

Proposal of Development Process Using DSM and Simulation

Shigeki FUJIMOTO*

Keiji TOJO*

Shoji KUSUNOKI*

Toru SAITO*

ABSTRACT

A development process was created to perform request-type development without reworking by determining the target system performance and the means for achieving that performance from the initial development stage. This process uses a Design Structure Matrix to determine the most reasonable development order, performs simulations to predict whether the target system performance can be achieved, and comprehensively calculates alternate solutions in case the target performance is not achieved. Data mining technology is then applied to these results to quickly extract the most reasonable solution from among multiple solution candidates. By way of illustration, using power train system development as an example, performance was investigated for the areas of fuel economy performance, acceleration from standstill, and low-engine-speed booming noise, and it was verified that the system designers could narrow down the optimal balance solution.

1. Introduction

In vehicle development, power trains continue to diversify as destinations expand due to globalization. Furthermore, rising safety needs are resulting in the increasing inclusion of Active Safety systems, so both devices and control are becoming more complex at ever-faster rates. However, development periods and costs are limited, so there are demands for technology that enables efficient development of these types of multifarious and complex automobile systems.

The V-model is an approach that stratifies the design and test processes of the system to be developed according to the respective grades, and matches the verification levels⁽¹⁾. In addition, model-based development (MBD) utilizes models from the initial design process and front-loads the verification to be performed in the test processes⁽²⁾. Use of a MBD V-model that combines this V-model with MBD (Fig. 1) makes it possible to reduce concerns in initial development processes and perform development without reworking in later processes. Figure 1 shows the MBD V-model for an overall vehicle system. Requirements include customer requests, market trends, laws and regulations, and other overall vehicle demands. The system designer determines the target performances in accordance with these requirements, and designs subsystems and parts to meet those target performances. This approach is spreading to each automobile maker, and the MBD V-model has also been used to promote efficient device development

for Honda powertrain systems^{(3), (4)}.

To apply the MBD V-model to an overall vehicle system, it is necessary to coordinate multiple differing target performances arising from various demands, and to design subsystems and parts. That is to say, even if the design staffs for each subsystem and part perform efficient development on their own, reworking occurs when the integrated subsystems and parts do not satisfy the target system performance.

Figure 2 shows an image of current vehicle system development. Here, the project leader (system designer) of the new model car summons the persons in charge (PIC) of each subsystem area to meetings to investigate the target performance of the overall vehicle. At these meetings, each PIC presents the target performance values of the subsystems investigated separately to that point. However,

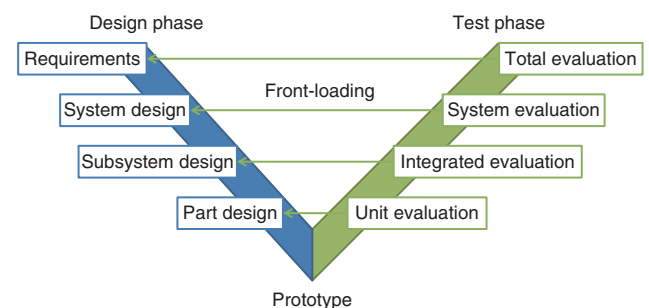


Fig. 1 MBD V-model

* Automobile R&D Center

when the target performance of the overall system is not achieved or a tradeoff relationship with another subsystem performance is discovered, the target performance values presented by each PIC remain pending, and the staffs of each subsystem area reinvestigate the target performances before the next meeting. This type of reworking is often repeated multiple times.

These issues in conventional system development stem from insufficient technical knowledge needed to achieve the target overall system performance and a failure to organize the relationships between the various performances. Target values with unclear prospects for achievement are merely nonbinding goals, so trial and error is repeated as long as the development period lasts. Conversely, if prospects for achievement can be reached from the start, the system designer can strategically choose the target performances from among the achievable Pareto optimal solutions, and request the needed performances from the subsystem staffs.

Therefore, to address the issues of conventional system development that involves much reworking, the following development process is proposed that enables determination

of the target system performance and the means for achieving that performance from the initial development stage.

- (1) Determine a reasonable development order.
- (2) Predict whether the target system performance can be achieved.
- (3) Search for other solutions in the event the target performance is not achieved with the initial investigation structure.
- (4) Quickly extract the most reasonable solution from among multiple solution candidates.

2. Overview of Development Process

Figure 3 shows an overview of the process proposed by this study. First, the method described in chapter 3 is used to sort the relationships between the target performances of the target system and the specifications and characteristics of each subsystem that comprise the target system in order to determine a reasonable development order. By modeling and visualizing each element and its dependencies, the order in which each element should be determined can be analyzed, and process enhancements can be proposed.

Next, the contents of the sorted relationships are clarified to predict whether the target system performance can be achieved. Specific methods and index values for determining characteristics, such as physical relationships, laws and regulations, trends, design procedures and timing, and designer knowledge and experience, are sorted based on the classification described in section 4.1. through hearings with designers, and the simulation environment is prepared.

However, the more complex the relationships between each system and subsystem performance, the harder it is to determine a development order completely free of reworking. Rework elements in Fig. 3 indicate the range of relationships between characteristics for which reworking occurs even when a reasonable development order is achieved through sorting. Reworking indicates the presence of a self-referencing relationship when obtaining a certain performance or characteristics value, where

