (2) represents the logical implication, and the constraint on the right side is satisfied when test t is assigned to test vehicle v , as indicated by the left side. In other words, this means that the start date and time of test t should be the delivery date and time of test vehicle v or later.

The constraint for the case that the test cannot be conducted unless the parts are ready is expressed by the following equation.

$$s_t \geq R_t, \quad \forall t \in \mathcal{T} \quad (3)$$

where R_t is a constant representing the time that test t can be started, and sets information such as the date and time that facilities become available and the date and time that the parts needed for the test are delivered.

These constraints enable setting of the timing at which the test can be started.

2.3. Consideration of Test Vehicle Damage

Some tests cannot be conducted using vehicles that have been damaged in other collision tests.

First, the constraint expressed by the following equation is added to achieve consistency among decision variables.

$$x_{t,v} = 1 \Rightarrow w_t = v, \quad \forall t \in \mathcal{T}, \quad \forall v \in \mathcal{V} \quad (4)$$

where w_t is a variable expressing the test vehicle assigned to test t .

Equation (4) indicates that when test t is assigned to test vehicle v , that test vehicle is set to w_t . Furthermore, by setting the constraint of Eq. (5), if tests t_1 and t_2 are to be conducted with the same test vehicle, t_2 can only be performed before t_1 in which the vehicle is damaged.

$$w_{t_1} = w_{t_2} \Rightarrow s_{t_2} + P_{t_2} \leq s_{t_1} \quad (5)$$

With these constraints, the damage condition of the test vehicles can be considered before assigning another test.

2.4. Optimization of Number of Test Vehicles

Equation (6) expresses the objective function representing the sum of the penalties in order to use the fewest test vehicles possible.

$$\min \sum_{v \in \mathcal{V}} C_v^V u_v \quad (6)$$

$$x_{t,v} \leq u_v, \quad \forall t \in \mathcal{T} \quad (7)$$

where C_v^V represents the penalty constant applied when test vehicle v is used in a collision test, and u_v is a binary variable that indicates whether test vehicle v was used in the collision test. The value of u_v is 1 if the vehicle v is used in the collision test and 0 otherwise. Under the constraint of Eq. (7), if any collision test is assigned to test vehicle v , $u_v = 1$ and the penalty in Eq. (6) becomes larger.

The above objective function and constraint enable helping to minimize the number of test vehicles.

2.5. Consideration of Test Deadlines

In order to use fewer test vehicles, it is desirable to conduct as many tests as possible with a single test vehicle before the test deadlines. This is achieved by setting an object function Eq. (8) that considers the test deadline.

$$\min \sum_{t \in \mathcal{T}} C_t^T o_t \quad (8)$$

$$s_t + P_t \leq D_t + o_t, \quad t \in \mathcal{T} \quad (9)$$

where C_t^T is a constant representing the penalty applied for the delay of test t from its test deadline, D_t is a constant representing the deadline for test item t , and o_t is a variable representing the delay of test t from its deadline.

Under the constraint of Eq. (9), the delay of each test is expressed as o_t . Equation (8) helps to minimize the total delay, enabling assignment of all the tests within the deadlines.

2.6. Order of Each Collision Test

Collision tests include tests such as for occupant protection, safety equipment, and low-speed collision, and these tests are conducted in a certain order. When conducting tests in that order, it is also necessary to set intervals of a certain time period or more, as shown by U in Fig. 2.

Therefore, the constraint expressed by Eq. (10) was added to provide an interval between the n th test t_n and the $(n+1)$ th test t_{n+1} .

$$s_{t_n} + U_{t_n, t_{n+1}} \leq s_{t_{n+1}} \quad (10)$$

where $U_{t_n, t_{n+1}}$ is a constant representing the length of the interval provided between the n th test t_n and the $(n+1)$ th test t_{n+1} , as shown in Fig. 2.

This condition enabled to fix the order of each collision test with the necessary intervals.

