KEYWORDS Project planning · Project scheduling · Functional dependencies · Functional integration · Design structure matrix

Applying DSM Methodology to Improve the SCHEDULING OF CALIBRATION TASKS IN FUNCTIONAL INTEGRATION PROJECTS in the Automotive Industry

ABSTRACT

Functional integration projects in the automotive industry are highly complex development projects, determined by multi-level dependencies, iterative processing, limited resources, last-minute changes, and a multi-project working environment. Given this complexity, there is a risk that problems may arise in timing and quality. This paper presents a novel generic project scheduling technique for functional integration projects based on the design structure matrix methodology. This is intended to improve the planning of delivery dates and required resources and capacities, to ensure tighter synchronization between project teams, to help prioritize tasks in parallel projects, and to help anticipate changes to the project stages when development changes or delays need to be accommodated.

Thomas Gaertner
BMW Group

Sebastian Terstegen
Institute of Industrial Engineering and Ergonomics, RWTH Aachen University

Christopher M. Schlick
Institute of Industrial Engineering and Ergonomics, RWTH Aachen University

1. Introduction

Within the premium automotive segment, customers expect vehicles to operate faultlessly and exhibit optimal response behavior in any driving scenario. The dynamic response of modern vehicles in areas such as acceleration, load changing, fuel consumption and emissions are determined and set by a large number of electronic controls. To facilitate precise and accurate control at a granular level, the software functions are perfectly matched to the engine and the vehicle as a whole through the calibration of system parameters, including specific values, performance curves, engine maps and physical models *(Knoedler et al., 2005; Sung et al., 2007; Lygoe et al., 2010)*. This process, which can be highly complex, is described as "functional integration". The calibration process is one of the key elements in shaping the brand profile of the automobile manufacturer or original equipment manufacturer *(OEM) (Weber, 2009)*, and it is thus one of the development tasks usually carried out in-house – even at a time when development is increasingly outsourced to suppliers. Due to the growing number of software programs being integrated into automobiles and also due to higher requirements of driving characteristics, today's premium vehicles typically require the setting of tens of thousands of calibration parameters, with many more yet to come. The functional integration of the power train makes it necessary to coordinate a number of very different areas, such as driving response, fuel consumption, and emissions, as well as the strategy for choosing between the electric motor or the combustion engine in the case of hybrid vehicle systems.

The functional integration of these different areas cannot be implemented independently, as the areas are highly interdependent. And the introduction of hybrid vehicles increases both the impact and interdependency of calibration subjects. Frequently, calibration is required to resolve not only single issues but also multiple conflicting objectives. For example, most functions need to be calibrated over and over until the power train response perfectly matches customers' expectations with regard to minimum fuel consumption and emissions and maximum vehicle dynamics in any driving scenario. Consequently, all design engineers iteratively fine tune their calibration parameters right up to the project deadline. Some calibration teams use project schedules typically designed for conventional vehicles. But these project schedules have limited benefits, as they have been developed for a restricted subset of the general calibration process and are not synchronized for the rest. Therefore a delivery schedule for the functional integration process and a project schedule that resolves the conflicting objectives of multiple subjects had to be developed.

Functional integration, then, is incorporated into a multi-project environment as it is part of the complete vehicle project and, at the same time, entails a number of highly complex sub-projects, determined by the following factors:

- Multi-level dependencies, such as between the engine, gearbox, hybrid components and the overall vehicle
- Calibration tasks spread across different organizational units
- Iterative processing
- Limited resources, e.g. test vehicles and test benches
- Possible delays caused by upstream development issues
- Last-minute changes and new requirements
- A multi-project working environment

This complexity can cause problems in the development process, forcing OEMs to make costly investments in order to avoid project delays and quality issues.

The dramatic increase in software functions and calibration parameters in recent years has so far been met by increasing staff numbers. In our interviews with calibration engineers, we discovered that each calibration team develops different efficient strategies for the calibration process, which, however, do not work effectively throughout the functional integration process. For example, one team starts developing and approving each part of the software program at the same time and carries out an iterative optimization process with all functions afterwards, whilst the other team gradually develops and approves each function of the software program step by step and carries out optimization within each cycle. Both strategies have advantages and disadvantages with regards to the stability of the whole system and the connection of the approved software functions used by other calibration teams. Thus, the question arises as to which strategy should be chosen to reduce the number of iterative cycles for calibration through the optimal allocation of tasks as well as to reduce the number of resources, e.g. test vehicles and test benches, by an optimally timed and synchronized calibration process. Numerous attempts at optimizing the calibration process by combining different project schedules into one coherent schedule have failed due to the high level of complexity and the interdependencies between the schedules. However, especially given the growing volume of variants and cross-functional requirements, an in-depth revision of functional integration working methods and working structures is now needed.

For this purpose, a project was launched to develop a generic project schedule for functional integration in vehicle development projects and to establish project-specific functional integration schedules based on the overall generic project schedule. This project is intended to address the following objectives: *(1)* to improve the planning of project timelines and required resources and capacities; *(2)* to ensure tighter synchronization between the organizational units *(such as engine, gearbox, hybrid, overall vehicle)*; *(3)* to guarantee compatibility with subject-specific schedules and thereby ensure transparency and simplification; *(4)* to provide a guide for prioritizing tasks in parallel projects; *(5)* to measure project maturity for facilitating early identification of deviations and on-time corrective and support measures; *(6)* to anticipate changes in the project stages when development changes or delays need to be accommodated.

Another question concerns the project schedule's ability to monitor and control the calibration process in adequate detail and recognize and manage risks at an early stage. On the one hand, a very detailed project schedule is a prerequisite for efficient project controlling but might impair the balance between the utility of controlling-related information and the time and effort required to produce it. On the other hand, a rough project schedule could lead to the link between the application objects being overlooked. It is therefore important to give the project schedule an appropriate level of detail so that it can properly monitor and control the project, particularly because the schedule also has to comply with the application engineer's requirements, who perceive regulation and standardization of project schedules as an unnecessary restriction per se.