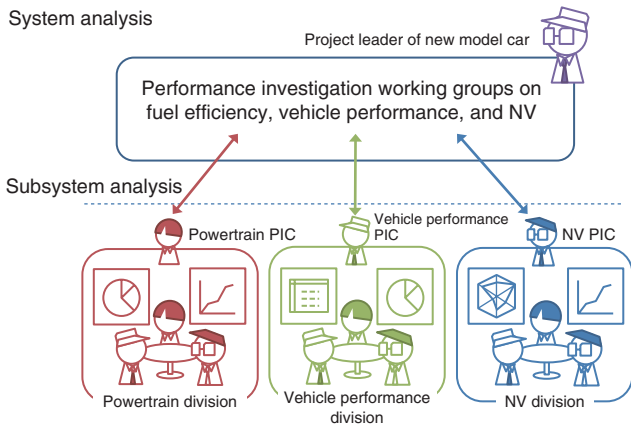


Fig. 2 Complexity of current product development phase

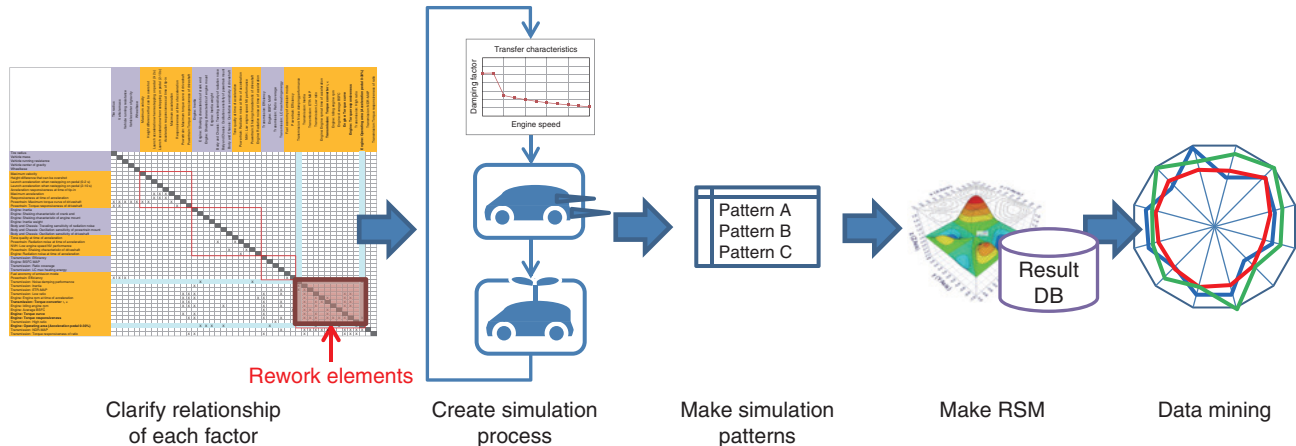


Fig. 3 Process overview

the preconditions for performing calculation include the characteristic value in question. Even when this kind of relationship exists, a solution can be obtained analytically as long as the calculation logic can be expressed by a simple equation such as shown in the upper part of the analytical approach in Fig. 4. However, when the logic includes a complex derived relationship such as the simulation shown in the lower part of the analytical approach in Fig. 4, it is not easy to analytically obtain a solution. In addition, since there are cases where no solution exists, or where multiple solutions exist, this paper performed solution search by statistical methods so that other solutions can be found when the target performance is not achieved. In the statistical approach in Fig. 4, when reworking occurs between characteristics, the process sets the initial conditions (initial values) and convergence conditions for the characteristics that are the rework factors, and searches for the initial values that satisfy those convergence conditions. However, when there are multiple rework factors, the search range becomes extremely large and there is no guarantee that the sought-after solution can be reached within the specified calculation time. In addition, there are also many cases in which the solution cannot be narrowed down automatically because of a tradeoff.

Therefore, the data mining technology developed by Ariyoshi et al.⁽⁵⁾ described in section 4.4. is used to quickly extract the most reasonable solution from among multiple solution candidates. The method developed by Ariyoshi et al. makes it possible to create a model by response surface methodology (RSM) that achieves accurate prediction from the calculation results, and that RSM can be used to efficiently search data in a short period of time.

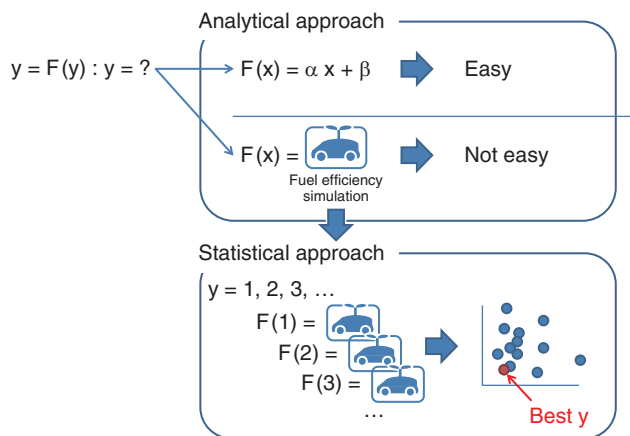


Fig. 4 Approaches toward of solution search

3. Sorting of Relationships Using DSM

A design structure matrix (DSM) is used to sort the relationships between the specifications and characteristics of each subsystem comprising the target system⁽⁶⁾. A

DSM is a simple method for performing both analysis and management of complex systems. Use of a DSM to model and visualize each element and the dependencies between elements of an arbitrary system enables analysis of the system configuration, which also facilitates management of each element. In addition, system enhancement proposals can be made by rearranging the order in which each element is determined based on these analysis results⁽⁷⁾.

Typical DSM modeling procedures are shown below. First, all elements comprising the target system are extracted and arranged equally in order of determination in the vertical and horizontal directions of a square matrix. Next, the elements in each column are input to the row elements that they influence with values indicating the respective influence dependency. This is called numerical DSM [Fig. 5(a)]. Rearranging the order of each DSM element enables sorting into a modularized element that reduces reworking, so this can be expected to effectively enhance the existing determination process for the target system.

A DSM is a matrix with the system and subsystem specifications, characteristics and target performances needed for development as elements, and is used to arrange the order in which each element is designed in order to optimize the development order. In the following chapters, this process is used to predict the system performance from the relationship between each element. However, these relationships cannot be expressed by simple values in the manner of the influence dependency needed by numerical DSM. Therefore, this paper used binary DSM [Fig. 5(b)], which models only simple dependencies, to determine the optimal development procedure. The detailed relationships were then determined separately as described in the following chapter. The DSM modeling procedure was performed as follows.