2.7. Consideration of Capacity of Collision Test Facilities

The capacity of collision test facilities is limited, making it necessary to establish an upper limit on multiple tests

using the same resources simultaneously.

Therefore, considering the total number of facilities to be used as a constraint results in Eq. (11).

$$\text{Cumulative}(\{I(s_t, P_t, 1)\}_{t \in \mathcal{T}_g}, L_g), \quad \forall g \in G \quad (11)$$

$\text{Cumulative}(\{I_t\}_t, L)$ is a constraint that upper bounds the overlap of the interval variables given by $\{I_t\}_t$ to L .

Here, G is the set of resources, and $\mathcal{T}_g \subset \mathcal{T}$ is the set of test items that use the resources $g \in G$.

While general resources can be expressed by Eq. (11), most collision test schedules consist of the three processes of preparation, hit, and post-process, as shown in Fig. 3. For this reason, the period occupying the collision test facilities is one day in the test duration. Assuming that up to two hits can be conducted in a single day, the constraint of Eq. (11) is specialized as shown in Eq. (12).

$$\text{Cumulative}(\{I(s_t + P_t^{\text{Prep}}, P_t^{\text{Hit}}, 1)\}_{t \in \mathcal{T}}, 2) \quad (12)$$

where P_t^{Prep} is Preparation in Fig. 3 and represents the preparation period of test t , and P_t^{Hit} is Hit in Fig. 3 and represents the collision test period.

Addition of this constraint enabled consideration of the capacity of test facilities.

2.8. Preserving Time for Function Tests

When using the same test vehicles in both collision and function tests, the function tests cannot be conducted after the collision test, making it necessary to conduct them before the collision test.

To achieve this, it is necessary to make the period r_t in

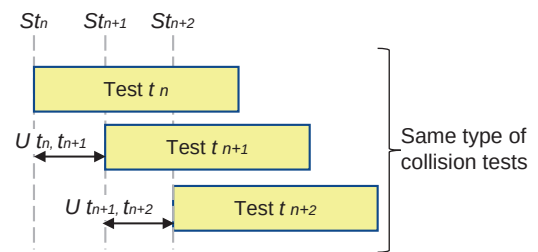


Fig. 2 Sequence of collision tests and start-time offsets

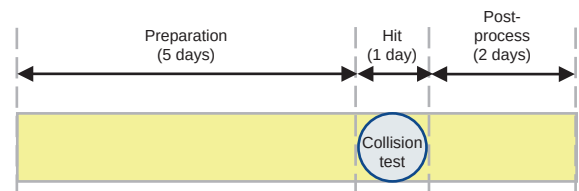


Fig. 3 Schedule for front collision tests

Fig. 4 smaller and shift the test completion time as much as possible toward the test deadline to increase the number of days available for the function tests before the collision tests. Therefore, the objective function and constraint expressed by Eq. (13) and Eq. (14) were set.

$$\min \sum_{t \in \mathcal{T}} C_t^R r_t \quad (13)$$

$$D_t - s_t - P_t \leq r_t, \quad \forall t \in \mathcal{T} \quad (14)$$

where D_t is the test deadline, r_t is a variable representing the reduction in the number of days available for other tests due to test t , and P_t is the duration of test t . In addition, C_t^R is a constant representing the penalty for the reduction in the number of days available for other tests.

This objective function and constraint enable assignment of the collision tests to help maximize the number of days available for the function tests before the collision tests.

Consideration of the above constraints and the objective functions in the developed OR model enabled automated collision test scheduling that preserves time for function tests such as shown in Fig. 5.