2. Structural and Process Scheduling

The established approach to generating a project schedule is to first divide the project into its constituent work packages using an object-, function-, timeline- or process-oriented breakdown structure *(Kerzner, 2009; Shtub et al., 2005)*. These work packages are defined both quantitatively and qualitatively, and refer to detailed tasks such as calibration activities and solution finding. The aim is to delineate the contents and intended results of the individual work packages. Furthermore, the tasks that have to be completed and the parties responsible for them need to be clearly defined. On the level of the individual processes, detailed scheduling needs to be prepared. Of particular importance is the identification and specification of interface points, i.e. the points where individual work packages are linked.

During the planning of the process schedule, the previously compiled and specified work packages are distributed across the project timeline. The project schedule provides a visual representation of the logical and chronological structure of the work packages across the project. The project schedule also specifies when the individual steps need to be completed in order for the ensuing steps to commence *(Shtub et al., 2005)*. Critical points in time, such as the completion of sub-goals, are highlighted as milestones; these divide the project timeline into more manageable stages.

3. Applying DSM to the Planning of Functional Integration

Due to the high level of structural complexity in the functional integration projects as well as the high level of dynamic complexity, there was little point in creating a project schedule and depicting the sequence of calibration activities without extending the methodology described above. The high level of structural complexity in functional integration *(Cardoso, 2006)* is caused by the diversity of calibration tasks, the large number of dependencies, and the different types of dependencies, such as functional or organizational dependencies. The dynamic complexity of functional integration projects *(Senge, 2006)* is a result of the dependencies in the calibration process not being static across the project timeline, for example because of needing to accommodate other calibrated parameters or functional revisions. Calibration activities may have local effects as well as effects on other calibration subjects, and dependencies may have unexpected outcomes.

The design structure matrix *(DSM)* is a method for representing and analyzing product and process architectures. It was developed by Steward *(1981)* for the analysis of design descriptions and it is also used today for the analysis of dependencies between products, organizational units and activities. In a DSM, the system is broken down into n elements to be investigated, e.g. a development project is broken down into its activities. Each element is entered as a label in the rows and columns of an n x n matrix.

In an activity-based or task-based DSM, also called process DSM, values in the matrix cells indicate an informational coupling between the corresponding activities. Feed-forward informational dependencies are documented below the diagonal in the matrix; feedback informational dependencies are documented above the diagonal *(Eppinger et al., 1994)*. The DSM thus provides a very useful framework for sequencing the flow of information in complex product development projects. This framework can be used to deduce work policies. Two activities can be processed in parallel, when they are informationally independent and when sufficient resources are available, but they have to be processed sequentially when one activity is dependent on the output information generated by an upstream activity. A common approach to decreasing project lead time is to overlap an upstream with a downstream activity through a periodic exchange of information. This enables the concurrent processing of the activities in contrast to the one-time transfer of finalized information in a sequential process. When preliminary information is utilized by the downstream activity too early, changes in information have to be incorporated in time-consuming subsequent iterations *(Krishnan et al., 1997)*.

Gaertner et al. *(2009)*, for example, extended the binary DSM by overlapping values that indicate the earliest starting point of the downstream activity as a percentage of the duration of the activity to be performed by the upstream activity. If activities are coupled, each activity requires information from another activity in order to be completed. To resolve the coupling, the processing of one activity is started with information assumptions and both activities have to be processed iteratively until the quality of the information output is satisfactory.

The DSM can also be used for the analysis of product architectures, whereby the parts of the product and the interdependencies between the parts are documented in the matrix. Product managers can use the product DSM to identify all the parts that are affected by a change in a specific part *(De Weck, 2007)*.

The domain mapping matrix *(DMM)* is an n x m matrix that connects different types of DSM with each other by using n elements of one DSM and m elements of another DSM as the title for the rows and columns *(Yassine et al., 2003a)*. This technique is suitable for capturing and analyzing dependencies between activities, parts of a product, design parameters, and organizational relationships.

According to Yassine et al. *(2003b)* and Huberman and Wilkinson *(2005)*, the work transformation matrix *(WTM)* is a static task-based DSM covering all parallel development tasks. Continuing feedback and feed-forward loops are modeled by the off-diagonal elements. The WTM enables the project manager to model, visualize and evaluate the dynamic dependencies among development tasks and to derive suggestions for reorganization. The task-centered approach is inspired by the work of Tatikonda and Rosenthal *(2000)*, who relate project complexity to the nature, quantity and magnitude of organizational subtasks and subtask interactions.

4. Approach to Creating a Project Schedule

Where a large number of interdependencies are present, pre-defined supply chains are needed to manage them. Such supply chains must specify which calibration characteristics need to be achieved when, by whom, at which maturity level, and who relies on each of the characteristics. When a common definition of the *(interim)* outcomes is to be achieved at the interface points and deadlines are to be met, synchronization of the project teams is much more structured than it is when simply giving a definition of the activities. This can be illustrated by the following example: For the purposes of synchronizing functional integration, the statement that an engine calibration is being trialed for winter usage at a certain time is less relevant than the statement that the engine will be able to start reliably at minus 20 °C at a given time. Consequently, the generic project schedule is focused on milestones and synchronization points rather than activities. The description of the supply chains comprises the contents of the generic project schedule. It was put together as follows:

1. Compilation of a project breakdown structure
2. Identification of the dependencies between calibration subjects; documentation within a design structure matrix (DSM)
3. Specification of dependencies in reference to the project timeline and calibration maturity using SIPOC diagrams (suppliers, inputs, processes, outputs and customers, cf. Section 4.3)
4. Identification of closely linked work packages through DSM-based clustering
5. Process scheduling
6. Optimization of tasks within a single time slot

4.1 Compilation of a Project Breakdown Structure

Firstly, a top-down project breakdown structure was compiled for the systematic ordering of the functional integration process. The first level of this shows the object orientation of the calibration process, which is in reference to the 26 calibration subjects *(such as the gearbox calibration)*; the second level shows its timeline orientation. The calibration process is divided into a defined number of time slots following the WTM method *(see Section 3)*. These time slots are determined by the delivery of new software and the integration of the different sets of calibration parameters into the new software. Each calibration subject is subdivided into four generic stages with pre-defined maturity levels. This provides not only an easier overview but also a rudimentary tool for monitoring project progress. The functional integration thus consists of 104 work packages.