- (1) Arrange the elements needed for vehicle development in the same order in the vertical and horizontal directions of a square matrix.
- (2) When determining the vertical axis elements (outputs), mark the required horizontal axis elements (inputs).
- (3) Rearrange the element order to reduce the occurrence of marks in the upper right half of the matrix, while paying attention to performance modularization for the entire vehicle. Strive to minimize the range of the element group for which reworking cannot be avoided.
- (4) Use the order of the partitioned vertical axis elements as

Element	A	B	C	D
A				1
B	3		3	
C				
D	2		1	

(a) Numerical DSM

Element	A	B	C	D
A				x
B	x		x	
C				
D	x		x	

(b) Binary DSM

Fig. 5 DSM

the element determination order.

A typical system consists of multiple subsystems, and has a tree structure down to each individual part. If the level of each element extracted from the system differs greatly, subsequent description and partitioning of relationships becomes more complex, and arranging the elements becomes difficult. Therefore, it is necessary to extract elements on the same or higher levels as much as possible, and to show the relationships between these elements.

In addition, when setting the degree of influence with a DSM, it is not easy to describe relationships in a manner that takes all influences into account. While there may be relationships between each element, when the degree of influence is small, adjustments should be made to remove these small influences in order to reduce the DSM complexity. Furthermore, regarding elements for which the degree of influence is unclear in advance, the process should be adjusted (tailoring) in accordance with practical operations, by taking these relationships into account at the start of investigation, and then removing them from subsequent investigation.

There is no absolutely correct solution with regards to the partitioning of each element. Elements can be mechanically rearranged using a partitioning algorithm such as the reachability matrix method⁽⁸⁾ or identify loops by path searching⁽⁹⁾, but the approach toward modularization and the allowable rework range differ according to the target system, so DSM partitioning should reflect the intentions of the system designer.

As shown above, even if relationships can be sorted using a DSM, this does not mean mechanically extracting elements, adding relationships, and deciding the optimal determination order. The system designer and the staffs of each element should work together closely to construct and tailor the DSM.

However, even if the design order of each element to be developed can be sorted using a DSM, there will be element groups for which reworking cannot be avoided.

4. Derivation of Solutions by Data Mining

Technology was developed to perform predictive calculations in advance using various alternate solutions for element groups for which reworking cannot be avoided, and to quickly derive the most reasonable solution from among multiple solution candidates. Specifically, rework element group values are sampled over a wide range within the range of the design restrictions. Element group prediction simulations are then performed using these sampled values as the initial values, and interactive data mining is used to search for solutions from among the obtained multiple results.

4.1. Definition of Relationships between Elements

The relationships between each element were clarified and simulations were performed based on those

relationships to predict and visualize system performance.

The presence of influence between each element was demonstrated using a binary DSM that modeled only simple dependencies, but this DSM does not express the details of these relationships. Therefore, in order to clarify the relationship between each element, grouping was performed as shown by the specific examples below, and the inputs and outputs, calculation logic, and evaluation criteria needed to calculate numerical solutions were determined. Some of the groups (classifications) used to sort the relationships between elements are described below.

Constraint relationships refer to items determined as target values from market demands such as laws and regulations, comparison with competitors (benchmarking), and trends, and items with little or no freedom of design due to cost, production, and other limitations. Changes to these elements involve review of preconditions, so they have a large influence on other elements.

Subordinate relationships refer to items that have clear rules and principles and are determined uniquely according to the preconditions. These include items with clear decision algorithms, such as those determined according to simulation results.

Design policy and conditional relationships refer to items determined inversely from simulation results. These are items for which multiple solutions may appear, and for which the solution selection guideline is unclear.

Free design relationships refer to items determined more or less based on staff knowledge and experience, and from trends. These are items for which generalization is a challenge, and for which new relationships other than extracted items may occur.

As shown by the examples above, the basis for calculating each element is clarified by grouping the relationships between elements, making it easier to obtain the understanding of related parties when changes to calculation conditions occur.

4.2. Calculation of Each Element Value

Each element value is calculated in the order determined by the DSM. Regarding element groups for which reworking cannot be avoided, calculations are repeated multiple times based on design of experiments (DOE) among the elements comprising the largest loop. The values of the elements or element groups that are the reasons for reworking are sampled comprehensively within the design space as initial values, and these sample values are used to calculate subsequent element values using the previously determined calculation logic in the order determined by the DSM (Fig. 6).

The specific calculations handled here are performed at arbitrary levels ranging from simple spreadsheets to combinations of multiple analysis tools in conformance with the contents investigated beforehand in each performance area.

Each element is calculated in accordance with the DSM

determination order using the element values calculated in the previous stage and the initial values.

Each element value calculated using different initial values is linked and compiled into a database (DB), and visualization is performed using analysis tools in accordance with the purpose.

4.3. Method for Finding the Optimal Solution

The optimal solutions for the system designer and PIC are searched from each element compiled in the DB. Sorting is performed to minimize the range of the rework element group subject to optimization. However, when investigating overall system performance such as in this case, there are many inputs so multi-aspect issues cannot be avoided.

Various algorithms, ranging from analytical to statistical approaches, have been proposed to search for multivariate, multi-aspect optimal solutions⁽¹⁰⁾. However, when performing multi-aspect optimization with four or more dimensions, it is a challenge to obtain the optimal solution analytically within a practical number of calculation times⁽¹¹⁾.

Moreover, in actual design, reconciliation is performed based on implicit knowledge that is not easily quantified. For this reason, it is not easy to obtain the understanding of PIC with conventional methods that automatically search and narrow down the optimal solution.

Therefore, instead of a method that obtains the optimal solution analytically or statistically, this paper used data

mining technology that enables the system designer and PIC to derive the solutions to be obtained from among each element compiled in the DB in an interactive manner via a graphical user interface (GUI).

4.4. Solution Search by Interactive Data Mining

Figure 7 shows the interactive data mining flow proposed by Ariyoshi et al. The data search procedure is as follows.

- (1) An RSM is applied to the previously created calculation results DB to approximate the data within the design space in order to increase the solution density.
- (2) Grouping is performed to bundle the data in step (1) above into each related area.
- (3) The typical values of the grouping results for each area are shown in a radar chart, so the user selects the preferred group.
- (4) The data of the selected group is extracted, and grouping is performed again within this extracted data. The typical values of these results are shown in a radar chart.
- (5) The user repeats steps (3) and (4) to narrow down the solutions until the sought-after solution is obtained.