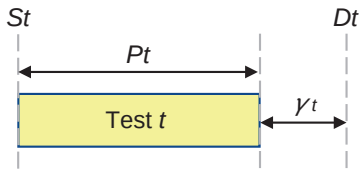


Fig. 4 Reduction of vehicle-days due to collision tests

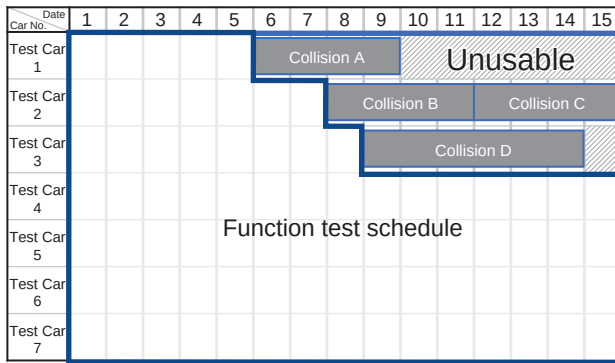


Fig. 5 Example of collision test schedule with days available for function tests

3. Scheduling of Function Tests

3.1. Approach

The task of assigning performance tests one by one to each test vehicle is thought to be similar to selecting the path on the transition network of test items conducted on a

certain test vehicle, as shown in the upper part of Fig. 6.

For example, by interpreting Start and End in Fig. 6 as nodes that represent the start and end of test vehicle usage and other nodes as test items, the path through the nodes from the Start node to the End node on this network can be seen as a test schedule such as shown in the lower part of Fig. 6.

The size of the penalty for each transition is set when considering this network. For example, in the case of transition within the same test category such as the transition from Test A to Test B in Fig. 7, there is no test vehicle handover work, and thus a smaller penalty is set. However, if the test category differs such as the transition from Test A to Test C or D, a larger penalty is set.

Use of this approach to find a test schedule that efficiently allocates each test to the test vehicles was found to be similar to the Vehicle Routing Problem (VRP)⁽⁷⁾, which is a mathematical optimization method that determines the route to be taken by a delivery vehicle to complete delivery by the size of the total penalty. Therefore,

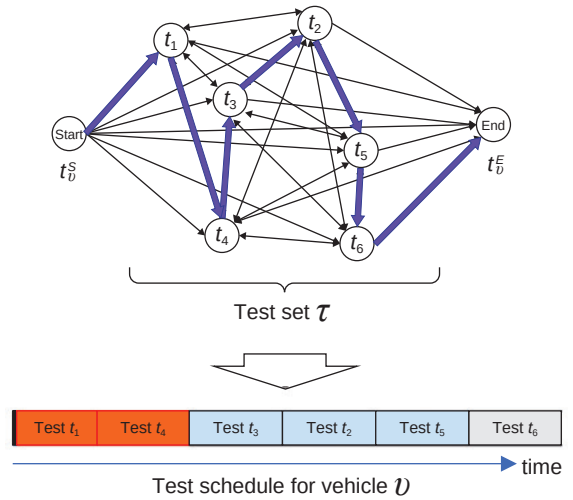


Fig. 6 Correspondence between test network and test schedule

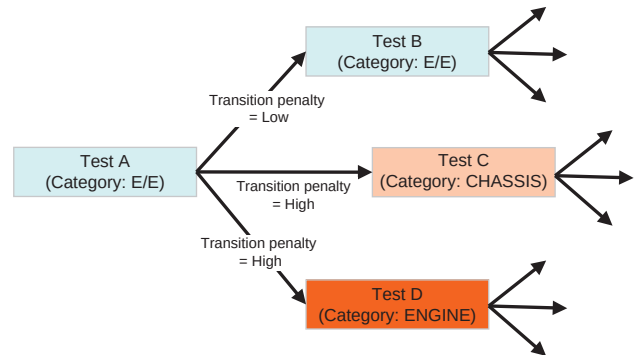


Fig. 7 Transition penalties

as a new approach, some fast solving methods for VRP were incorporated into function test scheduling.

3.2. Helping Prevent Test Overlap

Like the collision tests, multiple function tests cannot be conducted at the same time with a single test vehicle.