4.2 Identification of the Dependencies between Calibration Subjects

Based on the 26 calibration subjects, a DSM was created that shows the main dependencies between the subjects. To identify the relevant dependencies, individual and group surveys of experts were conducted for each of the calibration subjects. The entries in the subject rows are based on the replies detailing the dependencies of each subject. The columns, on the other hand, state which other subjects depend on the row's subjects. The interviews with the experts showed that a calibration subject's dependency on other subjects is unlikely to cause problems. Conversely, the experts found it more difficult to identify other calibration subjects that are influenced by their own subject. The result was a DSM with a large number of dependencies and interdependencies *(see **Figure 1**)*. This demonstrates the high level of complexity in functional integration and indicates the necessity for more detailed project scheduling, as well as for the calibration subjects to be worked on iteratively and concurrently. Filling in this DSM along with the application engineers responsible for the different calibration areas primarily served to encourage further reflection on the dependencies and interdependencies. This was a good door opener for detailing the work packages.

4.3 Specification of Dependencies using SIPOC Diagrams

To obtain a detailed description of the dependencies between the work packages, the experts of the calibration subjects in question were again surveyed so that we were able to document which interim outputs need to be achieved by other subjects and which additional requirements need to be met by the specified deadlines in order for the project to be completed. In order for project maturity to be measured, the *(interim)* outputs also need to be quantifiable. This way, measuring the achieved outputs can provide feedback on the maturity of the work package. For example: in one of the gearbox work packages, a specific level of precision in the supplied engine torque is required at a specified time; this request may be included in all of the gearbox's work packages, but over the course of the project it will become stricter and more detailed. Note, however, that a function – or the calibration itself – will only mature linearly during development if the same hardware and software is used. When a function is migrated to new hardware, it usually has a lower maturity level than with the old hardware; in this case, it first needs to be elevated to the existing maturity level via the calibration.

For the potential delivery deadlines, points within the project timeline that were known to everyone working on the calibration process were selected. Also surveyed were the time spans needed for completing the individual work packages, the most important technical functions, the throughput times, and all the required personnel capacities and resources. For each work package, the data were documented using so-called SIPOC diagrams, which were expanded to include a chronological level. SIPOC diagrams are very useful tools for clearly describing, in a table, the suppliers, inputs, processes, outputs and customers *(Pyzdek and Keller, 2009)*. They are widely used in business process management, total quality management and Six Sigma programs, and process improvement. For the methodology at hand, the outputs correspond to the inputs requested by another calibration subject *(see* **Figure 2***)*.

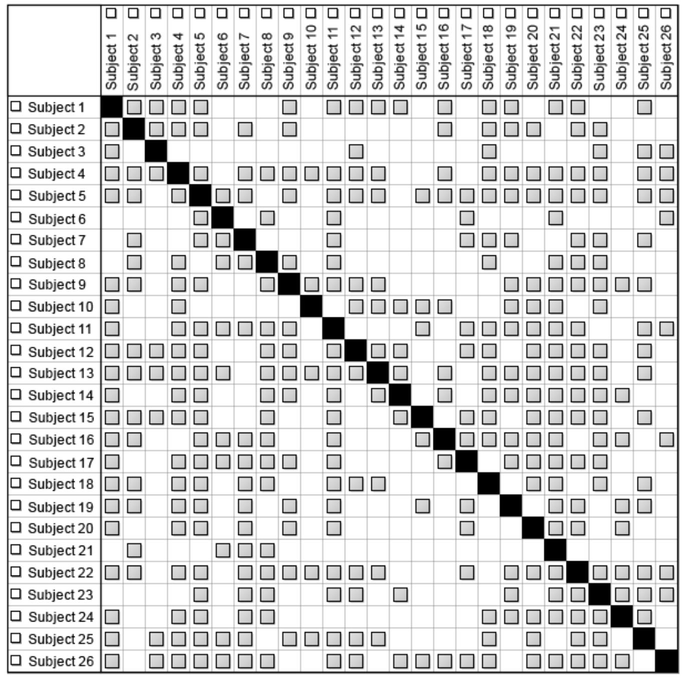


FIGURE 1. *Dependencies between calibration subjects*

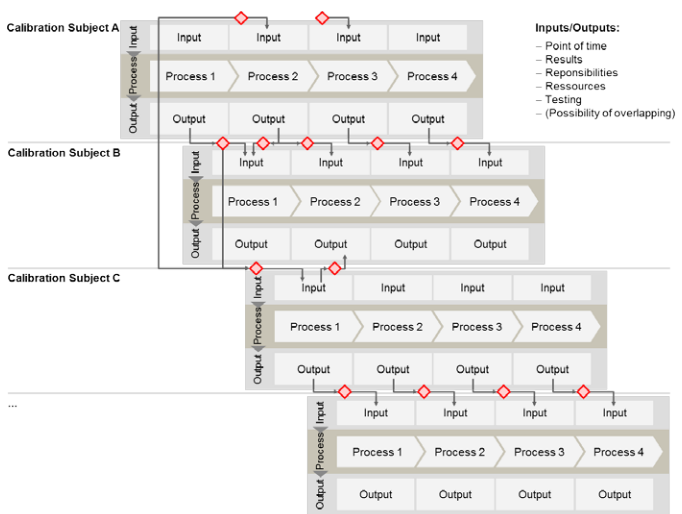


FIGURE 2. *Linking of SIPOC diagrams*

4.4 Identification of Closely Linked Work Packages

A clustering process based on the DSM was used to identify closely linked project work packages *(Steward, 1981; Yassine, 2010)*. Clustering of dependencies is a widely applied matrix-based computation method *(Maurer, 2007)* by which clusters can be identified that have a high number of internal dependencies and that represent subsets suitable for the definition of modules *(Lindemann et al., 2009)*. Danilovic and Browning *(2007)* present several clustering approaches of product and process DSMs in various applications. A clustering algorithm in the DSM toolbox provided by software tool Cambridge Advanced Modeller *(Wynn et al., 2010)* was used to automatically reorganize and group the matrix into strongly connected clusters.