The radar charts for each area are linked in the DB, so when one radar chart is selected, the results are also reflected to the radar charts for other areas. In this manner the optimal solution is derived by narrowing down the data in consideration of the influence of each area on other areas.

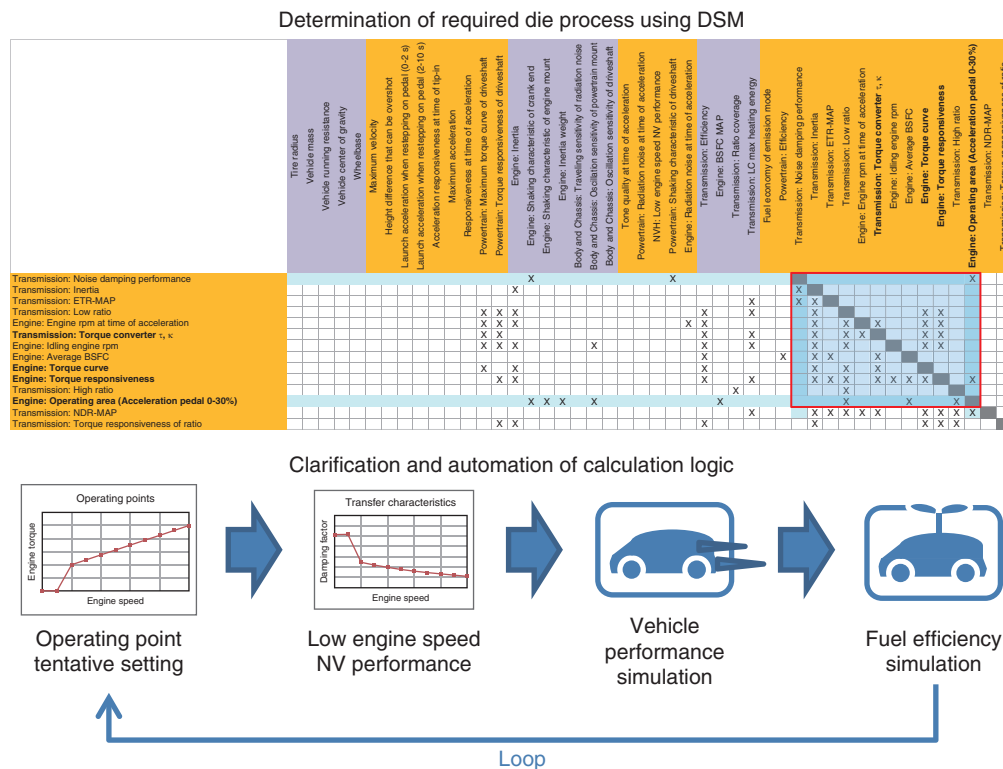


Fig. 6 Coordination process of DSM and simulation

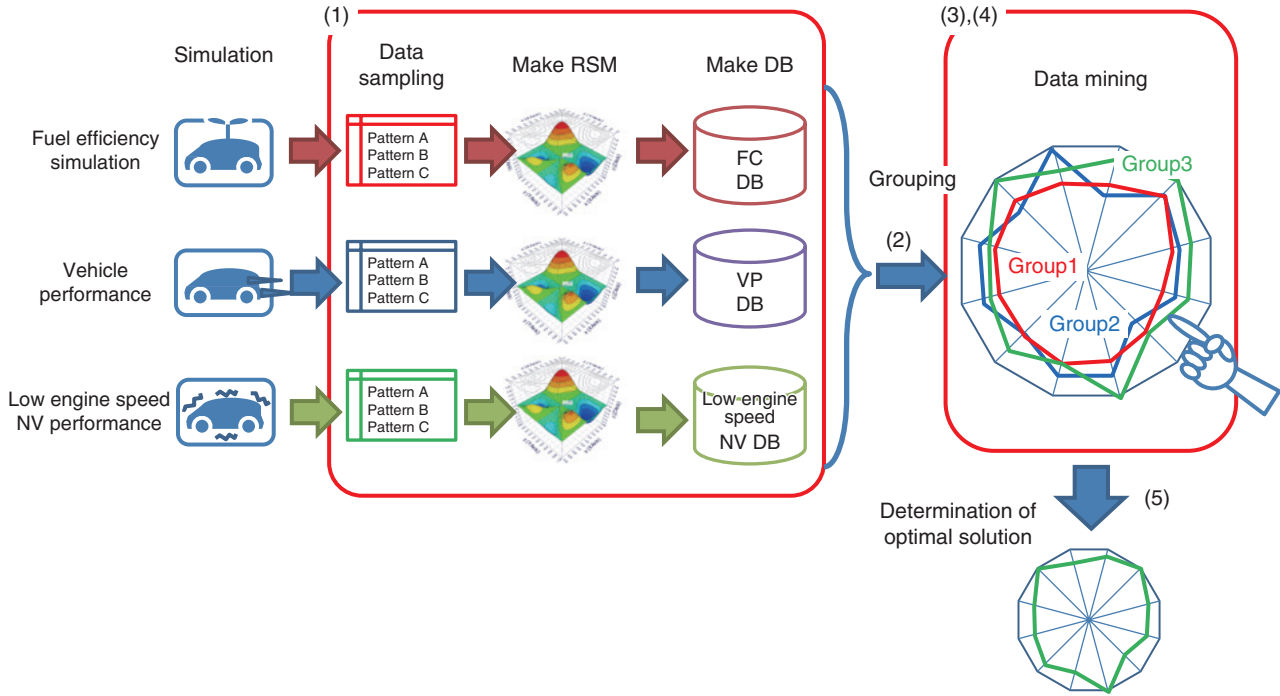


Fig. 7 Interactive solution search

5. Application to Actual Development

The process described in chapters 3 and 4 above was applied to a model currently under development and the effects were confirmed. The performance of a conventional engine vehicle was investigated using the proposed process for the areas of fuel economy, acceleration from standstill, and low-engine-speed booming noise. Figure 8 shows the DSM created by extracting the elements needed for the investigation.