Equations (15) and (16) are constraints indicating that the transition is necessarily from a single test to another single test on the test network from the first test to the last test. Each test need to appear only once on the transition path from the first test to the last test on the test network.

$$\sum_{a \in \mathcal{A}_t^O} y_a = 1, \quad \forall t \in \mathcal{T} \cap \{t_v^S\}_{v \in \mathcal{V}} \quad (15)$$

$$\sum_{a \in \mathcal{A}_t^I} y_a = 1, \quad \forall t \in \mathcal{T} \cap \{t_v^E\}_{v \in \mathcal{V}} \quad (16)$$

where $\mathcal{A}_t^O$ is the set of directed links from the test t node, $\mathcal{A}_t^I$ is the set of directed links to the test t node, y_a is a binary variable that is 1 if passing through link a and 0 otherwise, t_v^S represents the start node of vehicle v , and t_v^E represents the end node of vehicle v .

This condition enables helping to prevent test overlap.

3.3. Setting the Test Start Times

Equations (17) to (20) express the passage of dates and times that tests are conducted and are constraints representing the consistency of the time change from the first test to the last test.

$$y_{(t_v^S, t)} = 1 \Rightarrow s_t \geq Q_v, \quad \forall t \in \mathcal{T}, \quad \forall v \in \mathcal{V} \quad (17)$$

$$y_{(t_1, t_2)} = 1 \Rightarrow s_{t_1} + P_{t_1} \leq s_{t_2}, \quad \forall t_1, t_2 \in \mathcal{T} \quad (18)$$

$$y_{(t, t_v^E)} = 1 \Rightarrow s_t + P_t = e_v, \quad \forall t \in \mathcal{T}, \quad \forall v \in \mathcal{V} \quad (19)$$

$$y_{(t_v^S, t_v^E)} = 1 \Rightarrow e_v = Q_v, \quad \forall v \in \mathcal{V} \quad (20)$$

where Q_v is the delivery time of vehicle v , and e_v is a variable representing the time at which vehicle v usage ends.

Of these, Eq. (17) is the constraint that the tests can be conducted only after delivery.

The above constraints enabled defining the start of the tests.

3.4. Consideration of Damage

It is not possible for some tests to be conducted after a collision test or after a function test in which the vehicle is damaged. First, the constraints of Eq. (21) to Eq. (23) define the consistency between the flow of the test vehicle on the test network and the variable w_t that represents the

assignment of the test vehicle to the test t .

$$y_{(t_v^S, t)} = 1 \Rightarrow w_t = v, \quad \forall t \in \mathcal{T}, \quad \forall v \in \mathcal{V} \quad (21)$$

$$y_{(t, t_v^E)} = 1 \Rightarrow w_t = v, \quad \forall t \in \mathcal{T}, \quad \forall v \in \mathcal{V} \quad (22)$$

$$y_{(t_1, t_2)} = 1 \Rightarrow w_{t_1} = w_{t_2}, \quad \forall t_1, t_2 \in \mathcal{T} \quad (23)$$

The above equations and the aforementioned Eq. (5) enable consideration of damage to test vehicles in test execution.

The above basic VRP constraints that define the test vehicle transitions between tests and the consistency of time⁽⁸⁾ enable assignment of all the tests to all the test vehicles, in consideration of helping to prevent test overlap, test start dates and times, and damaged test vehicles.

3.5. Handling of High-Priority Tests

Some function tests need to be performed as early as possible, including those that impact other test conditions such as aerodynamic tests, and those that impact other tests such as initial confirmation of engine performance.

The objective function and constraint expressed by Eq. (24) and Eq. (25) were set to conduct such high-priority tests as early as possible. Here, $\mathcal{T}^P$ is the set of high-priority tests, C_t^P is a constant representing the penalty applied for the delay of the start time of high-priority test t , B_t is a constant representing the earliest start time for high-priority test t , and o_t^P is a variable representing the difference between the start time s_t of high-priority test t and the earliest available start time B_t .

$$\min \sum_{t \in \mathcal{T}^P} C_t^P o_t^P \quad (24)$$

$$s_t - B_t \leq o_t^P \quad (25)$$

This objective function and constraint enable assignment of high-priority tests as early as possible.