The dependencies between the work packages of the calibration subjects documented in the SIPOC diagrams were transferred into a hybrid DSM consisting of the 104 object-oriented and function-oriented work packages. Each input-output relationship between the calibration subjects represents a dependency in the DSM, in relation to the maturity stages. The hybrid DSM derived from the SIPOC diagrams is displayed in **Figure 3**.

As Maurer *(2007)* stated, matrices are primarily suitable for purposes of information acquisition, whereas the implied structure of processes and projects are easier to understand through graph-based representations, even if both representations contain the same information. So, entering the dependencies gathered in the interviews would also have been possible using the DSM. However, the SIPOC diagrams are much more intuitive and easier to use than the DSM. Also, working with SIPOC diagrams is always in relation to the relevant section of the DSM. This makes it easier to fill in the DSM when there are widely distributed dependencies.

The hybrid DSM shows that most of the existing dependencies and interdependencies are located within the same stages of the timeline. Sequencing this hybrid DSM revealed that a purely forward-directed process cannot be achieved. To analyze the interdependencies between the calibration subjects within the individual maturity stages, the work packages of each maturity stage were manually compiled into clusters. Next, the clustering algorithm was again applied within each of the four clusters. An exemplary clustering result is shown in **Figure 4**.

The correctness of some of the identified clusters can be confirmed already, as even nowadays project meetings are being held on these clusters to discuss the interdependent subjects. A comparison between the clusters across the different maturity stages shows that there are clusters containing dependencies that remain stable throughout the project and clusters that differ distinctly from the clusters in the other project stages. To address this, the calibration teams should meet up in the best possible constellations when each maturity stage is reached, for example, to conduct joint trials.

4.5 Process Scheduling

An overview of the functional integration process was generated from the data of the SIPOC diagrams. This overview is called the "generic calibration project schedule"; taking the shape of a Gantt diagram extended by links, it depicts all the main aspects of the calibration process. These include not only the chronological dimension of the work packages' maturity stages but also resources, trials, etc. A further level of granularity shows all the requested interim outputs as well as the dependencies between the calibration subjects. The design engineers are thus provided with a network of synchronization points that can be used as a starting point for the subject-specific schedules. These, in turn, describe how to achieve the required results in reference to the input data.

Due to the concurrent engineering *(CE)* workflow process for the integrated, concurrent design of products and related processes, the functional integration projects are characterized by a strong informational coupling of tasks, numerous iterations, a high percentage of rework, and unpredictable product changes. When there are changes or delays in incoming deliveries, a DSM-

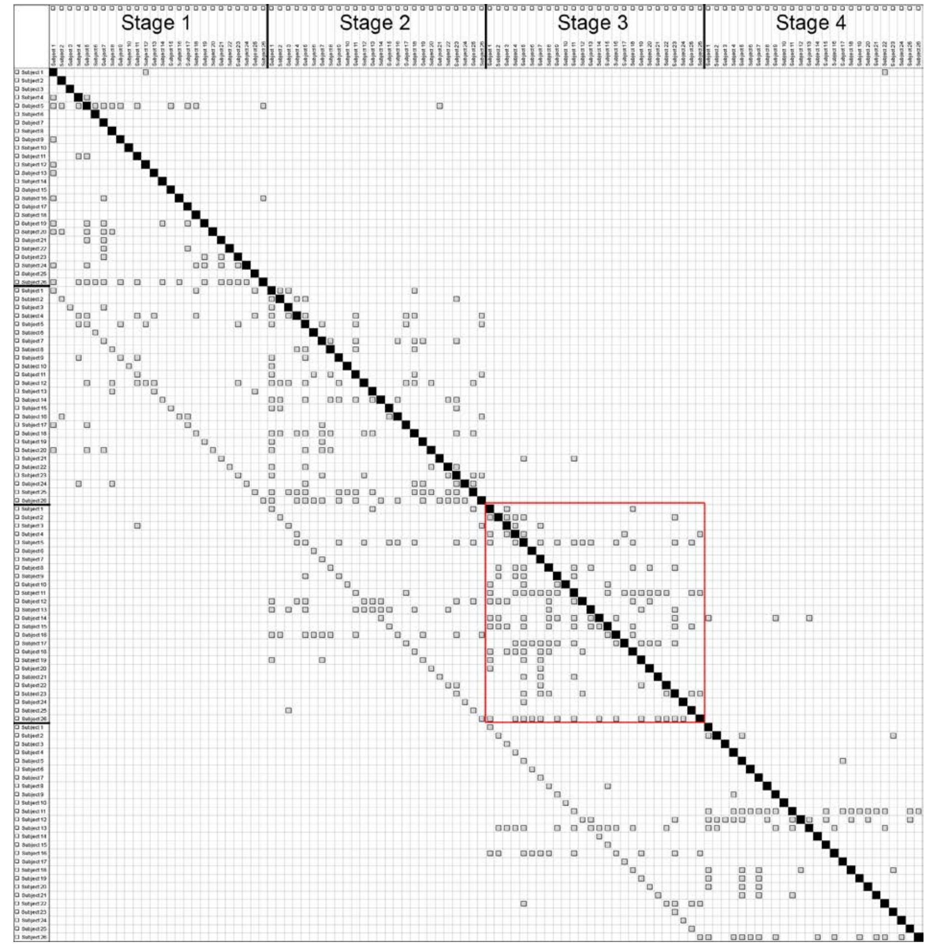


FIGURE 3. *Hybrid DSM with dependencies between calibration subjects in relation to maturity stages 1-4. The work packages of each maturity stage were compiled into clusters. The clustered DSM for maturity stage 3 (marked in red) is shown in Figure 4.*