The elements extracted in this investigation were narrowed down to typical items such as vehicle body specifications, running resistance, engine and transmission specifications and characteristics values, and control specifications in order to enable design and management in the initial stages of actual development. Elements with predetermined values set due to constraints other than extracted elements, that is to say elements with performance designs that cannot be changed, are indicated by gray. Conversely, elements that can be derived from the relationships with other elements in the DSM are indicated by orange. Gray elements are constraints, so these are basically shifted to the front part of the DSM, but there may also be cases where gray items are intentionally set in the middle part from the viewpoint of modularization. The created DSM was then partitioned from the viewpoints of reducing reworking and promoting modularization as described in Chapter 3. Figure 9 shows these results together with the rework and modularization areas.

As a result of partitioning, the three upper areas enclosed by red frames in the figure could be grouped as areas that enable modularization of acceleration from standstill, low-engine-speed booming noise, and fuel economy performance. In addition, element groups for which reworking cannot be avoided were gathered in the bottom right area enclosed by the red frame for the next optimal solution search.

Concrete performance design was performed using the sorted DSM as a reference. This investigation treated the engine characteristics as fixed values in subsequent processes in order to limit the target area. The rework area elements included many transmission area specifications and control specifications that play a central role in power transmission. In addition, the rework elements were determined to be the three elements of the recommended engine operating points (engine speed, engine torque), the torque converter characteristics (τ , κ), and the final gear ratio.

Next, relationships between elements within the rework area were clarified, and the calculation logic and evaluation criteria were determined.

Figure 10 shows an example of the logic used to calculate the transmission damping performance. An input/output equation approximation model was created beforehand using an analysis model that calculates the crank-end torque fluctuation based on the cylinder pressure measured on an engine bench. This approximation model was set as the crank-end excitation characteristics.

The recommended engine operating points (engine speed, engine torque) were input to the crank-end excitation characteristics, and the crank-end torque fluctuation at each engine speed was calculated.

Here, when the target wheel-axle excitation characteristics are set as the constraint, the damping performance demanded of the transmission can be obtained from the crank-end torque fluctuation and the wheel-axle excitation characteristics.

The evaluation criterion was that the calculated damping performance could not exceed the damping ratio of torque converters that can be used by the model to which the process was applied. The created calculation logic was mounted using a program and stored as a calculation module.

A calculation module was created in the manner described above for each element in the rework area, and calculation was performed by linking the respective inputs and outputs. The initial values of the three rework elements

were comprehensively sampled within the design space. The results of performing calculations within the rework area one time in the DSM order were taken as one sample, and approximately 400 samples were calculated. Figure 11 shows some of the calculation results.

Figure 11 shows an example of a tradeoff relationship where elements related to fuel economy performance, acceleration from standstill, and low-engine-speed booming noise mutually influence each other. Figure 11(a) shows the fuel economy performance in JC08 mode on the vertical axis and the 0-100 km/h acceleration time at wide-open throttle on the horizontal axis. The fuel economy is better and the acceleration time is shorter towards the upper left of this graph, so this shows good performance.

However, there are multiple samples that show good results, so two of these points were extracted and linked, and the damper performance was plotted as shown in Fig. 11(b). These figures show the damping factor on the