3.6. Handling of Tests with a Decided Order

Some tests are conducted in a certain order such as Tests J and K in Fig. 8. For example, driving performance tests may be performed after making the suspension settings. In these cases, the order in which two tests t_1 and t_2 are conducted can be specified by precedence constraint of Eq. (26).

$$s_{t_1} + P_{t_1} + M_{t_1, t_2} \leq s_{t_2} \quad (26)$$

where M_{t_1, t_2} is a constant representing the time period between test t_1 and test t_2 .

Furthermore, when it is necessary to conduct tests t_1 and t_2 with the same test vehicle, the constraint of Eq. (27) is added.

$$w_{t_1} = w_{t_2} \quad (27)$$

Equations (26) and Eq. (27) enable assignment of consecutive tests to the same test vehicle.

3.7. Enhancing Efficiency by Consecutive Same Category Tests

As mentioned above, when the category that uses a test vehicle switches, additional workload is incurred for test vehicle handover between different categories. Therefore, the same test vehicle should be consecutively assigned to tests in the same category as much as possible. This corresponds to “Same category tests” in Fig. 8.

For this purpose, the objective function of Eq. (28) was set so that the proper penalty for the selection of transitions between tests is set to achieve the least switching between categories.

$$\min \sum_{a \in \mathcal{A}} C_a^A y_a \quad (28)$$

where $a = (t, t') \in \mathcal{A}$ is the set of directed links representing the transition from test t to test t' , and C_a^A is a constant representing the penalty incurred with the test transition represented by directed link a .

Less switching between categories can be achieved by setting the above objective function.

3.8. Test Completion Within the Deadline

All tests should be scheduled to be completed within the test deadlines. However, due to the large number of tests and the many constraints, not all tests may fit, and the initially planned test deadline may be exceeded as shown in Fig. 9.

In order to achieve the smallest total of the delay of each test from the test deadline, a condition to achieve the earliest overall end time was added to the objective function that considers the completion deadline of Eq. (8) which was also considered in the collision tests.

Date	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Test Car 1						Collision A				Unusable					
Test Car 2	High priority tests							Collision B		Collision C					
Test Car 3	Function E								Collision D						
Test Car 4															
Test Car 5								Function J		Function K					
Test Car 6	Function L T section				Function M T section			Function N T section		Function O T section					
Test Car 7	Function P														

Fig. 8 Arrangement of function tests

$$\min \max\{e_v\}_{v \in V} \quad (29)$$

This results in the earliest completion of all tests.

The VRP method was incorporated into function test scheduling, the basic VRP constraints that define the test vehicle transitions between tests and the consistency of time were set, and constraints and objective functions specific to function tests were added. This enabled automated function test scheduling that satisfies the conditions for function test transitions.

Date	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Test Car 1	Function A		Function B			Collision A									
Test Car 2	Function C			Function D				Collision B			Collision C				
Test Car 3	Function E		Function F		Function G					Collision D					
Test Car 4															
Test Car 5															
Test Car 6	Function L			Function M				Function N			Function O				
Test Car 7	Function P				Function Q					Function R				Function S	

Fig. 9 Example of schedule exceeding deadline

4. Verification

When actually performing calculations, it is necessary to give the necessary conditions, create an input/output function to check the calculation results, and create applications to execute the mathematical optimization calculations for OR and VRP.

To this end, spreadsheet software was used in the input/output interface to enable description of the test vehicle information, test information, and information of the conditions related to the test schedule.

It was decided to create two different applications for OR and VRP using a popular programming language. The programming language was the commonly used Python, and the calculations were executed with a computer (Xeon Platinum 8000 Series, clock frequency: 3.6 GHz, memory: 8 GB).

Figure 10 shows an example of the calculation result.

This is an example of a full model change with relatively few variations, but even so, 45 test vehicles and 931 tests are scheduled.

It previously took 90 hours or more for an expert to manually create such a test schedule, but when calculated on a computer with the aforementioned specifications, test scheduling could be accomplished in about 6 hours.