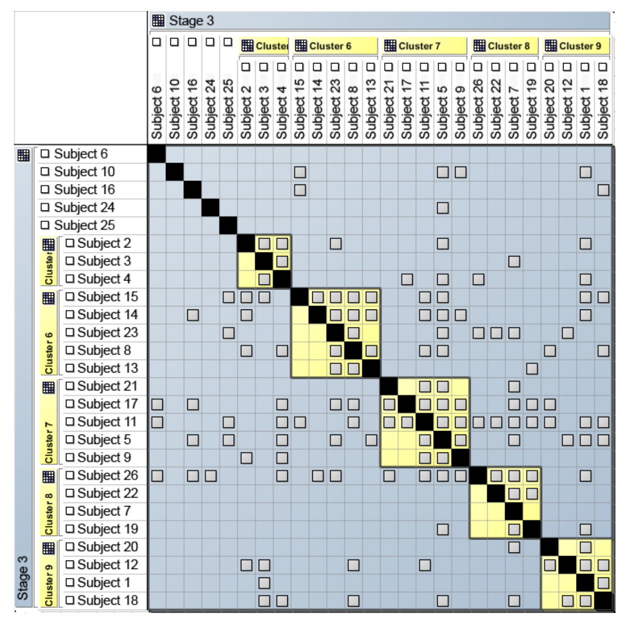


FIGURE 4. *Clustered DSM for maturity stage 3*

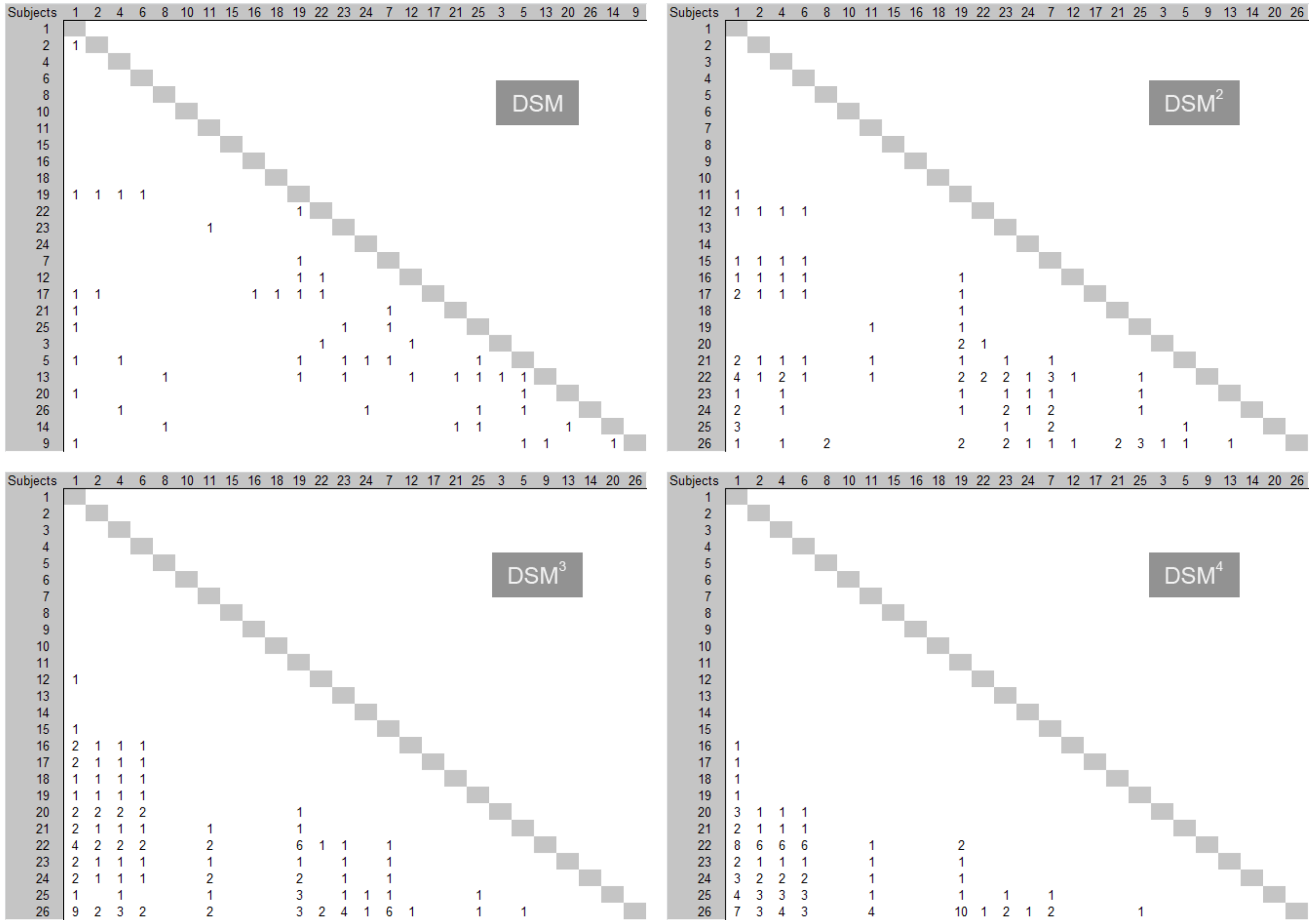


FIGURE 5. *Adjacency matrix of the functional integration project*

based simulation model developed by Gaertner et al. *(2009)* and Gaertner *(2011)* can be used to predict probability density functions of the project lead time and to analyze effects of these probabilities evolving from changes both to single subjects and parts of a product.