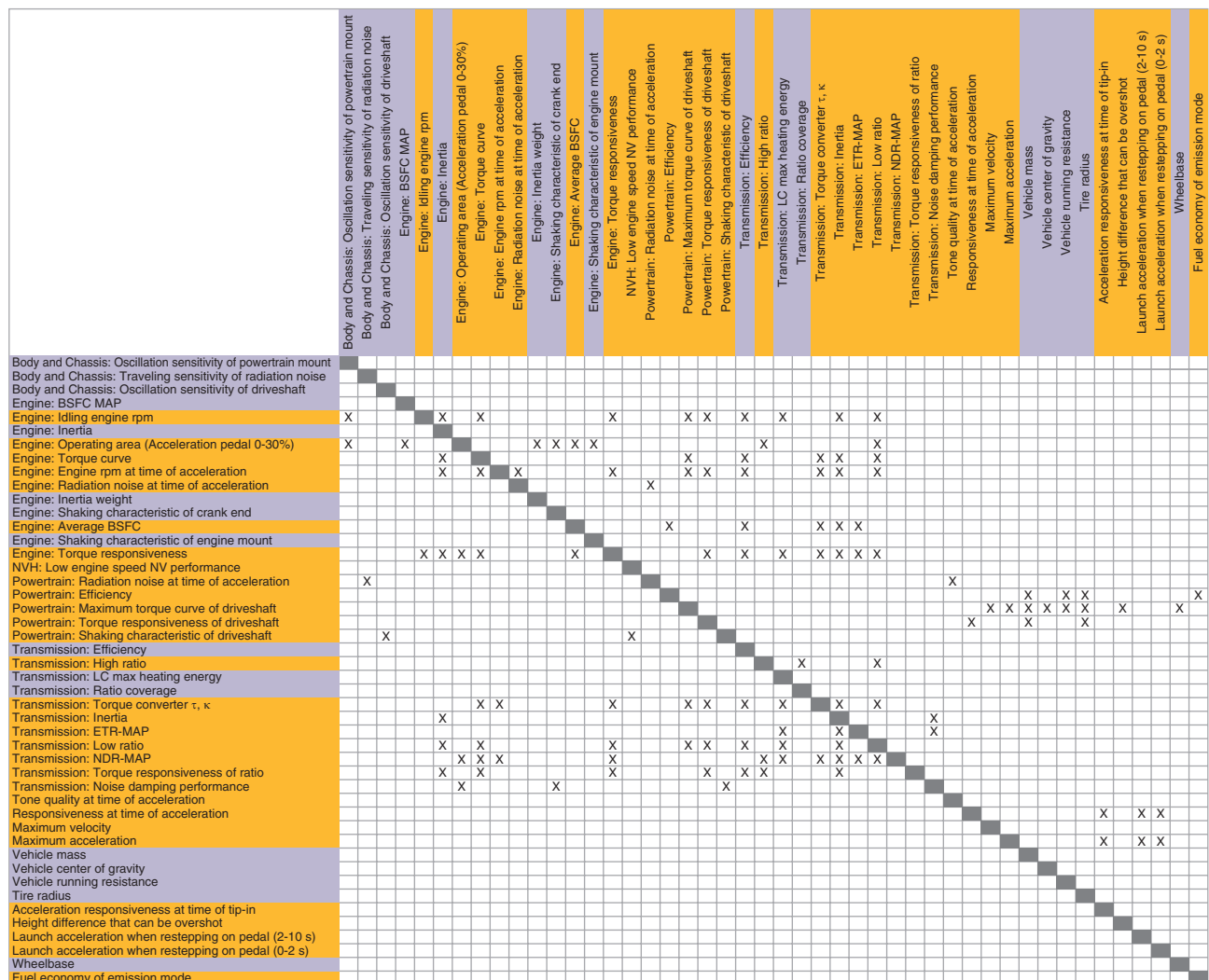


Fig. 8 DSM data before sorting

vertical axis and the engine speed on the horizontal axis. This shows that the damper performance differs even at the same fuel economy and acceleration from standstill, indicating that the usable torque converters also differ.

In addition, Fig. 11(c) shows the acceleration characteristics at that point. This figure shows the torque converter characteristics on the vertical axis and the acceleration response (acceleration after 0.5 s when accelerating at wide-open throttle from standstill) on the horizontal axis, and shows that the performance becomes better toward the right side. These results show that the two extracted samples exhibit the best fuel economy and acceleration time, but that other samples are better when viewed in terms of acceleration response.

Confirmation of simulation results of different levels from various viewpoints in this manner enables the system designer and PIC to discuss performance. However, the method of comparing the results separately for each sample point takes time to find the optimal solution.

Therefore, data preference-type data mining technology was applied to complete the process. An accurate RSM was created from approximately 400 calculated samples,