The calculation time is thought to vary depending on the number of test vehicles and the number of tests. However, as shown in Fig. 11, it was confirmed that automation of



Fig. 10 Calculation result

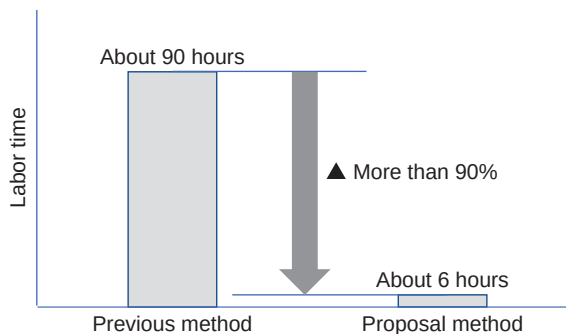


Fig. 11 Comparison between previous scheduling time and proposal scheduling time

test scheduling by the mathematical optimization methods of OR and VRP shortens the calculation time by 90% or more compared with manual scheduling by an expert.

5. Conclusion

This study examined a method to automatically create the shortest test schedule, using the number of collision test facilities and the number of test vehicles as conditions for calculation, and given the respective numbers of collision tests and function tests, the number of days needed for each test, and the test order. The results are summarized below.

- (1) A method was established that incorporates the mathematical optimization methods of OR and especially VRP to automatically create the shortest test schedule.
- (2) Definition of the constraints and objective functions of

collision tests by the OR method enabled automated collision test scheduling that preserves time for function tests.

- (3) Definition of basic VRP constraints that define the test vehicle transitions between tests and the consistency of time, and constraints and objective functions that represent the conditions between function tests enabled automated function test scheduling.
- (4) In order to verify automated test scheduling, automated test scheduling applications that consider all the test conditions of the collision tests and function tests were created.
- (5) Automated test scheduling was performed using existing vehicle model data and information, and it was confirmed that the scheduling time was shortened by 90% or more.

References

- (1) Toyoshima, T., Matsuzawa, T., Ito, K., Endo, Y., Takano, H., Ushimura, T.: Development of Chassis Performance Design Technologies for MBD of Mass-production Vehicles, Honda R&D Technical Review, Vol. 29, No. 1, p. 52-62
- (2) Fujimoto, S., Tojo, K., Kusunoki, S., Saito, T.: Proposal of Development Process Using DSM and Simulation, Honda R&D Technical Review, Vol. 29, No. 1, p. 106-116
- (3) Yasuhara, S., Yoshimoto, T., Okada, H.: Modeling and Utilizing Expert's Knowledge, Experience, and Thinking in Automobile Development, Honda R&D Technical Review, Vol. 32, No. 2, p. 91-98
- (4) Reich, D., Shi, Y., Epelman, M., Cohn, A., Barnes, E.,

- Arthurs, K., Klampfl, E.: Scheduling Crash Tests at Ford Motor Company, *Interfaces*, Vol. 46, No. 5, p. 365-462, (2016), doi: 10.1287/inte.2016.0855
- (5) Shi, Y., Reich, D., Epelman, M., Klampfl, E., Cohn, A.: An analytical approach to prototype vehicle test scheduling, *Omega*, Vol. 67, p. 168-176, (2017), doi: 10.1016/j.omega.2016.05.003
- (6) Beldiceanu, N., Carlsson, M., Rampon, J-X.: Global Constraint Catalog, HAL, 1406p., (2005), <https://hal.science/hal-00485396>
- (7) Braekers, K., Ramaekers, K., Nieuwenhuysse, I. V.: The vehicle routing problem: State of the art classification and review, *Computers & Industrial Engineering*, Vol. 99, p. 300-313, (2016), doi: 10.1016/j.cie.2015.12.007
- (8) El-Sherbeny, N. A.: Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods, *Journal of King Saud University - Science*, Vol. 22, Issue 3, p. 123-131, (2010), doi: 10.1016/j.jksus.2010.03.002

■ Author ■



Atsushi ARISAKA



Takeshi MIYAZAKI



Kazuki FUKUI