4.6 Validation of the Generic Calibration Project Schedule

Spearman's rank correlation coefficient was calculated to validate the generic calibration project schedule, and the non-parametric Wilcoxon signed-rank statistical hypothesis test was performed to evaluate the correlation between the data of the generic project schedule and the data of a real completed application project *(Bortz, 2005)*. The level of significance in the statistical test was set to a = 0.05. At r = 0.695, the Spearman's rank coefficient shows a significant positive correlation between the schedules of the calculated generic and the real projects using a two-tailed statistical test and a p-value = 0.01. The Wilcoxon signed-rank test shows a minimal rank-sum of w = 195 and a critical rank-sum of wcrit = 127 at a significance level of 5% *(Wilcoxon and Wilcox, 1964)*. Hence, the test shows that the schedules of the calculated generic and the real projects do not have significant deviations. Both the Spearman's rank correlation and the Wilcoxon signed-rank test indicate that a reliable project schedule can be developed using the generic calibration project schedule. Therefore it can be assumed that the generic calibration project schedule is particularly well suited for future planning of application projects.

4.7 Optimization of Tasks within a Single Phase

Results so far have shown that the methods are indeed well suited for analyzing the interdependencies between calibration subjects, but they reach their limit with the identification of the optimal project schedule if the duration of activities and the occurrence of feedback loops are not specified. For this reason, specific DSM-based analysis tools have been applied to the modeled matrices in each project stage. In this way, a methodological approach for the scheduling of functional integration projects could be conclusively developed.

Partitioning or sequencing are methods for respectively minimizing the number of activities within iteration loops and identifying clusters containing strongly interdependent activities. Tearing a partitioned DSM can further optimize it with regard to project scheduling, as Browning *(2001)* and Yassine *(2004)* proposed. Therefore, the DSMs with dependencies between calibration subjects that are based on the input-output relationships were partitioned in a first step in order to schedule the application process and to identify critical interdependencies between calibration subjects. The result of this operation was an automatically generated DSM, whose activities have been rearranged into a processing sequence with the shortest possible iteration loops. However, it is not appropriate to include the processing sequence indicated by the partitioned DSM into the project schedule since it is intended to integrate and parallelize activities and thereby to reduce the development time as much as possible. For this reason, further steps had to be taken toward the systematic parallelization of calibration subjects and corresponding activities.

By applying the adjacency matrix it is possible to identify activity paths in a system based on indirect dependencies. This method can be used for workflows that consist of feed-forward informational dependencies only, to identify the activity sequences with the longest duration. For further reading we refer to Maurer *(2007)*, who presents the background and applications of DSM-related analysis methods and describes in detail the feedback loop analysis by the "powers of the adjacency matrix." This method considers the interdependencies between activities. Analogously to the critical path in network diagram methods, sequences of activities and hence subjects that significantly determine the schedule of the functional integration process can be identified. Determining these so-called critical sequences of activities can be used as a basis both for prioritizing specific calibration subjects or activities and for monitoring a project and implementing appropriate steering measures for dealing with risks. For this purpose, the adjacency matrix was consolidated with a Gantt chart to display the activities of the functional integration project in a bar chart. First of all, the powers of the adjacency matrix were calculated *(**Figure 5**)*. Depending on the Boolean power, the length of the sequence of subjects with indirect dependencies within a project stage could be identified.

As a result, the longest sequence of indirect input-output dependencies and calibration subjects are displayed on a horizontal axis based on their logical input-output relationships *(**Figure 6**)*. This clear Gantt chart can be used for both scheduling and visualizing the potential effects of several calibration subjects on the application process. The red arrows in the Gantt chart mark the critical path of calibration subjects, i.e. the longest information supply chain of the process. It should be noted that the assumptions are valid only if the activities of the calibration subjects have approximately the same duration. The symbols displayed in the Gantt chart *(**Figure 6**)* indicate the informational interdependencies between two subjects, e.g. whether an output is requested by a certain calibration subject. In practice, the Gantt chart could, for example, be used to specifically analyze the red-colored outputs as well as to reflect on and to critically question interdependencies and the critical path of calibration subjects in a certain project stage. Thus the main objective of the developed planning methodology would be achieved: A systematic scheduling that visualizes critical project subjects and that can be used for verification and detailing in collaboration with calibration engineers. In general, the Gantt chart