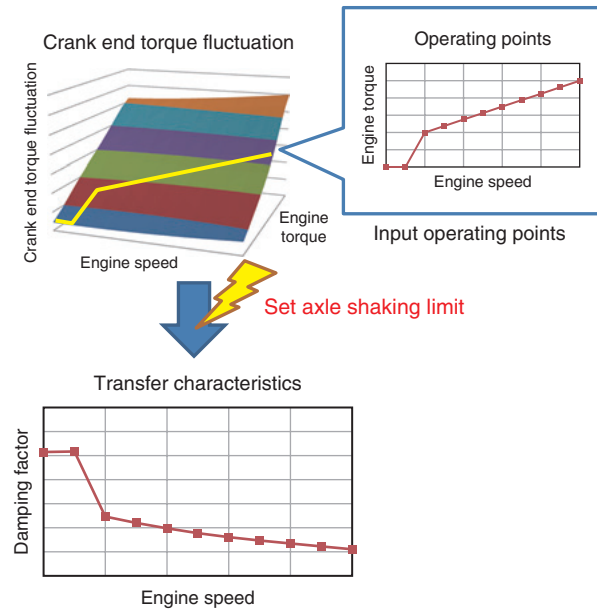


Fig. 10 Example of related definition

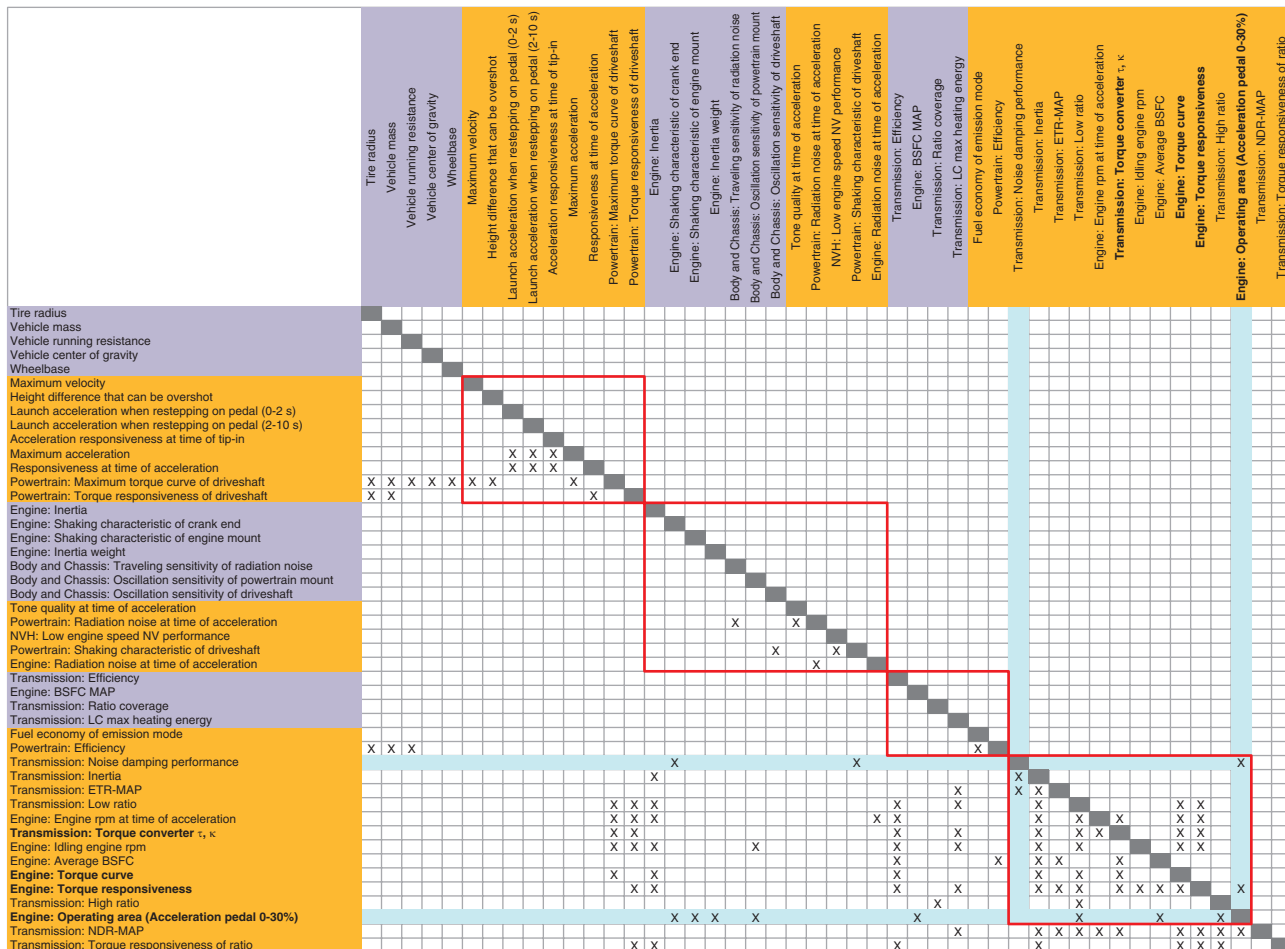


Fig. 9 DSM data after sorting

and this RSM was used to efficiently search the data. Figure 12 shows the process of narrowing down by data search, and demonstrates that as the process proceeds from

left to right, the optimal solution is narrowed down from among the solution space. The radar charts in the top part of the figure show typical performances and characteristics