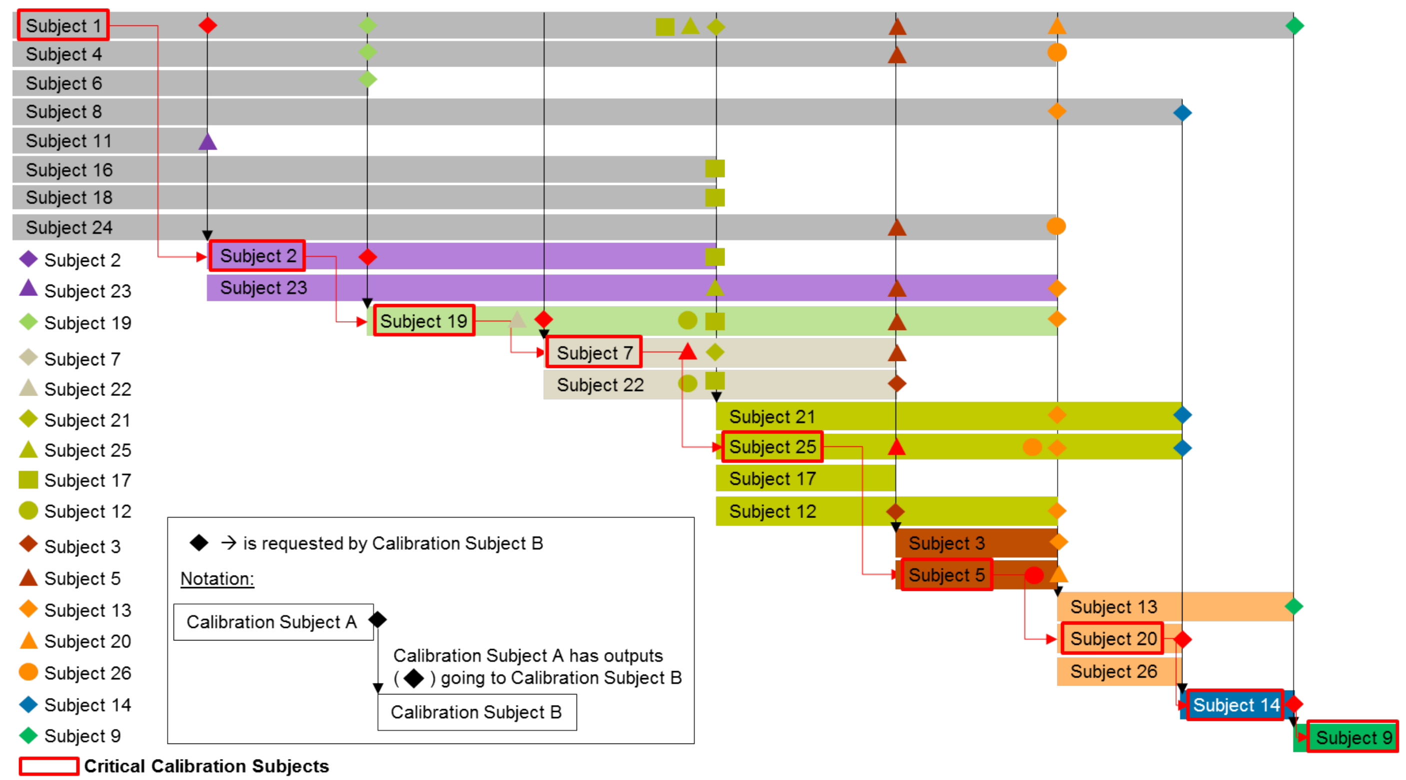


FIGURE 6. *Gantt chart for scheduling project stages of the functional integration project*

offers a very good basis for project managers wishing to investigate further relationships in a functional integration project.

5. Conclusion

The methodology of analyzing the input-output dependencies using the DSM provides a high degree of transparency in functional integration projects. The DSM method sequencing allows calibration subjects to be scheduled systematically and to be automated depending on their informational interdependencies. The main result of this study is a generic project schedule including the different project stages, which are shown in a Gantt chart. This provides many opportunities for responsible project managers: 1) it indicates which calibration subjects are serious threats to the successful completion of a project due to their longest activity sequences; 2) it illustrates which calibration subjects are mutually dependent and are thus more difficult to take into account in the scheduling; and 3) it offers a good basis for discussion and decision-making.

A comparison between the schedules automatically generated by adjacency matrices and the manually generated schedules showed that the planning methodology as developed accurately represents the interdependencies between calibration subjects.

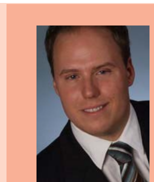
As a result of applying the described methodology, the following goals were achieved:

- Structuring the functional integration process using a generic project schedule
- Achieving unambiguous interface points between calibration subjects
- Determining synchronization points with defined and measurable interim outputs
- Facilitating scheduled iterative loops
- Integrating resource planning
- Ensuring change management structures for accommodating new requirements and unscheduled iterations

Due to the complexity of the calibration process and the related inaccurate progress evaluation reported by the engineers, it is very difficult to monitor and control projects without the support of appropriate project management methodologies. *(Kerzner, 2009; Shtub et al., 2005)* The generic project schedule for functional integration projects in the automotive industry developed in this study and presented in this paper should counter this.

Given the results presented in this work, the next step is to analyze and validate further essential causal relationships and interdependencies between multiple application subjects, as these are likely to significantly affect the calibration process during a project stage and, thus, the delivery dates of a functional integration project. Furthermore, potential for improvement in the collaboration of the calibration teams could be identified through further DSM-based analytical methods applied to the project stages. In the long term, the generic project schedule should also consider the duration of activities and the dependencies of external inputs – for example, the availability of test stands and vehicles – with the objective of identifying critical calibration subjects. In this context, the successful results obtained so far provide the basis for specifically selecting calibration subjects within a project stage for further analysis of functional dependencies, which was very difficult with the previous project planning methodology.

authors



Thomas Gaertner was born in Germany on June 22, 1979. He received the M.S. degree (Dipl.-Ing. Univ.) in Mechanical Engineering from Technical University of Munich in 2005 and the Ph.D. degree (Dr.-Ing.) in Mechanical Engineering from Aachen University of Technology in 2011. In 2010 and 2011 he worked as a senior consultant in the IT industry. Since 2011 he works at the BMW Group, currently as a project manager in the design transmissions department.

Sebastian Terstegen received the M.Sc. (Dipl.-Ing.) degree in Electrical Engineering from the University of Paderborn, Germany, in 2009. He currently works as a PhD student and research assistant at the Institute of Industrial Engineering and Ergonomics of RWTH Aachen University. His research focuses on the development of simulation and optimization techniques for project management.

Christopher M. Schlick was born in Germany on July 1, 1967. He received the M.S. degree (Dipl.-Ing.) in Electrical Engineering from Berlin University of Technology in 1992, Ph.D. degree (Dr.-Ing.) in Mechanical Engineering from Aachen University of Technology in 1999, and the Habilitation degree (Dr.-Ing. habil) also in Mechanical Engineering from Aachen University of Technology in 2004. He worked for the computer industry in 1992 and 1993 as a design engineer. From 1994 to 2000, he joined the Institute of Industrial Engineering and Ergonomics at RWTH Aachen University of Technology. From 2000 to 2004 he was the head of department of human–machine systems at the Research Institute for Communication, Information Processing and Ergonomics, Wachtberg, Germany. He is now a full professor of industrial engineering and ergonomics at RWTH Aachen University and deputy director of the Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE. His current interests are mathematical models of cooperative work in development projects and information-theoretic methods to evaluate emergent complexity.

references

Browning, T. R. *(2001).* Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions. IEEE Transactions on Engineering Management, 48*(3)*: 292-306.

Bortz, J. *(2005).* Statistik für Human- und Sozialwissenschaftler. Berlin, Germany: Springer.

Cardoso, J. *(2006).* Approaches to Compute Workflow Complexity. In F. Leymann, W. Reisig, S. R. Thatte, & W. van der Aalst *(Eds.)*, The Role of Business Processes in Service Oriented Architectures *(pp. 1-15)*. Schloss Dagstuhl, Germany: Internat. Begegnungs- und Forschungszentrum für Informatik.

Danilovic, M., & Browning, T. R. *(2007).* Managing complex product development projects with design structure matrices and domain mapping matrices. International Journal of Project Management, 25: 300-314.

De Weck, O. *(2007).* On the Rule of DSM in Designing Systems and Products for Changeability. In U. Lindemann, M. Danilovic, F. Deubzer, M. Maurer, & M. Kreimeyer *(Eds.)*, Proceedings of the 9th International DSM Conference *(pp. 311-323)*. Munich, Germany.

Eppinger, S. D., Whitney, D. E., Smith, R. P., & Gebala, D. A. *(1994).* A Model-Based Method for Organizing Tasks in Product Development. Research in Engineering Design, 6*(1)*: 1-13.

Gaertner, T. *(2011).* Simulationsmodell für das Projekt- und Änderungsmanagement in der Automobilentwicklung auf Basis der Design Structure Matrix. In C. M. Schlick *(Ed.)*, Schriftenreihe Industrial Engineering and Ergonomics. Aachen, Germany: Shaker.

Gaertner, T., Rohleder, N., Schlick, C., & Schulz, L. *(2009).* A Simulation Model for the Planning and Optimization of Product Development Projects based on the Design Structure Matrix. In M. F. Zaeh *(Ed.)*, Proceedings of the 3rd International Conference on Changeable, Agile, Reconfigurable and Virtual Production *(pp. 462-471)*. Munich, Germany: Herbert Utz.

Huberman, B. A., & Wilkinson, D. M. *(2005).* Performance variability and project dynamics. Comput Math Organ Theory, 11*(4)*: 307-332.

Kerzner, H. *(2009).* Project Management: A Systems Approach to Planning, Scheduling, and Controlling,. New York, NY: Wiley.

Knoedler, K., Poland, J., Merz, P., & Zell, A. *(2005).* Using Memetic Algorithms for Optimal Calibration of Automotive Internal Combustion Engines. In W. E. Hart, J. E. Smith, & N. Krasnogor *(Eds.)*, Recent Advances in Memetic Algorithms, vol 166 *(pp. 87-104)*. Berlin, Germany: Springer.

Krishnan, V., Eppinger, S. D., & Whitney, D. E. *(1997).* A Model-Based Framework to Overlap Product Development Activities. Management Science, 43*(4)*: 437-451.

Lindemann, U., Maurer, M., & Braun, T. *(2009).* Strcutural Complexity Management: An Approach for the Field of Product Design. Berlin, Germany: Springer.

Lygoe, R., Cary, M., & Fleming, P. *(2010).* A Many-Objective Optimisation Decision-Making Process Applied to Automotive Diesel Engine Calibration. In K. Deb et al. *(Eds.)*, Simulated Evolution and Learning, vol 6457 *(pp 638-646)*. Berlin, Germany: Springer.

Maurer, M. S. *(2007).* Structural Awareness in Complex Product Design. Dissertation, Technical University of Munich, Institute of Product Development.

Pyzdek, T., & Keller, P. *(2009).* The Six Sigma Handbook. New York City, NY: McGraw-Hill Professional.

Senge, P. M. *(2006).* The fifth discipline: The art and practice of the learning organization. New York, NY: Doubleday/Currency.

Shtub, A., Bard, J. F., & Globerson, S. *(2005).* Project management: Processes, methodologies, and economics. Upper Saddle River, NJ: Pearson Prentice Hall.

Steward, D. V. *(1981).* The Design Structure System: A Method for Managing the Design of Complex Systems. IEEE Transactions on Engineering Management, 28: 71-74.

Sung, A., Kloepper, F., Mitterer, A., Wachtmeister, G., & Zell, A. *(2007).* Model-based online optimisation in simulation and on the engine test bench. MTZ worldwide 68*(1)*: 16-19.

Tatikonda, M. V., & Rosenthal, S. R. *(2000).* Technology Novelty, Project Complexity and Product Development Project Execution Success. IEEE Transactions on Engineering Management, 47: 74-87.

Weber, J. *(2009).* Automotive Development Processes: Processes for Successful Customer Oriented Vehicle Development. Berlin, Germany: Springer.

Wilcoxon, F., & Wilcox, R. *(1964).* Some rapid approximate statistical procedures. New York, NY: Pearl River.

Wynn, D. C., Wyatt, D. F., Nair, S. M. T., & Clarkson, P. J. *(2010).* An introduction to the Cambridge Advanced Modeller. Presented at the 1st International Conference on Modelling and Management of Engineering Processes, Cambridge, UK.

Yassine, A. *(2004).* An Introduction to Modeling and Analyzing Complex Product Development Processes Using the Design Structure Matrix *(DSM)* Method. Quaderni di Management *(Italian Management Review)*, 9: 72-88.

Yassine, A., Whitney, D., Daleiden, S., & Lavine, J. *(2003a).* Connectivity Maps: Modeling and Analysing Relationships in Product Development Processes. Journal of Engineering Design, 14*(3)*: 377-394.

Yassine, A. A., Joglekar, N., Braha, D., Eppinger, S. D., & Whitney, D. *(2003b).* Information hiding in product development: the design churn effect. Research in Engineering Design, 14*(3)*: 145-161.

Yassine, A. A. *(2010).* Multi-Domain DSM: Simultaneous Optimization of Product, Process & People DSMs. In M. Kreimeyer, K. Eben, M. Maurer, U. Lindemann, & J. Clarkson *(Eds.)*, Managing Complexity by Modelling Dependencies: Proceedings of the 12th International DSM Conference *(pp. 319-332)*. Munich, Germany: Hanser.