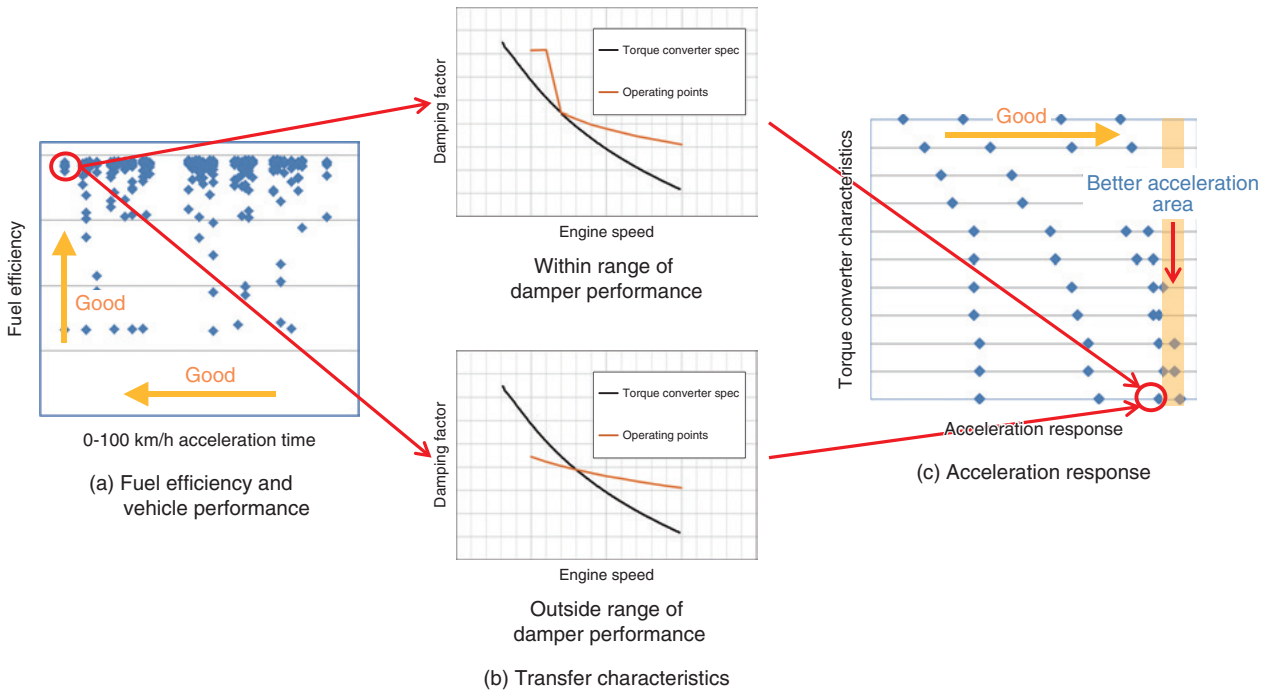


Fig. 11 Example of tradeoff relationship

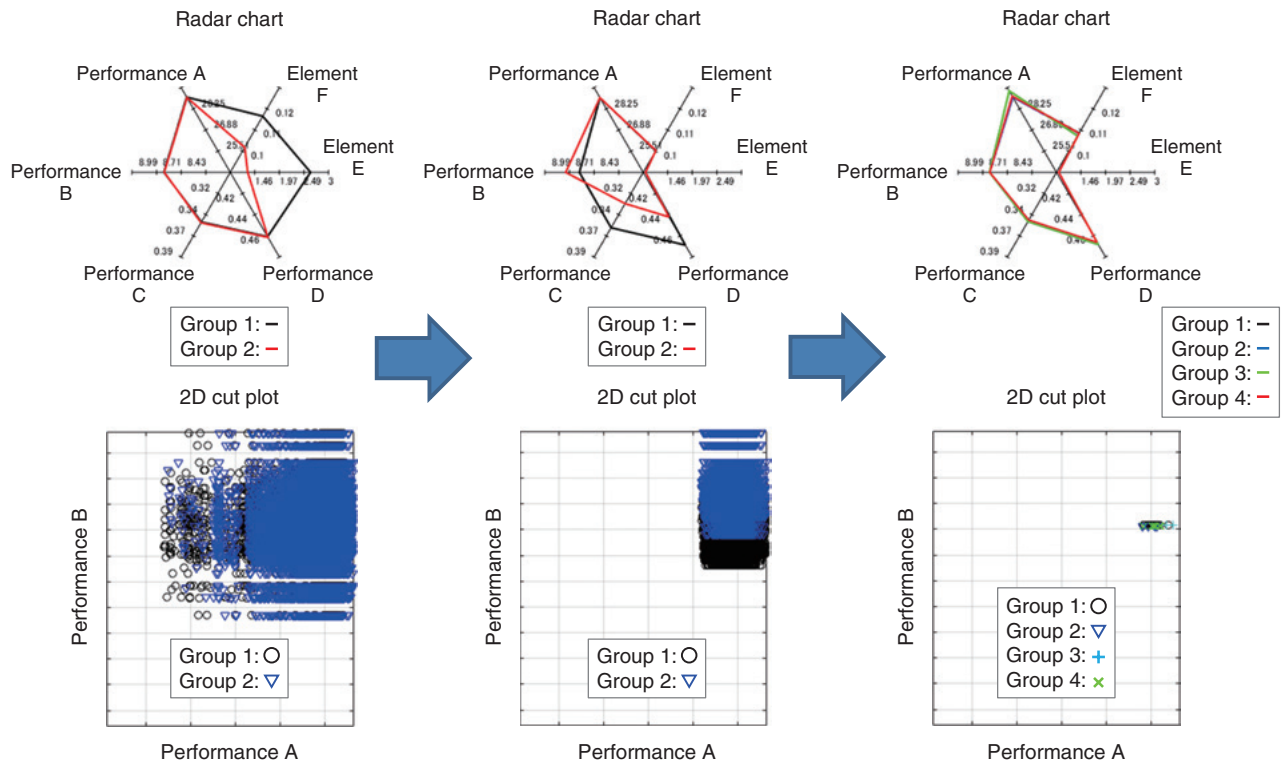


Fig. 12 Data mining process

to be confirmed together, such as fuel economy, acceleration from standstill, the torque converter type to be used, etc.

The 2D cut charts at the bottom cut out and plot only Performances A and B. The large number of sample points are automatically clustered, and the typical values are shown in the radar charts. The left figure shows the initial clustering results for the entire solution space, and shows two groups: Group 1 and Group 2. Here, Group 2 that has small Element E and Element F values was selected, and then clustering was performed again using only the selected Group 2 solutions to show the two groups in the center figure.

As a result of repeating this process, only solutions with roughly equal performance and characteristics remained as shown in the right figure. In this manner, the system designer and PIC were able to narrow down the solution with the optimal balance from among the solution space by continuing to select the radar chart that exhibits the desired tendencies while observing the balance between each performance value.

However, there are still many uncertain factors in the initial development stage and characteristics also change easily, so target performances coordinated beforehand using simulations in the initial stage may not necessarily be achieved by the actual component. Therefore, it is necessary to set robust solutions even in the stage when initially adjusting the target values.

As described above, an development process without reworking was proposed. This process consisted of sorting and calculating the influence of various performances in the fuel economy, acceleration from standstill, and low-engine-speed booming noise areas, and using data mining technology to determine the target system performance and the means for achieving that performance in the initial stage of powertrain system development.

6. Conclusion

The following development process was proposed that combines DSM, simulation calculation, and data mining technology in order to perform request-type development without reworking by determining the target system performance and the means for achieving that performance from the initial stage of development.

- (1) A DSM was used to clarify the relationships between system and subsystem performances and characteristics, and a reasonable development order was determined.
- (2) Logic for calculating the relationships between performances and characteristics was established, and enabled prediction of whether the target system performance can be achieved.
- (3) Simulations were performed in a wide design space for rework elements based on (1) and (2) above, and made it possible to search for other solutions in the event

the target performance is not achieved with the initial investigation structure.

- (4) Data mining technology was introduced to the vast calculation results, and enabled rapid extraction of the most reasonable solution from among multiple solution candidates.

Using powertrain system development as an example, performance was investigated for the areas of fuel economy performance, acceleration from standstill, and low-engine-speed booming noise, and it was verified that the system designer and PIC were able to narrow down the optimal balance solution.

References

- (1) Forsberg, K., Mooz, H.: The Relationship of Systems Engineering to the Project Cycle, First Annual Symposium of the National Council On Systems Engineering, (1991)
- (2) Schatz, B., Pretschner, A., Huber, F., Philipps, J.: Model-based Development of Embedded Systems, *Advances in Object-Oriented Information Systems*, p. 331-336, (2002)
- (3) Information-technology Promotion Agency, Japan: Kumikomi shisutemuno sentanteki moderubesu kaihatu jittai chosa, Research report (2012) (in Japanese)
- (4) Fujimoto, S., Ohtaka, Y., Kodo, S.: Development of High-performance Engine Plant Model and Sequential System Identification Environment for Realization of Model-based Development, *Honda R&D Technical Review*, Vol. 24, No. 2, p. 126-133
- (5) Ariyoshi, T., Harumatsu, M., Yano, K., Fukushima, T., Yamada, K., Narukawa, K.: Vehicle Performance Design System Using Data Mining Techniques towards Model-based Development, *Honda R&D Technical Review*, Vol. 29, No. 1, p. 188-196
- (6) Steward, D. V.: The design structure system: a method for managing the design of complex systems, *IEEE Trans Eng Manage*, Vol. 28, No. 3, p. 71-74, (1981)
- (7) Danilovic, M., Browning, T. R.: Managing complex product development projects with design structure matrices and domain mapping matrices, *International Journal of Project Management*, Vol. 25, Issue 3, p. 300-314, (2007)
- (8) Warfield, J. N.: Binary matrices in system modeling, *IEEE Trans Syst Man Cybern*, Vol. 3, No. 5, p. 441-449, (1973)
- (9) Gemala, D. A., Eppinger, S. D.: Methods for analyzing design procedures, *ASME Design Theory and Methodology*, Vol. 31, p. 227-233, (1991)
- (10) Pratab, A., Deb, K., Agrawal, S., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, In *KanGAL report 200001*, Indian Institute of Technology, Kanpur,

India, (2000)

- (11) Laumanns, M., Deb, K., Thiele, L., Zitzler, E.: Scalable test problems for evolutionary multi-objective optimization. tik-technical report, no. 112. p. 16-25, (2001)

■ Author ■



Shigeki FUJIMOTO



Keiji TOJO



Shoji KUSUNOKI



Toru SAITO